

OPERATION MANUAL

SELF-PROGRAMMABLE TESTING AND SCORING SYSTEM



High Volume, Measurable, Computer Assisted Substance Abuse/
Rehabilitation Programs Specific To Large Prison Populations.

C.A.R.P.

Jerry Marzinsky B.A., M.Ed., C.P.C.

OPERATORS MANUAL

USER PROGRAMMABLE SCORING AND TESTING SYSTEM

PURPOSE / HISTORY/OVERVIEW:

The User Programmable Scoring and Testing System were developed for the purpose of providing a high volume, measurable, cost effective way to test and validate a variety of correctional programs. Prior to its development, inmates found numerous methods of cheating on rehabilitation and substance abuse programs. Such behavior embraced cheating of all kinds and ranged from the wide spread use of cheat sheets to the actual theft and distribution of the written tests and everything in between. Even when efforts were made to run valid and measurable programs, inmate behavior of this type remained effective in defeating testing of inmate knowledge. The result was that the cost and effort to operate a valid program inmates could not cheat was driven so high that prison staff did not put out the effort and expense necessary to make sure that inmates are learning what they are supposed to. The bottom line is that most prisoners will take the path of least resistance and cheat their way through rehabilitation oriented programs if possible, rendering even the most expensive programs, such as Hazeldens' Design for Living, ineffective.

It has been estimated that 80% of inmates within our prisons have been incarcerated for substance abuse issues. They are often sent to prison for drug related crimes and remain in denial of their substance abuse problems.

Since they are in denial, they do not feel that they need substance abuse programming. They blame their problems on others, viewing themselves as victims of society.

Such inmates full of negative emotion and resentment make them difficult to counsel or approach with more lawful and effective lifestyles. Even if they were approachable, their numbers are overwhelming, and far too numerous to deal with on an individual basis. This makes individual therapy not only extremely expensive but impractical.

Realizing the impracticality of individual therapy, prison therapists began to run groups. In these groups, therapists were faced with the massive denial systems of inmates and their supporting each other against the therapist. Often inmates played games and tuned out what the leader is trying to get across. If prison administration put pressure on them to attend groups, many did nothing more than daydream or "appear" to be interested once they arrived. Prisoners are acutely aware that the vast majority of groups run in prisons do not test their knowledge of anything upon completion. In many cases, it is possible for them to sit through virtually every prison group and graduate having learned absolutely nothing. Inmates attend such groups only with the ulterior motive of gaining a certificate to show parole or classification, not to learn anything.

Inmates know that they will not be tested, and as such do not make any effort to retain the material or incorporate it into their lives. This kind of inmate behavior appears common across all custody levels and different prison environments.

After many years of attempting to validly measure what inmates learn in large scale psycho-educational and substance abuse programs, it became crystal clear that if inmates were not validly tested, they learned little or nothing. It was imperative that a valid, secure, high volume testing and measurement tool was needed. (The detailed story of the development of this system is available in another paper.) The end result of years of effort to develop such a program resulted in the User Programmable Testing and Scoring System. This system was developed specifically for use with prisoners, is secure and provides both testing and validity for a variety of prison educational programs.

A hard lesson for us was the discovery that if inmates were not tested over what they were expected to learn, they were learning nothing. This system was developed to validly measure what inmates learned in a variety of large clinical and substance abuse programs. After years of studying the manner in which inmates cheated their way through programs rendering them ineffective and a waste of staff time and taxpayer money, this program was developed as an unbeatable, stable, inexpensive, high tech way to counter their numerous ways of cheating on programs and getting credit without having learned a thing.

This testing program accomplishes the task of verifying what inmates learned through the utilization of a large data base of test questions which are randomly presented as an examination on a computer screen.

Not only are the test questions randomly presented, the order of the answers to those test questions is also scrambled each time a test question is presented giving a double randomizing effect. If the data base of test questions is large enough, the probability of any two students receiving the same test is virtually nil. The effect is to neutralize all commonly used, methods of inmate cheating and return our ability to validly measure what inmates are actually learning. The effect of games, flattery and con jobs are also defeated. In short, the computer creates a kind of controlled chaos which only the computer can track. The success of all inmate cheat sheets depends on the predictability of presented test questions. Nothing about the administration of CAP tests is predictable. Two inmates sitting side by side taking the same test may not get the same test questions. If they do, the chances that the position of the answers will match up are remote. Despite all the scrambling going on, the computer tracks each correct answer to all presented test questions. Such a profound mixing up of questions and answers eliminates almost every conceivable form of cheating, permitting this system to give educators a valid, reliable measure of student knowledge. If the inmate does not meet performance standards, he is failed and told to try again, by the computer.

Since the tests are presented electronically, there is nothing for the inmate to steal. Cheat sheets are ineffective against such a system, as is sitting next to your buddy and trying to copy answers during a test. When operated within its design parameters, this system has defied all inmate attempts to beat it. Despite their massive denial systems, inmates must know the content material of any tested program, or they will fail the computer generated test. This system gives the program manager quick and accurate information about which inmates are making progress and which are not. It also gives a gross indication of the intelligence, educational level and mental health of the inmate.

Trying to accomplish this same task through the use of paper and pencil tests proved to be a nightmare. Inmates argued the composition of each test question, begged for extra points and still cheated in massive numbers. Program staff tired of dealing with the nonsense burned out in record numbers and didn't want to deal with paper and pencil tested programs. With the Computer Assisted Testing Program, program managers never have to grade, score or data bank any paper and pencil test information. The system does all of this automatically and then resets itself to administer the next test to a different inmate, all in record time.

This could not be accomplished in any other manner than through the use of computers. These machines afford tremendous advantages in the area of testing and validation of large scale inmate programs. In addition, they are also extremely efficient and inexpensive to operate. The implementation of this program format enables institutions to use inexpensive, obsolete computers in place of much more expensive program staff for the administration, scoring and recording of numerous high volume inmate programs. When set up and operated correctly, program administrators do not have to worry about these machines being utilized for any other purpose by inmates than the testing of inmates.

They cannot be used to access the internet and floppy drives can be disconnected to prevent inmates from putting in or taking out any information from the computer.

In addition to defeating prolific inmate or student cheating, many other security features have been incorporated to prevent tampering with the program. Computer misuse is a common concern of correctional administrators, and this system covers all angles and alleviates those concerns when the system is managed within its operational envelope.

One such feature incorporated into this system is the Test Timer. The timer serves two important functions. It keeps inmates who have not studied or are trying to cheat from hogging the testing machine. This allows for a steady and constant flow of students through the lab and massive program utilization. Secondly, it does not allow inmates time to try to break into the computer.

The instant the first test question comes onto the screen, a program test timer starts. Each student is given fifteen minutes to answer twenty test questions over a program study booklet. For a thirty page booklet with study questions outlined within for the inmate, fifteen minutes has proven to be more than enough time for students to complete the computer driven test.

When this timed test system is applied across the general inmate population, the program is able to identify those inmates who are having difficulty with reading and comprehension. It was discovered that when programs were validly tested and cheating eliminated, inmates began forming their own study groups to help each other study.

An unexpected byproduct of the system was that it proved capable of shifting through large inmate populations in very short periods of time. The system indirectly tests each inmate's reading ability and his ability to focus and concentrate. Consequently, the system can be used to assist in the identification of students who are having trouble. Inmates who are mentally ill and have slipped through the cracks of normal institutional screening often come to light during program testing. With a few simple questions these inmates can often be positively identified.

Due to the high volume of inmates this system can process, literally hundreds of inmates can be put through rehabilitative programs in the period of a month. Students measured test scores and their reaction to valid testing can give correctional administrators important information about individual students and their institutional population as a whole. Program managers who discover students who are unable to pass tests can then determine the reason. Students who are identified having trouble reading can be sent to GED or remedial classes to improve their skills. Inmates who are having severe trouble concentrating may need to see Psychology. Composite inmate test scores give a rough measure of the individual's verbal and intellectual abilities as well as their mental status.

The timed test feature mentioned above is only one of a number of sophisticated security devices built into this system. When considered as a whole, all these factors combine to give this program format the ability to accomplish the critical goal of validly testing correctional programs within menacing correctional populations while maintaining program security, integrity, and measurability. The system when operated correctly assures that even the most manipulative and recalcitrant inmates come out of a correctional programs knowing what the program was designed to teach them.

Even though this piece of software is "heavily armored" and has a substantial number of security systems built in, in order to accomplish these goals, the system **MUST** be operated within its design parameters. Giving inmate clerks the encrypted passwords to these systems as some staff program managers have done invites corruption. In possession of program passwords, inmate clerks have started lucrative businesses in the sale of tests to other inmates. Although program numbers remained high in such circumstances, inmates learned nothing. This happened in several different prison systems and every time inmate clerks were put in charge of these programs. In each case the corruption caused little concern to prison administrators who cared for nothing other than turning in high program numbers to the legislature. These programs had to be forcibly terminated. This program is a two edged sword. In the right hands, it can educate tens of thousand of inmates as to their problems and what to do about them. In the hands of corrupt inmate clerks, it is nothing more than a cheap tool to reinforce criminal behavior on a massive scale.

These systems should never be operated within a correctional environment without the presence of a staff program lab manager to monitor inmate activity on the computers. Despite the fact that this program has a substantial number of security safeguards, any unattended computer is vulnerable to inmate manipulation. Because of the number and complexity of the security devices built into this system, specifically to maintain program validity within correctional populations, this is one of the most secure programs available for use in prisons. However, given enough unmonitored access, and a competent hacker, there is virtually no system which cannot be broken into or manipulated by hackers with enough time and access to a targeted computer.

The recommendations with regard to how labs should be operated in this operations manual are the result of years of experience using these systems within correctional environments in different states. It is a condition of use that this system be operated according to the instructions in this manual. Again, there has not been a single case where inmate clerks given the security passwords to this program have not succumbed to the pressures of the inmate population and ended up using the programs to promote their own selfish ends. If this system is not operated strictly according to the instructions for secure operation found in this manual, no claim can be made for either its validity or security. These requirements take little time and effort, however, meeting the requirements for secure operation is a **MUST**. As such, program operators must clearly understand the program's operational limits. Since these are critically important, we will

outline.

Each one of these steps has been learned through the College of Hard Knocks and they have not been added to the operating requirements lightly. If these requirements are strictly followed, a large scale program will remain both valid and secure around the most treacherous of inmate hackers.

USER PROGRAMMABLE TESTING AND SCORING SYSTEM

SEQUENTIAL OVERVIEW OF PROGRAM SECURITY FEATURES

This testing system was designed to operate in one of the most manipulative and hostile psychological environments in the U.S., our prison system. The program is designed to remain secure even with long exposure to scores of criminal thinkers with all levels of education and computer literacy. As such, this testing system has had to have a large number of security features incorporated and is heavily armored against hackers.

The numerous security systems make this program more complex and cumbersome than similar systems operating in non-criminal environments such as schools. The overall system of security has many facets, each of which need to be thoroughly understood by the correctional program operator. The following section will give the program operator an overview as to the type and function of the various individual security aspects built into this testing program. There are many, and it may appear there are too many, over the years each and every one of them has proven necessary for this program to remain viable within correctional settings. These security systems will be summarized in the order in which they would be encountered during normal operation.

1. THE SYSTEM LOCK:

The first line of security is the system lock. This is an independent security lock program which prevents any unauthorized entry onto a dedicated testing computer. (A computer used only for testing of these programs.) This program runs independently of the other systems in the program. Its purpose is to deny unauthorized users access to the computer. On newer computers, the CMOS password serves this function. However, since this system was built to be inexpensive and to utilize older computers that prisons are throwing out, or those that may be purchased inexpensively in used computer stores, a system lock was necessary. These older machines work perfectly well for testing but are often not protected by a CMOS password. When installed onto the autoexec.bat of the computer, the system lock is the first program to initialize when

the testing program is fired up.

“Syslock” will ask the user to enter an access password twice. Only if the two entries match up, it will allow the user access to the computer.

If the two passwords do not match, the lock program will ask the user to reenter the password. If the program detects a mismatch, it will lock the computer and sound an audible alarm which cannot be disabled in any other manner but turning the computer off. This alarm should serve as a warning to the staff program manager that an inmate has attempted illegal access to the system. As a further security measure, the syslock password is encrypted. In short, it cannot be obtained by entering the disk operating system and asking the operating system to display the contents of the syslock file. Should a hacker attempt this, the only thing they would get would be a string of unintelligible pictures and letters. The system lock is tied into an audio alarm. If the user tries to circumvent syslock via keyboard commands, a large "lockout" will appear on the computer screen and the computer will sound an alarm. If it appears that a hacker is making random attempts to access the computer by trying different passwords, syslock will also sound an alarm. Syslock is impenetrable as long as specialized software not usually found in prisons is not brought against it. Such software would be considered contraband in most correctional settings. But that does not mean they can't get it. It is amazing what can be smuggled into prisons.

Syslock will keep unauthorized users out of your computers. It is however possible to circumvent syslock by booting the computer with a separate operating system. For this reason, inmates should not be permitted to carry computer disks into a testing lab. They should never be allowed to work on testing computers unsupervised. The testing programs residing on a computer can be run unless the system lock password is successfully negotiated.

Once the correct password has been fed into the system lock, the user will be given a menu of the various programs the computer is capable of testing. It is common for a computer to be programmed to test over several different clinical or substance abuse programs. One of the advantages of this system is that many different testing programs can be operated from a single computer. Different menus exist which hold different numbers of programs. The menu contains the names of the many different programs the program manager can select from. The menu is a separate program and operates independently of the main program. It is a DOS based program. The menu serves the function of a program distribution center. .

Once the system password is negotiated and a particular program is chosen for testing, the user will be confronted by an encrypted password that protects each individual program. Again, this password cannot be seen if the computer is somehow broken into and requested to display the code surrounding it. This password is both compiled and encrypted password.

2. THE PROGRAM ENTRY PASSWORD:

Each separate program that tests over a particular topic is protected by its own individual password. For example, Alcohol Education has its own password as does Drug Education and Aids Education. Like all the passwords used in this system, it is also encrypted so it cannot be seen or broken by typing out a readme into the operating system. This is the second level of program protection that needs to be negotiated prior to program entry, the first level being syslock or the system lock.

SECURITY FEATURES WITHIN THE PASSWORD:

If any credit at all is given inmates for the completion of these programs, there will be inmates attempting to break into the system. As a staff program operator goes to feed encrypted passwords into the system to start it up. One can count on some inmate somewhere along the line trying to look over his shoulder at the key strokes. More often than not they may not get the entire password and will try to feed in different combinations containing parts to break into a testing computer.

In order to prevent a hacker from feeding in large numbers of passwords, to overcome this aspect of the program's security, if three wrong passwords are put into the system, the program will sound an audible alarm. The alarm is loud enough to get the attention of any staff in the vicinity of the computer and there should always be a staff member around when these systems are being operated. The computer sounds an alarm for approximately six seconds. The password program then resets itself and sounds the alarm again if the hacker tries again. The alarm will continue to repeatedly sound as long as the person attempting entry fails to come up with the correct password.

If it is suspected that an inmate might have gained access to your password, the program permits you to change your password at any time. The new password will also be encrypted into an unreadable form when stored. A password change can be initiated at any point where the program asks you if you want to change your password. It is recommended that passwords be changed periodically and that all the various individual programs be given the same password.

For security reasons, if you do elect to change your password, the program will not display the letters you type onto the computer. What will appear on the screen will be a series of XXXX's as you type in your new password. This prevents inmates from reading what the password is from the computer screen. The safest option to maintain security of these systems is not to allow any inmate into a testing lab until all the

program operator feeds in all the passwords and brings up all the computers to their ready state. It is critical that the program operator not permit students into the room until the system is to the point of asking for a students name and number.

Once past the program entry password for a particular subject is initiated, a whole new array of security programs is activated. At this point in the program, all the security locks are operational and the individual program testing over a particular subject is placed into a secure locked testing loop that cannot be broken out of. This was done to prevent inmates from attempting to break out of the testing part of the program. The locked testing loop does not permit access to any other part of the program or computer. Once the program enters the testing phase, the program will ask for the students name and number. It is at this point that the student should be allowed to station himself at the computer and sign in with his or her individual unique number. We utilized the inmates prison number as an ID. The program recognizes students only by number as many have the same names. Students very quickly pick up on how to do this, and it saves the program manager much time and effort over having to sign the student on. We will go into much greater detail about how to do this later on.

The student must sign onto the computer with his or her own unique correctional number in order to access their personal testing file. The student's personal file gives the student a complete record of his testing progress including a complete history of all of his or her test scores and the dates the tests were taken. Once their personal record is brought up to the screen, the inmate can then choose the particular program test they want to take.

As the program flashes the inmates test history onto the screen, it is easy for the program monitor to check to see that the inmate is taking the next test they are supposed to be taking. Strange enough, we frequently see inmates who take a study booklet back after having made a low passing score and study more then take a test again to bring their score up.

3. DISPLAY OF USER NAME AND IDENTIFICATION ON COMPUTER MONITOR DURING TESTING:

Once the student signs into the computer and begins testing, the next security device initiates, a constant display of the user name and ID at the top of the monitor. This is because some inmates, in their infinite quest to cheat have given other inmates their ID numbers and paid them to go to the lab and sign into a particular testing computer with their number and take a test for them. Since the computer has no way to verify the actual identity of the person sitting at the keyboard, without staff being at the helm of these programs, inmates have succeeded in beating the system in this manner. If an inmate clerk is monitoring the lab, he risks a serious beating by other inmates if he exposes this fraud. Any inmate clerk who has any kind of control over a lab other than

collecting and distributing testing booklets for people finishing one section and moving to another, are at serious risk if they don't help other inmates cheat their way through. Again, there is not one single instance on record where an inmate clerk has been put in charge of these programs where it was not shortly compromised there after.

If you don't have the staff to monitor a testing lab, you shouldn't be running one

This feature makes it relatively easy to periodically check the inmates ID number and picture ID against the name and number presented on the computer screen. This ID display remains constant the entire time the student is testing.

4. THE AUTOMATED PROGRAM TEST TIMER:

At the same time the first test question comes up, a test timer goes into effect. This is the third security feature and operates automatically. This feature measures out fifteen minutes before the computer shuts down and sounds an alarm. The reason for this feature was that inmates instinctively want to test the system and see if they can guess their way through a test or get through with little study. They will come into labs in record numbers with this mindset and would sit at the testing computers for long periods of time denying the use of the computer to other students while they were fooling around seeing if they could beat the system. This happened so regularly that the fifteen minute timer had to be added. This feature gives each inmate fifteen minutes maximum to finish a twenty question electronically administered test over a particular study packet. The inmate has little spare time for anything other than the actual test. This unique feature successfully limits inmate exposure to the testing system.

It forces inmates to remain on task. Without it, you will find significant number of inmates taking large amounts of time to attempt to guess their way through test questions.

This program segment times each test from the instant the first question appears on the screen down to the fifteen minute time limit. This limit can be programmatically adjusted for special populations. The average inmate can finish a twenty question test in approximately ten minutes. The programmed test timer flashes a series of warnings on the screen notifying the inmate of precisely how much time he or she has left prior to the computer locking itself up. If the inmate exceeds the time limits, the computer will automatically sound an alarm and locks itself up. The system can quickly be brought up again by feeding in an access code.

5. AUTOMATED RANDOMIZATION OF TEST QUESTION SELECTION AND SCRAMBLING OF TEST QUESTION ANSWERS WITH EACH QUESTION PRESENTATION.

If you can't measure what inmates are learning in correctional programs, you can safely assume that they aren't learning anything. Because of prolific inmate cheating and the enormous time and effort required to validly measure what inmates are actually learning, most prison rehabilitation programs don't even attempt to measure inmate learning.

To complicate matters, inmates offer very stiff resistance to being subjected to testing.

Within the prison environment EVERY attempt to validly measure what inmates had learned in any program was met by numerous innovative methods of inmate cheating. Any attempt to attain measurable program results was confounded by inmate resistance and cheating. It was discovered that any test that was in the least predictable or amenable to being stolen, would be stolen or defeated by various means of inmate cheating within six months. The problem is so difficult to deal with that most conventional substance abuse and psycho-educational programs do not even attempt to measure what inmates are learning through any kind of testing.

Because cheating proved to be such an immense problem to valid measurement of inmate learning in programs, Computer Assisted Programs were designed so they could not be stolen. They were also designed to be unpredictable. This was accomplished with a double randomization test administration component. This subprogram creates a controlled chaos which destroys all predictability to the test taking process. This feature is active at two levels. It is applied to both the selection of test questions and the scrambling of test question answers. This feature completely scrambles the order in which the test questions are presented to the inmate. At the same time, it also randomly scrambles the order of the test answers each time a test question is pulled up. Thus, every test question is doubly randomized, making it impossible for them to determine which test questions they will be presented on a test. At the same time, the answers to every question are completely scrambled each time a random question is presented. The computer keeps track of the position of the correct answer. These features assure that no two inmates, even if they are taking a test over the same study material, will get the same test. Bringing this system to bear in the measurement of inmate programming renders even the most elaborate inmate cheat sheet impotent. With five answer options to each test question, there is a standard of 625 different answer position combinations.

The more test questions programmed into the test data bank, the smaller the probability that inmates taking the same test at the same time will even get the same test over the same material. The system requires that twenty test questions over each study packet be programmed into the computer. If the minimum of only 20 test questions are put into the data bank, each of those 20 questions will be presented every time that particular test is given. However, the order in which those questions will be presented and the order of the answers to those 20 questions will constantly be scrambled. However, each of the 20 questions will be presented every time that particular test is given. If 40

test questions are programmed into the data bank, only one half of them will be administered to any student at any one sitting and the number of possible combinations of random presentation becomes astronomical. The more test questions the user programs into the system, the greater the probability that no two students will ever get the same test questions over the same study packet ever.

The software was designed to be user programmable, and it is easy to program additional test questions into the system. Giving the user the ability to program this system allows a flexibility not found on many secure testing systems of this type. Many similar systems want to keep the programming under their control. This system allows program administrators to design their own test questions and test over materials they deem important for students to know.

Since the test questions administered by the computer are randomly selected prior to presentation, they will always be numbered differently. The software scrambles the test questions in such a manner that test question one for one on computer A. may appear as test question 18 for a second inmate on computer B. The more test questions that are placed into the programs data bank, the more material you will be able to test and the more secure your program will be.

6. THE LOCKED LOOP \ RESET TESTING SEQUENCE:

It was discovered early on that inmates with even a little knowledge of computers would attempt to break out of the test they were taking and get a look at another test, or try to randomly take another test to see if they could pass it by chance.

They would also try to break out of the test being presented and attempt to break into other sections of the testing program. It became clear early that inmate cheating had now moved out of the arena of stealing paper and pencil tests into the high tech arena of breaking in the computer or navigating to other parts of the testing program. Although this proved a significant problem, we had managed to reduce the number of inmates who cheat from the entire population of prisoners who could construct a tiny cheat sheet, to the far fewer number who were computer knowledgeable enough to break into either the computer or the program.

The Locked Testing Loop serves the purpose of locking or confining the test taker to one section of the program, the test. The locked testing loop locks the student into the test taking part of the computer program and does not permit the student into any other part of the computer or the program. When finished, this program sub-component sounds an audible alarm alerting the program manager to the fact that the inmate has finished his test and has no further business on the computer. The system immediately displays the students test score and resets the system to administer yet another test to

another student.

From the instant the first test question is displayed on the computer screen, the program is in the locked testing loop. This feature is not evident until a student tries to break out of the program. The computer is limited to responding only to the five key strokes necessary to answer the test questions. All keyboard escape or program termination combinations are blocked, such as control, alternate, delete which will usually allow any program to be terminated.

The keyboard only responds to the input from five keys, A, B, C, D and E, the five multiple choice test answers. This system effectively prevents computer literate inmates from accessing other areas of the program or the computer's Disk Operating System. The overall effect is to keep students from playing on the computers and maintaining a rapid flow of students through the testing labs as efficiently as possible. In conjunction with a competent lab manager, no student is permitted more than 15 minutes of access to the computer. It is essential that a staff member respond to the program alarms. An inmate clerk subject to numerous pressures from the inmate population will not be able to do this. Since the computer cannot make students leave once finished their test, a paid staff member has to pick up this function. An inmate clerk should not be put in the situation of keeping the flow of inmates moving through a lab by asking those who are attempting to manipulate the system to leave.

Locking out the keyboard except for authorized responses goes a long way toward preventing attempted inmate tampering during inmate testing. Inmates have proven themselves to be experts at distracting staff in order to perpetrate schemes to achieve their own ends. The program lab operator like any other human is subject to temporary distraction.

With this body of features, the system is designed to be able to withstand short periods of inattentiveness on the part of the lab operator and remain solvent in the face of the most knowledgeable computer hacker. However, the lab should never, ever be run without a staff member present or be turned over to an inmate trustee. We have closed down several labs where we saw this happen. Again, this program is a two edged sword. It can do a great deal of good if run appropriately. If it is turned over to inmates to run, it will shortly be used to reinforce criminal behavior at a level far surpassing any other inmate program out there.

7. ENCRYPTED PASSWORD PROTECT FOR THE EDIT TEST QUESTION SECTION:

One of the most desirable features of this program is that program test questions can be easily programmed into the system for testing by the program manager if they have access to the encrypted password for the program/edit section. This gives the user the

complete flexibility to test any program material that can be broken down into a multiple choice format. Current test questions can be changed or edited at any time. Test questions that have proved too confusing or problematic can be quickly and easily eliminated.

Test questions are stored within a data bank in the system. Most inmate computer hackers are not going to be very concerned about breaking into any other area of the program other than where the test questions can be found. Shortly after this program was implemented on the front lines, this proved true.

Inmate hackers in unsupervised programs broke into the system and looked at the section of the program where test questions were stored. Since no computer was ever hooked up to a printer, they had to write out these questions by hand if they were going to steal them. This took an inordinate amount of time putting the hacker at risk. The randomization of program answer presentation still presented a formidable challenge to inmate cheat sheets even if hackers were able to view the test questions.

In order to accomplish a break in to the test questions themselves, hackers would need to get through the system lock; they would then have to get through the program password and the test question section of the program itself. This section is protected by an encrypted password. Past this, there is one other unbeatable layer of protection for the test questions that we will speak about later. This last layer of protection will cause the program to electronically destroy itself rather than give up test questions to a hacker. We will speak more of this feature later. Prison staff who are incompetent enough to give some of the passwords to this system to inmate clerks may come into their labs one day to find their software non-functional because an inmate hacker tried to get to far into the system.

8. TEST QUESTION ENCRYPTION SYSTEM:

It became clear with the inception of rewarding inmates for completion of rehabilitation programs within prisons that they could and would work tirelessly to steal and sell any tests associated with such programs. Prison yard businesses based on the sale or barter in stolen tests proved very profitable for inmates clever enough to figure out ways to steal them.

Because the retired computers utilized to run these programs were not hooked to printers and were often kept in locked lab rooms, repeatedly prison program managers felt safe in giving their inmate clerks the initial passwords to bring the program up. Although they were repeatedly warned against giving inmates any of the passwords to this system, this saved them a few minutes. Shortly thereafter, prisoners were hacking their way into the computer to where the test questions were stored.

The Test Question Encryption System was designed to counter any such break in's. This sub-program is one of the front lines of security against break in once the User Programmable Testing and Scoring System is shut down in its resting state. This is the state in which most of these systems spend the most time and one where they are often not monitored by staff. There was one instance where an inmate clerk cleaning the floors in a lab carried a contraband computer boot disk in with him and broke into one of the testing machines. Taking these two incidents into account, a test encryption system was put into place. This program encrypted all written tests stored within the computer into gibberish upon program shut down and restored the encrypted tests once again upon program initialization. This encryption system could not be defeated.

This subprogram encrypts then locks and stores all test questions in an undecipherable form upon exiting the program. If a computer hacker breaks into the system during off hours and tries to copy the tests using a separate DOS disk, the only thing he will get will be encoded gibberish. The tests cannot be decoded unless the system is run up correctly and the appropriate passwords are used. Upon start-up, and once the correct program entry password is punched in, the encryption program will kick in and unencrypt each test one by one back into readable form. After feeding in the correct password, the program operator will see each test listed on the screen in sequential order as it is being unencrypted. Upon program shutdown, the computer should never just be flipped off. The quit option from the main menu must be exercised in order for the system to encrypt all tests for storage. If the power switch to the computer is merely flipped to turn off the computer after testing, the software will not encrypt the tests. The exit the program option must be utilized for the tests to be encrypted.

9. MONITORING OF COMPUTER LABS BY TRAINED STAFF:

Although this is a heavily armored free standing program, of all the security features the single most important is the specific human being chosen to monitor your computer labs. Nothing that this program does can compensate for a good, security oriented level-headed lab operator who understands the importance of keeping the labs and computers secure. Unfortunately, this is the security feature that has failed this program the most often. The single biggest mistake a warden can make is to assign a counselor to operate this lab who has no interest in programs and who will not operate it according to its design regulations. These labs can produce extremely high volumes of program numbers and some wardens have aptly demonstrated that they have little concern over the validity of the program, but only want the numbers to turn into their superiors to make it look like they are doing something. Unfortunately this has happened repeatedly and when discovered, those programs were shut down.

Unless operated on laptops, this system is not considered mobile. It cannot move out of harms way. Even a lion chained down in the middle of a city of ant hills will eventually be killed and eaten by the ants. You computer cannot get up and run away

from inmates who will be making constant assaults on it to get to the tests which if sold would make a very lucrative inmate business. Prison program labs are often stationary and subject to many different kinds of onslaughts from inmates who have an unlimited amount of time to think of a ways to try to steal tests.

The development of this system was like a brisk chess game of inmate attack and program development counter measures. It has been played in a in a variety of correctional systems and across different states where this system has been set into operation. This system will maintain its validity if strictly operated within its design parameters.

However, we cannot emphasize strongly enough that ***NO TESTING COMPUTER SHOULD EVER BE LEFT UNATTENDED AMOUNG INMATES. INMATES SHOULD NEVER BE ALLOWED TO TAKE THE PLACE OF STAFF IN THE INITIALIZATION, RUNNING, MONITORING, OR SHUT DOWN OF COMPUTERS USED FOR TESTING OF PROGRAMS.*** Inmates should never be in possession of any of the system passwords at any time or even be in the area when such passwords are fed into testing computers. If it is suspected that any inmate may have discovered any of the programs passwords, it should be changed immediately. The program makes it very easy to do this. It is suggested that all the passwords to all computers in any one lab be changed at the same time to the same password. Historically, having different passwords to different computers in the same lab has caused much confusion. A variety of passwords are difficult to remember and if written down, they are subject to theft.

This program should not be operated on computers that inmates may be using for other purposes. If the program is intended for front line use on a prison unit, it should be operated independently, on its own computer. Having no other programs residing on testing machines, and the machines physically secured from the inmate population during shut down is the most secure mode of operation.

Other extremely important security aspects associated with operator responsibilities are can be found at the end of this booklet. It is critical that operators read and understand these requirements prior to attempting to initialize this program. The single most important aspect of a successful computer assisted testing program is the human operator. If you operate your system strictly in accordance with the guidelines in this manual, you should be able to establish a large scale, inexpensive, valid, trouble free program validation system in short order. Violation of any one of the stated operating tenants found within this manual however, puts your program at risk.

GETTING YOUR PROGRAM STARTED

Unless this program is being set up for you, it is likely that you will have, or your provider will have a set of installation disks. Before the installation is made, the user needs to know if he will be using these programs on a computer that contains no other programs than an operating system.

Where this testing software will be used alone on a computer with no other software present other than the operating system, the computer is known as a "dedicated" machine. The computer is dedicated to the running of the testing software and nothing else. This is the safest and most secure type of installation.

TO START THE PROGRAM INSTALLATION:

Although your testing program will run from the floppy drive it will run very slowly and have none of the security features associated with hard drive operation. You should NEVER OPERATE YOUR TESTING PROGRAM FROM THE FLOPPY DRIVE. It would take little for an inmate to remove your floppy disk from the drive and walk off with it. Such a theft would put your entire program at risk.

THIS TESTING PROGRAM SHOULD BE INSTALLED ON YOUR HARD DISK AND OPERATED FROM THE HARD DISK

GETTING STARTED

OPERATING YOUR PROGRAM

The first rule in bringing up your testing system is to make sure that no inmates are around the computers as you feed in the passwords. They should not be in the area while the computers are running up. Your testing software will be vulnerable to break in from the keyboard during this time. Inmates or students should not be permitted around your systems until the computer itself is asking for the inmate's name. All passwords should be fed into the computer prior to the first student entering the testing lab. Inmates should never be permitted to bring up the system. Unfortunately this has happened all too often and has always resulted in prisoners taking over the system. Once this happens, inmates are being reinforced for criminal behavior on a massive

scale and are proving to themselves that they can beat the system. The staff lab manager is the single most critical aspect of keeping these systems secure. The systems should always be run by volunteers. Years of experience have shown that if staff are ordered to run any program and are not interested in doing so, they will allow inmates to manage such programs to the greatest extent possible. It will only be a matter of time before inmates are running programs as businesses for their own profit, especially if any credit is given for completion of a program. Testing computers should only be brought up by staff. Inmates should never be permitted into the testing room until all passwords are negotiated and the system is up and locked into the testing sequence.

THE SYSTEM LOCK:

Once your program begins initializing, a series of programs, many of which are not noticeable will activate. The function of the system lock is to as quickly as possible block anyone from getting into the computer. The encrypted system lock password is stored in a form that cannot be utilized by a hacker. It is stored in an encrypted form. The system lock password appears similar to that below. It is a single line of text that will appear at the top of your screen and will read:

Enter Password (14 Char. Max, Case Sensitive.)

Note: This password will initially be provided for you. You can change it at any time to one you can more easily remember. The appearance of this line means that syslock is in place, guarding your computer against intruders. Syslock is also giving you information. It is telling you that it will not encrypt a password larger than 14 characters. You need to keep your passwords below 14 characters of letters or numbers.

It is also telling you that your password is Case Sensitive. This means that the program will remember if you put in capital or small letters into your password. For the password to work, you must use the same combination of large and small letters you feed in at this point. In other words, passwords are case sensitive. For example, if you programmed syslock to use the password BisFlick, you must type in that same password combination of small and capital letters. If you forgot and used Bisflick, or bisflick, or BISFLICK, or bisflick, you would not be permitted entry to the computer. The system would keep asking you for a password. Making syslock case sensitive allows for literally thousands of different passwords and markedly increases the security function of this feature.

Once you see "Enter Password:" you are free to type in your system entry password. The password you type will not be displayed on the computer monitor. All the user will

see are little *** (astrics) to mark the place of each letter or number of the password. These little stars will appear immediately under the Enter Password message. Once the password is entered, hit the enter key on your keyboard.

If you enter a wrong password, you will be flashed a message saying:

Password Incorrect

2 tries left before Security Lockout

To prevent inmates from attempting to enter the computer through endlessly feeding in what they believe may be the correct password, if the password is entered incorrectly three times; the system goes into Security Lockout mode. It is programmed to see three wrong tries as a hacker trying to break into the machine. After the third wrong try, the system locks itself down. And across the entire screen in large letters will read the word:

LOCKOUT

At the same time, a siren like sound will go off alerting staff in the area of a possible computer break-in attempt. The siren will not stop unless the computer is turned off. As long as a first correct password is not made available, the system will continue to sound the break in warning and lock itself up every time an intruder tries to break in. There is no other way to turn off the alarm other than by turning off the computer. The system is useless against incompetent staff members who have given inmates these passwords.

THE ANTI-SOFT BOOT SYSTEM:

By the time system lock is negotiated, another security system has already been activated. This is the anti-boot system. This program blocks a soft boot from the keyboard. In other words, the user cannot use the Control + Alt + Del keys to break out of the program and into the operating system. These reboot commands are overridden by the antibooting system. If a soft boot were permitted, the hacker would have a potential window to break into both your system and your program.

Every security feature added to this program was proven to be necessary in the duel between program developers and inmates. There is not one that was added because it was thought to be necessary. The Anti soft boot block comes into operation automatically, and will not even be noticeable unless the user attempts to break out of the system using a soft boot. The only message you will actually see on the screen to

this point is one pointing out that you have successfully negotiated the system lock or:

Security Password Validated . . . Please Continue

Everything else is happening quietly behind the scenes.

The next messages you see on your screen will depend on the model of your computer and how new it is. This system was designed to operate on the older 8088 computers which are now obsolete but served very well as testing computers. There are not many computers this old still in operation. However, should you find computers this old in a used computer shop, note that many of the older 8088 computers do not come with a battery operated clock which notifies the computer or the user as to the time and date. Since the CAP program relies on this information to time and date stamp each test an inmate takes, this information must be fed into your computers operating system. With every test administered, the testing system will be looking for this time and date to date-stamp each test. If this date and time is not available, the software will date every test administered as having been administered in 1980. Keeping track of the time elapsed between tests is an important function of this testing system. If you are using an older computer without a clock, the date and time must be typed in. The input lines where the date and time are to be typed on the older computers look like this:

```
C:\>date  
Current date is Tues 01-01-1980  
Enter new date (mm-dd-yy):_ (type in date here!)
```

Once the date line is filled out on an older computer, it should appear like this:

```
C:\>date  
Current date is Tues 01-01-1980  
Enter new date (mm-dd-yy):08-27-96
```

Once the date line is filled out, you can then hit enter. The computer will flash up a series of very quick messages as additional small security programs are initiated and you will be passed to a main menu.

If you are using a later model computer with a clock, the program will automatically get the date and time from the computer's small battery powered internal clock. The computer will then automatically pass you on to your main menu where you will be free to select the testing program of your choice.

Remember, on the older model computers, it is critical that you give the computer a current date. The testing software which you will set into operation later uses the date

to date stamp each test. The testing software feeds each completed test into a data bank with the date it was taken. If you do not type in the correct date when running very old computers at the spot above, when you look at your data bank for each individual, each students test will be date stamped 01-01-80. This will give you no clue as to when the inmate took the program test or how fast they are progressing through your programmed material. Remember that your testing software will be constantly interacting with the date that your machine gives it, and if you are using very old computers, it is important that date information be given to the computer.

Once the correct date is fed into the computer and the enter key is hit, a main menu will appear which roughly looks like the following:

MAIN MENU

F1 ALCOHOL EDUCATION PROGRAM	F2 ANGER MANAGEMENT PROGRAM
F3 PARENTING EDUCATION PROGRAM	F4 SUBSTANCE ABUSE
F5 AIDS EDUCATION PROGRAM	F6 HOW TO WRITE A RESUME
F7 SEX OFFENDER EDUCATION	F8 CO-DEPENDENCY PROGRAM
F9 PSYCHOLOGICAL DAMAGE TO VICTIMS	F10 EXIT TO DOS

With this simple menu, all the program manager need do is push any one of the F keys to bring up the associated program to be run.

Newer versions of this menu may not have Function Key codes, you will only have to select the program and hit enter. The Menu can be programmed to run as many programs as needed. Once the initial menu has filled, the last slot on the menu can be utilized to bring up a second full menu. This menu can also be programmed full of different programs.

Each selection on this menu stands for a separate, individual testing program. It can be seen from the above illustration that each menu can contain as many as nine separate programs. This may differ on newer menus, but the function is basically the same.

The last key of the menu, the F10 key is reserved for the program manager to go into the computers operating system or exit the program. All other functions of the computer

are locked down, and the F10 key is the only legitimate way out of the testing program other than cutting power to the computer. If this is done, the program tests WILL NOT be stored in encrypted form and will be subject to inmate theft. The testing software must be shut down according to protocol if the tests are to be stored in a safe, encrypted form. The F10 option is password protected also with an encrypted password so that inmates cannot exit the computer or move to other tests from the keyboard. Again, this security option does not protect anything if inmates are given the passwords to these systems. This has happened all too many times in the past. Such incompetent action on the part of prison staff renders these systems worse than useless, they become destructive to the extent that inmates take them and turn them into gangster run businesses. **DO NOT LET INMATES RUN THESE SYSTEMS!**

Pressing any one of the menu keys or on the newer menus, clicking on a menu option with your mouse will bring up the listed program. You will then be required to feed in a PROGRAM PASSWORD. This password is different than the system password that protects the entire computer system referred to as the system lock password. In addition to the system lock password, each of the many individual programs is also protected by its own individual program password. The user cannot activate any of the listed programs without knowing the encrypted individual program password for each option.

The entry passwords guarding each individual program listed on the menu are also changeable at any time. Again we recommend that they all be changed to the same thing and all be changed at the same time. If at any time it is suspected that any of the programs may have been compromised they should all be changed immediately. The passwords should be changed on a regular bases anyway, every couple of months. Inmates have been known to work in teams to watch lab managers punch in passwords. This is the reason why no inmates, not even inmate clerks charged with handing out study materials should be allowed into a lab before the programs are all brought up and initialized.

The main menu can be viewed as the main repository for the testing programs you have available. On any computer containing a number of testing program, the program operator can switch out of any single program to another at any time by exiting the testing program they are in, shutting it down, and then selecting another program for testing. However, it is again highly recommended that if you are working with inmates, that you not be changing programs and punching passwords in their presence. Any time you are feeding in passwords with inmates in the area, your system is at risk.

A much more secure method of running several different testing programs at once is to have a lab containing several computers. Given a lab of six old computers, two can be programmed to test over Alcohol Education, two over Drug Education and two over parenting on any particular day. The next day they can be set to test over different programs. Once a program is initiated, keep it up and operational until all inmates are out of the testing room. Passwords for any program should not be fed into the computer

while inmates are in the area.

Once an inmate starts testing over a program on a particular computer, make sure he remains with that particular computer until he completes that particular program. For example, if you have a six computer lab, each of those computers will be capable of testing over, say, the Alcohol Education Program. A particular inmate can go to any of them and take his next test. However, if this is permitted, his test scores will be scattered over six different computers and it will be very difficult to gather all his test scores together to see if and when he finished his program. Inmates will take advantage of any confusion in program operation to suit their own ends. Make sure that once an inmate starts testing over a particular program on a specific computer that he stays on that particular computer until he finishes that program. It is suggested that each of the testing computers be numbered so inmates know which computer they are supposed to return to in order to finish their program.

Keep the same inmate on the same computer throughout the testing of a particular program.

Some inmates are bright enough to be involved in two different programs at the same time. If they take Alcohol Education on computer six, they need to remain on computer six until they are done testing over that program. That does not preclude them from moving over to computer two to test for Drug Education, but they must remain on computer two for the entire drug education program.

THE USER PROGRAMMABLE TESTING AND SCORING SYSTEM

Once one of the main menu options is selected, the User Programmable Testing and Scoring System will be initiated. The first screen you will see will be the specific program screen for the particular testing program you requested from the main menu. On this initial screen will appear basic information about the name of the institution this system belongs to, basic copyright information, and at the bottom, a box enclosed instruction which informs you to push any key to proceed into the program. If this is the free version offered through the Patrick Web site, it may state the program belongs to Singapore Juvenile Corrections.

This is not an issue as Singapore was the last prison this system this system was commercially prepared for. It is merely a convenience, to use this particular program as we know it is functioning perfectly and ready to go. The system is not copyrighted to Singapore, they were sent copies.

MOVING ON TO THE INDIVIDUAL PROGRAM PASSWORD

Once you have negotiated the System Password, you will now be confronted with the password for each individual program. Let's take a look at what it all looks like as it flashes up on your computer screen. Once you have negotiated the System Password, and chosen an individual program to run, the screen you see come up before you will appear similar to the one below.

**** TASMANIAN DEPARTMENT OF CORRECTIONS ****

DIVISION OF COMPUTER ASSISTED PROGRAMMING

(c) COPYRIGHT MARZINSKY, 1991, 1994. ALL RIGHTS RESERVED

**No portion of this program may be reproduced in any form
without the express written consent of the author.**

Encrypt 1.1 Copyright 1993 Jeff Prosie

Press any key to continue

Once you hit any key, you will be faced with the individual PROGRAM PASSWORD. This is the password that protects each individual program.

You will be given the option to change it each time it comes up and can only do so by

feeding in the correct original password. We recommend that you change all your computers to the same password at the same time, and make sure you do it out of the sight of inmates.

THE PROGRAM PASSWORD SYSTEM:

In the programming of passwords, do not use names of family members or any word that is in the dictionary. You may want to tie the number of the month onto the password or some such predictable event so that it is changed every month. For example, for the month of July, you may want to use something like PETRA07, with the 07 representing the month of the year.

Once you see the opening screen of the program (a screen similar to that illustrated above), push any key, you will be given a new program page. This page will ask for the individual PROGRAM PASSWORD, which protects each individual program. This password is different from the Syslock password, which you have already negotiated. This password, like the system password is also encrypted.

WARNING: In most prison systems it is against policy for inmates to have computer disks in their possession. However, this does not mean they don't. It may be possible to circumvent both the program password and the system lock passwords by booting the computer with a separate DOS or operating system disk. An advanced inmate computer hacker had tried this.

System passwords are tied into and protect the hard disk operating system that resides on the computer from intrusion. These passwords are helpless to protect your computer if it is circumvented from outside by replacing your computers operating system with a separate imported one contained on a floppy disk. This does not mean that this system is totally defenseless should this happen, but the replacement of the systems disk operating system by an inmate constitutes a serious intrusion.

We did not discount the possibility of this happening. What this hacker discovered was that although he was able to boot the testing computers, they were shut down properly and all tests were stored in encrypted form. He was not able to steal them or sell them on the prison yard.

For the most part, inmate hackers would find it too dangerous to bring down this program. They would be mainly interested in obtaining and selling the tests. A hacker who manages to boot up with a separate operating system would be able to damage your software. He would be able to enter your computer and destroy your autoexec.bat and your system files. In the scheme of things, this is really no big deal, as they can be replaced and nothing critical is lost.

On the remote chance such a hacker does get into, and bring down the program, such a hacker would really be doing the system a service. If security around these systems is so slack as to not be able to catch an inmate putting a separate DOS disk in one of these computers and rebooting the system, security is much to slack and the system needs to be brought down and staff retrained as to the proper method of monitoring these labs. DOS disk or no DOS disk, the part of the system the inmates would want, the tests are well protected from theft or intrusion by the machine language encryption program included with this system. If the encrypted tests were stolen, that would be a big deal. Because the encryption device stores all tests in a scrambled form, the theft of tests from this system is not an issue. A hacker will only get scrambled tests.

It does not take much to monitor one of these systems, not much at all, but the little it does take must be there or these systems should not be set into operation. Any lab manager worth his salt should be able to catch an inmate slipping a floppy disk into a computer and shutting it off.

To preclude inmates messing around with testing machines, what we did at many of our institutions was to disconnect the floppy drives once the program was installed. This made it impossible to put anything onto or take anything off of one of the testing machines.

With that warning in place, let's get back to the operation of the software. Once the system operator is past the initial System Password, he or she will then be free to choose a particular program from the Main Menu for testing. The Main Menu is the one that has F1 through F10 as program choices. Newer menus only have numbers and the name of the program. Once one of the program options is chosen, the system will run up to the Program Password screen. The Program Password screen will appear as follows:

```
*****  
* PASSWORD VERIFICATION*  
*****
```

All personnel other than the Program Manager
should leave the immediate area while security
passwords are entered for program access.

PASSWORD: _

The blinking cursor will remain at the end of password looking for a program entry password.

As with the system lock password, the letters of your Program Password will not appear on the screen, instead you will see a series of stars appear on the screen. Each star (*) represents a letter of the password. The letters themselves will not appear. Unlike the System Lock password, the program password is not case sensitive and the system will recognize this password whether it is typed in capital or small letters.

Your Program Password also differs from your System Lock Password in that it will permit only three wrong passwords prior to sounding a five second alarm. Once the alarm has gone off, it will reset itself for another three tries before the alarm again sounds. It will continue to do this indefinitely, sounding an alarm on the third try of each set. Unlike the System Password, the program password will not lock up the system; it will merely continue to sound an alarm that should be sufficient to warn any competent lab manager that something is amiss.

If a program password is typed incorrectly the first time the computer will flash a message "**Wrong Password, Try Again.**" It will do the same the second time a wrong password is typed in. However, on the third time, the alarm will go off, continuing to sound for about six seconds. Again, any time the system operator hears an alarm sound on one of these systems, it calls for an investigation.

The system operator will find that this system will sound different audio alarms when it suspects something is wrong. Anytime one of your testing computers is making noise, the person on the keyboard should be investigated. These alarms are loud enough that they can be heard throughout a rather large testing lab and even above the commotion of a busy lab. Each alarm is either warning the program manager of a potential security breach or notifying the lab manager that he or she needs to take some kind of corrective action with regard to clearing a machine and keeping the traffic moving through the lab. Any alarm the system sounds should never be ignored. The most common alarm is given when an inmate is finished testing. The lab manager should move him off the machine at this point or permit him to take another test. Each testing computer will sound a brief alarm when an inmate is done testing. This is to keep inmates from signing back on again and looking at the next test.

If the program manager knows his Program Password and types it into the computer correctly he will next be confronted with a screen which looks similar to the following:

```
*****  
* PASSWORD VERIFICATION*  
*****
```

All personnel other than the Program Manager should leave the immediate area while security passwords are entered for program access.

PASSWORD: XXXXX

If the password is correct, after the enter key is depressed, the program will give you the option to change the password. You will be given the option to make a password change each time you sign onto any testing program. The password change option will appear similar to this:

Password: *****

Do you want to change the password? y\n

1. WHAT HAPPENS IF YOU FORGET OR LOSE YOUR PASSWORD? If you forget your password you are screwed. You will not be able to access your program. The password was designed so it could not be broken. The chances that even a trained computer programmer would be able to crack this password are slim. If you loose or forget your password, the program is going to work just as hard to keep you out as it does any inmate.

There are a couple of ways you can avoid getting caught in this unpleasant situation. Upon program installation, make a copy of the pass_chg.dat file that first comes with your program. Store it in a safe place. This is the password program component to the testing system. If you forget your password, make a mistake on entering it and then cannot duplicate that mistake, or if the password program gets corrupted for some reason, just replace it with this known good copy. Enter the password associated with it, and your program should fire back up.

If you have more than one testing computer operating at your complex, and you know the password to one of those operating programs, you can unhide the pass_chg.dat file, copy it onto a floppy and replace the lost or malfunctioning file. Then re-hide the file once it is replaced. Keep in mind however that a computer literate inmate would be able to do the same thing if he had these instructions and were able to copy any

password from any working testing computer.

In order to accomplish this however, it would be necessary for a hacker to have two things, a separate DOS disk, and knowledge of what the password was for one of the other working computers. It is the program manager's responsibility to make sure that an inmate never has access to the computer lab except under monitored conditions. Should it become necessary to carry out any of the above described operations, they should not be done by an inmate. Every single computer lab that has ever been broken into by inmates has been the direct result of incompetent, lazy or careless staff not following operational protocol for these systems.

ONCE YOU HAVE NEGOTIATED THE PROGRAM PASSWORD CORRECTLY:

Once you have negotiated the password correctly, the screen will appear as follows:

```
*****  
* PASSWORD VERIFICATION *  
*****
```

All personnel other than the Program Manager should leave the immediate area while security passwords are entered for program access.

PASSWORD: XXXXXX

Do you want to change the password?

CHANGING YOUR PROGRAM PASSWORD:

USE CAUTION IN MAKING ANY PASSWORD CHANGES:

If you elect to change your password, be extremely careful and deliberate in making your keystrokes. There have been several instances where the program

operator struck an unintended key by mistake. Having forgotten which keys he used in making the mistake and trying to duplicate it again caused a lockout. The program became useless.

As you type in a new password, be careful to hit the correct keys. Make an effort to remember which keys you hit. If you make a mistake, type in the same exact mistake into the password section. When the program allows you to change your password again, make the change you first intended. The system is very unforgiving if you make a mistake and do not realize it or remember it.

When you first receive your system, it will be preprogrammed with both a System Lock Password and passwords that protect each individual program. We recommend that you use these to get into your program and experiment with its different features. However, as soon as you are comfortable, change the password to something personal to you that you will remember.

The actual changing of your password is very easy. As seen in the above example, every time you enter a correct password, the program will ask you if you want to change that password (see above for an example). To change your password, just answer Yes to the statement: **Do you want to change the password? (Y/N)**. Push either Y for yes, or N for no.

If you choose no, the program will admit you based on the correct password that you previously typed in. On the other hand, if you choose, Y or yes, the computer will know that you mean to change your current password. You will then be provided with the following screen which will permit you to change your password.

**** Password must be between 5 and 8 characters ****

Enter New Password:

This screen is informing you of your password parameters. Anything below five characters does not provide the number of different combinations of letters and numbers to allow for the kind of security we are looking for while operating within correctional environments.

The top line indicates that the maximum number of characters that you are permitted in formulating your password is eight. Once your new password is entered, this information will be encrypted within the password program and rendered into a code which cannot be pulled up by a hacker or viewed by asking DOS to type out the

password file.

BE VERY CAREFUL TO ENTER THIS NEW PASSWORD SLOWLY AND CAUTIOUSLY, A TYPING MISTAKE OR HITTING A WRONG KEY HERE AFTER ENTERING A NEW PASSWORD COULD RENDER YOUR PROGRAM USELESS IF YOU CANNOT REPRODUCE EXACTLY ANY MISTAKE YOU MAKE. TO MAKE THINGS EVEN MORE DIFFICULT, THE PROGRAM WILL NOT ALLOW YOU TO BACK SPACE AND RETYPE IF YOU MAKE A MISTAKE HERE!

Keep in mind that if you are lax about security when entering your passwords, and an inmate looks over your shoulder and attains your passwords, he or she could enter the program with your passwords and change these denying YOU access to your own program.

Access to the test compilation section or the Edit Test Question Section of the program is secured by a second independent password. This makes a third encrypted password necessary to enter this level of the program.

Unless you are developing your own tests or want to change the tests that are already there, you will not need to enter this portion of the program.

THERE SHOULD BE NO INMATES ANYWHERE IN THE AREA WHEN YOU ARE STARTING PROGRAM TESTING OPERATIONS AND ENTERING ANY OF YOUR PASSWORDS.

With that warning given, let's get back to the process for changing your password. The cursor will be waiting after the work Password below for your new password. Once you enter a new password and hit the enter key, the computer will respond with the following message:

**** Password must be between 5 and 8 characters ****

Enter New Password:

(Enter)

Password updated!

This means that your new password has been accepted by the computer and will be

used by the computer from now on as your program entry password. There is no limit to how many times you can change your password. **Your Program Password will not respond to numbers, only letters.** It will mark an X in place of every letter you feed into place for your password.

WHAT HAPPENS IF YOU KNOW THE RIGHT PROGRAM PASSWORD BUT MAKE A MISTAKE AND TYPE THE WRONG PASSWORD INTO THE COMPUTER?

It is assumed that the authorized user of your equipment will not need several tries to get his or her password correct. But just in case you are having a bad hair day, the program is fairly lenient as long as sooner or later you can come up with the correct program password. The program will give the user three separate tries to get the password right. If the password is typed in incorrectly on any of these first three tries, the computer will respond with:

Wrong Password. Try again!

It will repeat this message three times. Upon the third failed try the following message will appear:

Access Denied

At the same time, the warning alarm will go off, sounding for approximately six seconds. This cycle will be repeated indefinitely, sounding the alarm each three failed tries occur until the password is negotiated successfully or the machine is turned off.

INITIATING A TESTING PROGRAM:

Once you negotiate the Program Password, you are ready to fire up your testing program.

Let's review what you did to get to this point.

1. You would have had to successfully negotiate the System Lock. This is the initial password that permits you to use a dedicated computer at all. Without getting this password right, you would not be permitted onto the computer.
2. If your computer is ancient, you would have had to feed in the correct date and time.

Most computers this old are already in the junk yard. Most likely you have a computer with a clock that keeps track of the time and date. If you had an older model, you would then have to put the correct date and time into the computer. If your computer does not provide the testing system with a date, it cannot stamp the date of each test taken onto the testing record. With an ancient computer, if you don't feed in the time and date, you will have a bunch of records that all read some time in 1981.

3. Once you negotiated the **SYSTEM PASSWORD**, you will be passed on into the main program menu. This is the menu that contains the entire collection of all the different testing programs provided by a particular computer. The selections on this menu are triggered by pushing the F1 - F10 keys to bring up the specific program you would like to test or with the newer menu, just click on the name of the program.

4. Once you chose a program for the computer to test from the MAIN MENU, you will then be challenged by the **PROGRAM PASSWORD** for that particular program. Once you negotiate that you will be given the opportunity to change the program password if you so desire.

5. If you choose N or No and elected not to change your current program password the test encryption program will automatically initialize and unencrypt all of the program tests out of coded storage so they can be read and presented for the test. Remember, for the tests to be safely encrypted for storage while the computer is in an inactive state, the program must be shut down through exit which will allow the system to encrypt the tests before shutting down. If the computer is merely switched off, the tests will not be encrypted and can be stolen.

THE TEST ENCRYPTION PROGRAM:

Once the Program Password is punched in and the enter key pushed, the last phase of program initialization will begin. The system goes into the encrypted storage bank where the tests are kept and one by one unencrypts them from an unreadable, unintelligible code, back into English. Your tests are stored in encryption code to keep hackers from breaking into your computer while they are shut and stealing your tests. Once you put in the correct password and hit enter, you will see the computer un-encrypting your tests. While un-encrypting, the computer screen will begin to flash a series of messages which appear as follows:

Test1.dat was successfully encrypted.

Test2.dat was successfully encrypted.

Test3.dat was successfully encrypted.

Etc.: to the maximum number of tests that have been programmed into that particular testing program.

Upon appropriate shutdown, you will see the same series of encryption messages as above as each test is re-encrypted for long term storage before computer shut down.

Test1.dat was successfully encrypted.

Test2.dat was successfully encrypted.

Once encrypted for secure storage the tests are unreadable. Without this encryption device, a hacker would be able to boot your computer with a separate operating system disk and ask DOS to print out the test questions which the machine would then dutifully do. The hacker could then view your test questions and manually write them out if he had enough time. If the computer were hooked to a printer, there would be nothing to stop a smart hacker from printing out the test questions. Your test answers however are buried beneath another layer of security requiring yet again another encrypted password to access.

Like they did with paper and pencil tests, if a hacker could get hold of your test questions, especially if credit was given for test completion, he would be in an enviable position with a commodity that he could sell on the prison yard for a large profit. Inmates would be reinforced on a large scale for cheating and beating the system. This is exactly what happened when we tried to test large scale programs to measure what inmates had learned using paper and pencil tests. Within six months they had in one way or another stolen most of the tests and were selling them on the prison yard for store commodities.

The motivation for a testing computer break-in is always present as long as any kind of credit at all is being given for the completion of Computer Assisted Programs. This is a constant and fact that we repeatedly learned about the hard way. When hard copy tests were utilized for testing, inmate theft of paper and pencil tests were a fact of life. Any Tom, Dick or Harry could walk off with a hard copy of a written test by creating a distraction. In six months, the inmate population had control of virtually all written tests and they were being sold on a large scale on and between prison yards.

With the inception of Computer Assisted testing, the common thief had been eliminated. Now we were faced with a very small number of computer literate inmates who might pose a potential threat to this system. However, these specialized inmates could be prevailed upon by prison gangsters to attempt to break into computerized testing systems. They proved a significant threat to this system and the validity of testing until the implementation of the program encryption system which also eliminated them as a threat IF and only IF staff program managers monitored their labs during inmate usage.

Since good test questions are difficult and time consuming to formulate, it only makes sense to have a system in place to protect them from hackers or other unauthorized

computer users who might render your system invalid in a matter of minutes if they were to obtain your tests. The encryption device keeps your test questions secure, but does so only if your labs are monitored by staff during operation and locked up during shut down. .

The encryption program has its own password. Most users will not need or be provided with this key. To unlock the encryption program requires a key program which should never be kept on a testing computer to unlock it. If a hacker attempts to manipulate the password for the encryption program and does not get it absolutely right on the first try, the system freezes and becomes unusable. The bottom line is that the system will commit suicide before giving up the test questions to an unauthorized source.

Because it is important that the encryption system complete its job prior to shutting down your computer, it is necessary that you always use the QUIT option on your main menu rather than just flipping the switch on the computer to cut off the power in order to exit your testing program.

Once you feed in the correct program password, the encryption program will initiate and descramble all of your program tests for testing. The system will then pass you on to a screen which contains the name of the testing program that you requested. In the example below, the Drug Education Program was requested, and the name screen for that program is showing.

ONCE THE CORRECT PASSWORD IS ENTERED INTO THE COMPUTER:

The program will print across the top of the screen the name of the specific program it had been asked to test over. This name screen will appear similar to the following:

DRUG EDUCATION PROGRAM

Identification Number:

When this screen comes up for the first time, it will request of the student an identification number must be at least five digits in length. If the user attempts to type in an ID number smaller than five digits, the computer will respond with the following message:

Identification number must be at least five digits. Try again:

If the user has typed in less than five digits, the program will again ask the user for an identification number. ***The computer will recognize users based on their numbers only. The number cannot contain any dashes. The identification number must be at least five characters long.*** The number can be a social security number or an inmate number or any other number greater than five digits. The maximum number of digits which can be entered for an identification number is eleven. The numbers we recommend be used is the prisoner's inmate number. Although there are many names which are the same, there should not be any matching inmate numbers.

ONCE THE STUDENT IDENTIFICATION NUMBER IS ENTERED INTO THE COMPUTER the program will automatically begin a search through its memory banks to try and find a student with matching number already on record. If the student is signing on for the first time, the program will not find a match and will begin asking a series of questions to start a testing record for a new student. The system keeps a data which can consist of thousands of students. Each time a student signs on; the program will search its data bank to find the student. Once a student begins testing on a particular computer, they should finish all program tests on that computer or they will have test records spread through out an entire lab.

If you are a new user signing into a program for the first time the computer will need to start a new test file on you to keep track of all your test scores. The initial screen will appear exactly like the one in the figure above. ***When you are signing in for the first time, you can change this identification number by using the backspace and retyping the number over again prior to pushing the enter key. However, once you push the enter key, the identification number becomes the permanent identification number the computer will use to identify you. Any changes or corrections need to be made before the enter key is pushed.***

If the student, by mistake or otherwise, is permitted to take a test with the wrong identification number and he passes that test, the next time he signs on with his correct ID number the previous test he signed in with the wrong ID number will not appear on his personalized test record data bank. It will be stored in a different test file, under the wrong ID number he previously typed in. The computer only recognizes individuals by their ID numbers, not their names.

If the student types in the wrong ID number and recognizes that he made a mistake after hitting the enter button, the best thing for the system manager to do to get him or her back on track again is to just blow through the test pushing all A's for each question just to get through the test. After this test is failed, when the next test is presented, have the individual sign onto the computer again with his correct ID number and let him take the test under his correct number. The bogus test which you failed on purpose, with the wrong user ID, can be erased later from the main program menu.

If the user is signing on for the first time, the series of information gathering screens he will be confronted with is different from a user who has already signed onto the system.

The system will request the following information of a new user:

NAME OF PROGRAM

(Eg. ALCOHOL EDUCATION)

Identification Number: 12345 <enter>

Last Name: Smith <enter>

First Name: Harvey <enter>

Housing Number: 1B46 <enter>

THE PERSONALIZED TEST RECORD DATA FILE:

Once the above information is given to the computer on a new user, the program takes that information and starts a new personalized test file data bank for a new student. Right after the Housing Number information is filled in, as soon as the enter key is hit, the program automatically creates a personalized test record file. Across the top of the page of this new file is listed the name of the program to which the test record belongs. Within the file is the name of the inmate, his number and his housing unit. Each testing program has the potential of administering twenty tests over any particular program. Consequently, each personalized data record has space to record as many as 20 tests.

In the early days when we were using paper and pencil tests, inmates continually demanded instantaneous feedback as to their test score. We found that there was more program participation when such feedback was provided. However, scoring tests manually was very time consuming and problematic. Inmates nagged program manager to the verge of insanity with regard to their test scores, which they always seemed to forget. To resolve this issue, the computer displays the students personalized test history each time the student signs on for a test. Giving the student his personal history up front saves staff much aggravation and helps prevent burnout.

In part due to the constant haggling with inmates over what their test scores were, Program managers who dealt with manually tested programs would refuse such assignments due to having to manually score thousands of inmate tests by hand, and then have to leaf through massive record books to answer the requests of endless thousands of inmates about their test score history. This became a thankless, boring job and proved to be a major factor contributing to staff burn out.

To make sure that the student knows the results of his most recent test, at the end of the automated test, the student's test record is displayed to him once again, immediately upon test completion. The student is given instantaneous feedback as to his most recent test result. The system manager also knows immediately whether to pass him on to the next programmed section, or give him back his previous study material to study harder and retake the test again. This proved to be a vast improvement over expensive, time-consuming paper and pencil scored testing where inmates harassed staff relentlessly demanding their test scores and arguing and complaining bitterly that this or that was an unfair question which they should have been given credit for. If staff refused to give them credit, often they would up the intensity of their demands taking an emotional toll on the program manager and necessitating that the program supervisor expend the necessary amount of time and energy to deal with such a hostile situation. Also, when tests were scored manually, if staff refused to take the time to look up past scores for inmates, many of them would become hostile and potentially abusive. It cost them nothing to attempt to manipulate a higher grade out of a program manager scoring paper and pencil tests by insisting that this or that test question was not fair, was poorly written or that their answer was the truly right one and the one the test designer picked technically wrong.

To eliminate this kind of ceaseless inmate behavior and save wear and tear on the nerves of the System Manager, this testing program **INTENTIONALLY** does not provide feedback to the inmate as to which test questions he missed. It only tells him/her the score they earned on a particular test and whether or not they made a passing grade. A passing grade is set at 70%. The software was programmed to circumvent this argumentative tar pit of inmate behavior. Once this programming was completed, the number of verbal conflicts with staff over program grades and test questions dropped to zero. A major factor in the burnout of program counselors and managers was eliminated.

By showing the inmate twice during each testing period what his test record is, the endless questions as to what inmates had scored in the past, and the massive amount of time required looking up inmates' scores in a big record book were reduced to zero. Since inmates were never provided information as to which specific test questions they missed, they had nothing to argue about. As a result, constant negative inmate interactions with program managers with regard to test scores was totally eliminated as a source of staff burn out.

With that digression under our belt, let's get back on track and return to how the computer makes an individual test record for each student. As we were saying, the computer will formulate a brand new testing record for each student the first time he or she signs onto a testing computer. When a student signs on for the first time the program will ask for a name, number and housing unit. After the housing unit number is entered and the enter key is pushed, the new record is formulated automatically. This will be the student's individualized testing record for the entire program he signed on to

test over. As long as the student keeps signing onto the same computer for a particular program, all of his test scores will be stored on that computer. The computer will keep his testing record updated with each test passed, the date the test was passed and the test score.

When the program is finished creating the individualized test record, a flashing cursor will appear at the bottom of the screen. The computer will be waiting for the user to punch in the number of the specific test he or she would like to take.

The individualized testing record screen the computer creates for a student taking the Drug Education Program would look something like this:

DRUG EDUCATION PROGRAM

NAME: HARVEY SMITH NUMBER: 234567 HOUSING NO. 1B4

Test 1: 00/00/0000 Score: 0	Test 2: 00/00/0000 Score: 0
Test 3: 00/00/0000 Score: 0	Test 4: 00/00/0000 Score: 0
Test 5: 00/00/0000 Score: 0	Test 6: 00/00/0000 Score: 0
Test 7: 00/00/0000 Score: 0	Test 8: 00/00/0000 Score: 0
Test 9: 00/00/0000 Score: 0	Test 10: 00/00/0000 Score: 0
Test 11:00/00/0000 Score: 0	Test 12: 00/00/0000 Score: 0
Test 13:00/00/0000 Score: 0	Test 14: 00/00/0000 Score: 0
Test 15:00/00/0000 Score: 0	Test 16: 00/00/0000 Score: 0
Test 17:00/00/0000 Score: 0	Test 18: 00/00/0000 Score: 0
Test 19:00/00/0000 Score: 0	Test 20: 00/00/0000 Score: 0

LAST TEST TAKEN WAS TEST # ON / 00 /0000

TYPE TEST NUMBER (1-20) AND PRESS ENTER: _

As you can see, the testing record will keep track of up to 20 different tests for any particular program. The test record above is blank. This is what the record of a new

student who had just signed onto a testing computer would look like before any tests are taken. The computer will record only passed tests. The system is set up to use 70% as a passing score. If the student does not attain a 70% or better, the score remains 0. It is possible for a student to take a passed test again in an attempt to get a higher score. The computer will record the latest passing score for the record. It is surprising how many inmates will request to take a test again a week later to bring up their score.

Note the last line in the illustration above. The line reads:

TYPE TEST NUMBER (1-20) AND PRESS ENTER: _.

After the word “enter”, there will be a blinking cursor. The cursor is waiting for the student to enter the NUMBER of the test he or she would like to take. If the student wanted to take test number one, he or she would type in the number 1 at the blinking cursor and hit the enter key.

TYPE TEST NUMBER (1-20) AND PRESS ENTER: 1 <enter>

After pressing the enter key, the number of the test the student requested would appear on the screen and present the first randomly chosen test question. The order of presentation of the test questions is chosen by a random number generator within the program. Test questions are chosen randomly, but the same test question is never administered more than once. It was discovered that if the order of the test questions could in any way be predicted, inmates would find a way to cheat.

Inserting the number of the test and hitting the enter key will bring up that test. The command in the figure above would bring up test number one. From the instant the first question appears, the student is being timed and has fifteen minutes to answer 20 test questions. The program will repeatedly flash the remaining time a student has left to pass the test in the upper left hand part of the screen. Without this feature, inmates, disregarding the line of students waiting to take their tests would sit at a computer significantly longer than necessary. The idea is to move as many inmates through the testing lab as possible in the shortest amount of time. The test time helps accomplish this. If the student blows the fifteen minute limit, the computer locks up and sounds an alarm. In conjunction with a competent program manager, this feature limits the amount of time that any one inmate can spend on a particular computer thus having a secondary security function.

Although the system was designed to be able to administer as many as twenty tests

over any particular topic, the program does not have to have twenty tests programmed in to be viable. The program will administer the number of tests it has programmed into it over any particular topic. For instance the Drug Education Program at the present time has only nine study booklets on various dangerous drugs inmates use. If numbers one through nine were fed into the blinking cursor where a test number is asked for, that test would come up. If the user brought up this program and typed in any of the sections 10 – 20, slots where a test has not been programmed into the system yet, the computer would simply respond:

TEST 20 NOT SETUP. TRY ANOTHER.

Empty test slots can be programmed with test questions by using the Test Edit Section of this program. As such, this system lets you test over the material you currently have without precluding future expansion.

Before any test can be given by the computer, it must be programmed into the software. This can be done easily with the Test Edit section of the program. Each test must contain no less than twenty questions to be functional. We will go into this in more detail later when we get to the section on how to program test questions into this software package.

The previous section covers what the software does and looks like as a new student signs up for testing. What follows describes how the system operates the next time that student comes back having already signed up to test over a particular program. Remember, a student should remain on the same computer while testing over any particular program so all his test scores remain together.

SIGN ON FOR STUDENTS WHO HAVE PREVIOUSLY TAKEN A TEST:

THE PROGRAM WILL AUTOMATICALLY RECOGNIZE STUDENTS WHO HAVE PREVIOUSLY SIGNED IN BY THEIR IDENTIFICATION OR INMATE NUMBER ONLY.

If a student has signed onto the program previously, the next time he signs in, the computer will do a check of its memory banks to see if that student has tested previously. If it finds the user has already passed a test over a particular program, it will notify the user that his or her number already exists.

Once the program recognizes a previous user, it will not ask for additional information in order to expedite testing. This is so even if the inmate failed his first or last test. The software will still recognize the user. Once the system recognizes a user, it will display the following message:

Your number exists in the program
Press any key _

Once any key is pushed, the computer will then bring up the students entire test record for him to view. From that screen, the student will be able to choose the test he would like to take by typing in the desired test number at the blinking cursor and hitting the enter key.

All test questions are randomly selected from the test question bank within the computer, at the same time, the answers to each question are also randomly scrambled and the question is immediately displayed on computer screen. This means that two inmates taking the same test next to each other will not get the same test questions, in the same order and if they do, the answers will not be in the same position. Its randomized chaos. It is impossible for inmates to predict which test questions they will get or the order of the answers. The computer keeps track of this chaos, the inmate cannot.

If the computer is reset and the same test number is asked for a second time, the chances are that a totally different test questions will be displayed for each test question number. The test question which was number one last time will not be the same as test question one this time. Despite the seeming chaos of continually scrambled answers and large numbers of questions being randomly chosen for display, the program constantly tracks and monitors the correct answer and can differentiate it from all other possible choices.

IF A STUDENT BRINGS UP HIS TEST RECORD, BUT DOES NOT FIND A SCORE FOR THE LAST TEST HE TOOK:

One of the most common mistakes attributed to the program by students is blaming the computer for a missing test score. The scenario often goes like this: "I took test #x last week and passed it. I saw my score on the screen and now it isn't there! The computer lost my test!!" They rightly remember studying for, taking a specific test and seeing the score they attained, however when they sign on for their next test session they find their previous test score does not appear on their test record under their name. This has never proved to be a software problem.

This situation is usually due to one of two mistakes made by the student. Number one is that they did not sign onto the same computer they took their previous tests on. Two is that they made a one digit mistake while typing in their identification number. Making

a mistake by typing in an incorrect number in the identification string has the effect of starting a completely new testing record. The inmates test will be scored at completion and placed in this entirely new record, not in his original record. Consequently, his last test score will not appear under his name and correct identification number. It will be missing from his test history transcript.

Often, the student does not realize he made a mistake at the time he typed in his identification number, the computer starts a new record, he fills in the information and he proceeds to take the test. Remember, the system only identifies users by their ID numbers. You could have ten people with the same name taking tests and as long as their numbers are different, the program won't care. It starts records and assigns test scores according to the identification number which is originally fed into the computer when a student first signs on before taking their first test. Their missing test can be found in the computer's test storage bank under the students name with an incorrect identification number.

When a student complains that the computer did not save one of his tests, the missing test, 99.99 percent of the time can be found in the EDIT USER section of the program where student test files are stored. There is no way to transfer the score into the correct test file without the inmate re-taking the test within his correct data file.

THE TEST TIMER FUNCTION:

THE TIMER FUNCTION: A test timer is activated at the same time that the first test question is displayed on the computer monitor. The timer does not permit the student any more than fifteen minutes to finish answering the twenty test questions which the computer will randomly choose.

The test timer/program shut down feature serves two main functions. First of all, it serves as a program security feature which limits inmate access time to both the test questions and the computer in general. If the program manager gets distracted somehow, either by legitimate business or an inmate staged distraction the timer will automatically shut down the program at the end of fifteen minutes and lock up the machine. In most cases, the inmate would not have sufficient time to copy many of the test questions from the screen onto paper before the system shut itself down.

The timer also keeps inmates on track by forcing them to stay within allotted time limits. Inmates are given enough time to take their test and little more. They have very little time left to try to break into the computer, or experiment with the system.

To prevent an inmate from breaking into the program, once it initializes all keys, except

those used to answer the multiple choice test questions are trapped and programmatically made non-functional. In effect, the inmate is trapped within the specific test he or she has chosen to take and cannot gain entry to the operating system or any other part of the testing program.

The timer also serves to economize computer time. Inmates who did not study and are counting on having unlimited time to sit in front of the computer attempting to guess their way through test questions are short circuited. Without some kind of limitation, inmates can sit at a computer for hours denying other, more prepared students access to a testing computer. The timer feature serves to keep the line of inmates waiting to access the computer lab moving briskly. In no uncertain terms, the program timer lets inmates know that they will not have all day to sit at the computer and take a test. They had better have expended the study time and know the material well prior to getting onto a computer or they will either fail, or be kicked off by the computer itself, an action that would create plenty of conflict if it had to be done by a human lab manager.

The test timer also serves to utilize staff time afforded to computer assisted testing efficient and as economical as possible. Correctional staff in today's institutions often are forced to wear many different hats. They are busy and overworked and don't have all day to wait for unprepared inmates to try and guess their way through tests.

The timed test feature also short circuits the wear and tear on the nerves of program managers.

In the past when strict times for testing were imposed by staff to keep some kind of control over how much time they were allotting to testing, manipulative and potentially explosive inmates constantly attempted to manipulate staff for more testing time. "Just give me a few more minutes to finish." It was non-stop. Inmates cannot manipulate or con a computer and more appropriately, adjust their rate of answering test questions to the time they have left. The timed test feature does a very admirable job of eliminating a number of troublesome problems that human operators would otherwise have to face if tests were being administered and graded manually. These problems are unique to inmate populations and are very wearing on the nerves of program managers, often contributing to their early burn out with program administration.

Experience has shown us that the vast majority of inmates can answer twenty test questions in ten minutes. Very few cannot finish taking a twenty minute test in fifteen minutes. There are however, a few students who cannot finish their test within this time limit. When this happens, there's usually one of three things going on. (a) the most common is that the inmate has not studied and thought he could guess his way through a computerized test AND he ignores the timer warnings; (b) The student has a reading problem and takes an abnormally long time to finish his test. He should be referred to adult basic education. (c) The inmate may be mentally ill or severely depressed with a resultant inability to concentrate. Such a student may need to be referred to Mental

Health or Psychiatry. As a general rule however, very few students surpass the fifteen minute time limit necessitating a reset of the computer.

TIMER WARNINGS

The program is designed so that no student will be blindsided by the program timer and an unexpected computer shut down. Nor does the computer ever shut itself down without giving the student fair warning. Periodic warnings informing the student of the amount of time left prior to computer shut down for blowing test time limits are flashed before the students eyes. The student will see the first warning flashing before their eyes after five minutes of testing time has elapsed. The computer will flash in the upper right hand corner of the screen a warning indicating that there are ten minutes remaining. The first time this message appears, the student is made aware that he or she is taking a timed test. Most students will not allow the time limit to elapse prior to completing their testing.

WARNING 10 MINUTES LEFT

The test time warning will next be seen after ten minutes of testing time has elapsed. Once again a silent flashing warning appears in the upper right hand corner of the screen and flashes on and off for several seconds clearly displaying the message:

WARNING 5 MINUTES LEFT

This will happen five minutes before the program shuts down and locks up the keyboard.

The third and last warning is given the student one minute before the program locks up the machine. The upper left hand corner will display **WARNING 1 MINUTE LEFT**.

If the inmate does not complete his test prior to the elapse of the last warning, the computer will lock up and display in the middle of the screen:

ESTABLISHED TIME LIMITS EXCEEDED

At the same time, a warning buzzer will go off, sounding an alarm for several seconds to notify the program manager that there may be a potential security violation on the computer sounding the alarm. Any time one of the testing computers sounds an alarm, the program manager should investigate.

The ESTABLISHED TIME LIMITS EXCEEDED message and the warning alarm will be followed by another message which appears as follows:

ESTABLISHED TIME LIMITS EXCEEDED

PLEASE CONTACT COORDINATOR TO RESET ACCESS

When the Enter Key is hit, the computer will then display:

ENTER ACCESS CODE:

If the enter key is not hit and the computer is allowed to stand without input, the program will revert to only the above message and will remain frozen at that point with Enter Access Code being continually displayed in the middle of the screen. The computer will remain in this state until either an access code is fed into the machine, or the computer is turned off. Rebooting the machine will not work.

TO RESET THE PROGRAM ACCESS CODE: We recommend that prior to resetting your access that the area be cleared of all students or inmates. It would not be good if students saw your access code as the access code is not changeable as is your program password. It remains constant.

TO UNLOCK THIS VERSION OF THE TESTING PROGRAM ONCE OVERTIME LIMITS ARE EXCEEDED, YOU MUST HIT ALT F10 TO RESUME TESTING. IT IS ADVISED THAT YOU CLEAR INMATES FROM YOUR TESTING AREA PRIOR TO DOING THIS.

This procedure will reset the program and move you back into testing mode. An alarm will sound as the machine reverts back to a testing mode once again. Once reset is initiated, the computer monitor will display the record of the inmate or student who exceeded the testing limits.

At the bottom of the screen will be a box reading:

Testing session is completed Press Enter to reset for next test _
--

The program manager is then free to hit the Enter key. The program will then switch to display the name of the program which was being tested and will at the same time request the identification number of the next student. It is assumed that the student who exceeded time limits would have been sent back to continue studying. Once reset, the screen will appear as follows:

ALCOHOL EDUCATION PROGRAM

Identification Number: _

Experience has shown that although very few correctional students will permit programmed time limits to be exceeded, unprepared students will rush through and often fail a test prior to letting time limits lapse.

As mentioned previously, there appear to be three major reasons for students failing tests. At this time, we will take a closer look at these reasons.

(A) THE FIRST MAJOR REASON FOR TEST FAILURE IS THAT THE INMATE DID NOT STUDY AND THINKS HE CAN GUESS HIS WAY THROUGH THE COMPUTERIZED TEST WITH INADEQUATE PREPARATION.

This is the single most prevalent reason inmates may find the fifteen minute time limit to be not enough. However, very few actually break the time limit and cause program shut down. As they near the 1 minute mark, these ill prepared students will quickly push answer keys in order not to be denied at least a guess at the remainder of the questions.

If you have designed your test questions well, the probability of a student guessing their way through any test and passing it will be extremely remote. Chances of students accidentally guessing their way through established test questions is reduced even further due to the program requirement that multiple choice questions be composed of five potential choices instead of the traditional four.

If your test questions are designed poorly with little thought and can be successfully guessed at, your problems will be compounded. When inmates learn they can guess their way through your questions, you will have more inmates showing up to try and they will do just that. If test questions are poorly designed and easy to guess at, many students will not bother to study. With poorly designed, easy to pass test questions, your labs will be flooded with inmates attempting to gain credit by just pushing a few keys and gambling on their ability to come up with the right answer.

If the test questions you compose are well thought out and solid and cannot be passed by guessing, you will find that the heaviest inmate failure rates occur on the first three tests. In designing the test questions you will be given with this version, we actually called inmates into a testing computer and without studying, requested they try to guess their way through new tests. If they could come close, the test questions were redesigned.

Historically, inmates have not had to study for clinical or substance abuse types of programs while in prison. They have not been made responsible for their own rehabilitation. They are fed information in classes with the assumption that they are interested. They have slept their way through groups, often with inexperienced counselors. They have not been made responsible for knowing the program material in a large number of cases. When confronted with a system where they must learn the information related to their recovery or problems, rather than assuming they will have to study, they often "test" the system in search of an easy way to get through it. Only when they find out for themselves that they cannot cheat, con or manipulate their way through the system will they buckle down and begin to study.

For many inmates, taking the path of least resistance is a way of life. As such, many of them will test this system in an attempt to circumvent it before they make any effort to study at all. This is why the first three tests of any program are failed the most. For some reason, it takes three failed tests before they get the message that they must study to get through these programs and they must learn the material. There is no easy way to cheat. Consequently, the failure rate during the first three tests will almost always be significantly higher than for later tests.

When first exposed to this system, system operators can expect inmates to be probing the system to see just how much effort they have to put forth to get by. Many of them will not buckle down and study until they are convinced that there is no easier way to get the credit.

Experience shows us that if they fail the first test, they will usually but not always spend more time studying for it a second time but the passing score will often be low. Generally, they will pass a second test in the series with a score in the 70's on the first try. By the time they reach a third test the typical inmate will then begin to bring his scores up into the 80 range after the second test. .

(B) THE SECOND MAJOR REASON INMATES FAIL TESTS IS THAT THE INMATE HAS INADEQUATE READING SKILLS, OFTEN BELOW THE 6TH GRADE LEVEL AND IS HAVING DIFFICULTY COMPREHENDING THE STUDY MATERIAL.

One study done on several hundred inmates entering a large prison complex in the South West revealed that the average inmate coming into the prison system had a 7th grade educational level. This must be kept in mind when writing study booklets for these programs to test over. Identifying this particular impediment to inmate progress becomes difficult when one considers that many inmates are too proud to admit they have reading problems.

If an inmate appears to be trying hard failing test after test, but does not appear to be mentally ill, then a low reading level is most likely the cause for failure. Many of these inmates can be identified as they try to make it through your programs. We have found that pairing up inmates with lower reading levels with other inmates who are willing to

help them study has proved successful in helping low level readers legitimately make it through tested programs as well as helping them to learn to read.

(C) THE THIRD MOST FREQUENT REASON INMATES FAIL TESTS IS THAT THE INMATE IS MENTALLY ILL, ON, OR OFF PSYCHOTROPIC MEDICATIONS, IS DRUG IMPAIRED OR IS SUFFERING FROM EXTREME STRESS WHICH IMPAIRS MEMORY AND CONCENTRATION.

Since these programs are designed to be operated at very high volumes, they can be used as a rough gauge to screen large numbers of inmates for problems that interfere with memory, concentration and function. ANY inmate who exceeds program test time limits should be questioned to determine the cause. The same is true of inmates who repeatedly fail tests but appear to be trying.

When program time limits are actually exceeded the computer screen will display the following cues:

Time Limits exceeded.

As mentioned previously, an audio warning alarm will go off and the computer will have to be reset.

IF THE STUDENT DOES NOT EXCEED ESTABLISHED TESTING TIME LIMITS:

Once the student answers the 20th or last question, the program will sound a beeping alarm for approximately six seconds to notify the program manager that a student has finished testing and has no further business on that machine. Once the test completion alarm goes off, the inmate should be allowed only enough time to look at his score before proceeding out of the lab to pick up his next study booklet, or retake his present one of he has failed his test.

Prompt action at the completion of testing will foil curious inmates from experimenting with your testing computers. Prior to sounding the alarm however, the computer will instantly grade his test and display test results on the monitor for the student to view. There is no delay or waiting for the student to receive his score, it is provided instantaneously. This is a feature the inmates themselves wanted built into the program.

Purposely we designed this system NOT to provide the student with any specific information as to which test questions were missed or why. This was a design feature. The system was not meant to be a teaching machine for a number of very good reasons. These reasons are covered in the position paper on this program concept. The program was designed to be a very efficient measuring device to assure that a

certain level of proficiency is reached by inmates at minimal cost to the correctional system.

IF A TEST IS FAILED:

If the inmate fails a test, the failed score will be shown him by the system. The computer will report to the student something to this effect:

YOU RECEIVED A 65%. YOU NEEDED A 70% TO PASS

A failed score is never saved onto the inmate testing record in order to save space. However, depending on the version of software you have, a date may appear with no score, indicating that the inmate attempted a particular test and failed on that date. If the student fails his test, he is given his study booklet back, requested to study more and attempt to test again next week. We strongly recommend that the inmate NOT be permitted to take a test again the same day. Experience has shown that in most cases he nag the program manager half to death to be allowed to take the same test toward the end of the testing session and will again fail.

If the student earns a score of 90% or better, the computer will display a message of congratulation on the screen saying something to the effect of:

CONGRATULATIONS, YOU SCORED 90%, GOOD JOB!

At the bottom of this same testing record screen will appear a small box, spanning the length of the screen. Inside the box will appear the following message:

**Testing session is completed
Press any key**

Pressing any key will reset the computer to administer another test in the same program series. If the computer is set to test over Drug education, at the end of a test, the computer will reset itself to administer another Drug education section of the program. Again, the number of tests in any program series may span from one to twenty tests. A student may pick any one of these tests to take within a particular program, but they will not be permitted to exit the program or take a test from any other program. For example, a student testing on Drug education will not be permitted to take a test for Alcohol education without the direct intervention of the program manager who must exit the Drug Education program and reenter the Alcohol Education program. Since this necessitates the feeding in of passwords, it should not be done while any inmate is in the lab. Usually different computer in a lab are set up to test particular programs during lab operation.

Once a student finishes a test he should move out of the lab. A next student testing over the same program can then take his place. Once any key is pressed in response to the above message, the computer will once again display the name of the program it is set to test over and request a user identification number. The screen will appear as follows:

DRUG EDUCATION PROGRAM

Identification Number:

At this point the computer is ready to administer another test within the same program series to a new student.

At this point, the computer has completed its testing loop, identifying, administering and scoring a previous test and then resetting itself automatically to give another. With a correctional population, it is at this point that the system manager needs to be vigilant. If the previous student is not taken off the testing computer when the warning buzzer is sounded, there is absolutely nothing to stop that same student from signing in for another test. If he failed his previous test, he can try it again.

If he passed his previous test, he can sign on to the next test and take a look at some of the test questions or he can take the test for another inmate signed onto the same computer.

The program however does monitor this kind of activity and it is reflected in the Statistics section of the program in the form of a ratio. The program manager needs to pay attention to the test finished buzzer and move inmates who have finished their test along making room for the next student to utilize the computer.

As the previous student leaves, the computer will be waiting for the input of another identification number. The screen will appear as in the figure above. Once the new student signs in with his identification number the computer will do one of two things based on whether or not it recognizes the user. If the software determines that this is a new student, it will ask him or her for a lot more information that if the student is recognized before letting them proceed with the taking of a test. However, if the computer searches its memory banks and finds that a test record already exists for the student signing on, a box will appear on the screen with the message, "Your number exists in the program. Press any key." The configuration will appear as follows:

DRUG EDUCATION PROGRAM

Identification Number: 12345

Your number exists in the program.
Press Any Key

Once a key is pressed, the new students entire test record will be brought up for him or her to review. The student can review every test they have taken in the program series, the date they took the test and the score they received. At the bottom of the record page will be the statement:

LAST TEST TAKEN WAS TEST #___ ON Mo./Day/Year

If the student had for instance taken test one on 5/7/96 and failed it, this program line would indicate that they had taken a test on 5/7/96 and the number of the test they took. However, no score would be reflected for that test in the program record. This indicates that the student had indeed taken a test on that date and failed it. The next program line immediately under the Last Test Taken line will be:

TYPE TEST NUMBER (1-20) AND PRESS ENTER:___

At this point, the program is actually asking the student which test in this program series they would like to take. The program will be waiting for a test number from 1 to 20 to be typed in the blank which a blinking cursor.

Once the computer is entered into the RUN TEST mode, it is in a locked loop testing cycle.

There is no way to break out to the main menu, a different program or the disk operating system without rebooting or shutting down the computer. This is an intentional security design feature.

We recommend that if at all possible that different computers be dedicated to testing different individual programs on a continuous basis. This circumvents the necessity to keep rebooting or turning the computer on and off in order to switch to the testing of another program. It also lessens the number of times a security password is required to be entered into the machines in the proximity of inmates and saves much wear and tear on the computers themselves.

KEEP IN MIND THAT THE ONLY WAY THE MAIN MENU OR ANOTHER PROGRAM CAN BE ACCESSED FOR TESTING AFTER THE COMPUTER HAS BEEN PLACED IN TESTING MODE IS TO APPROPRIATELY EXIT THE PROGRAM USING THE ENCRYPTED PASSWORD OR TURN THE MACHINE COMPLETELY OFF.

When the student has finished testing, the latest score will be added to the student's

individual testing record and there will be a prompt at the bottom of the record file telling the user that:

Testing session is completed
Press any key

Once any key is pressed, the program will reset to administer another test in the same program series. The computer will flash onto the screen, the name of the program and request an identification number in order to either start a new test file for a new student, or access an already established test file on a different person. The message which will appear on the computer screen will be as follows:

Identification Number:

OTHER MAIN MENU FUNCTIONS

Below is a picture of the main program menu. The previous section of this manual explains the RUN TEST option number, selection number one, and its functions. We will now be moving to

2) The EDIT USER LIST section of the program.

EDIT USER LIST FUNCTION:

Choosing the EDIT USER LIST option of the main menu will cause the program to display a sub menu which gives options for the Edit User section of the program. This sub menu will look like this:

EDIT USER:

The Edit User List function was designed to allow changes **ONLY** to the identification number and the housing unit number after they had previously been entered into the computers memory. Periodically when a student is signing onto a computer the user makes a mistake and enters a wrong student identification number. Without a means to change that number, the student is saddled with the responsibility of remembering an incorrect number in order to gain access to program testing.

IF THE STUDENT REALIZES THAT HE OR SHE HAS TYPED IN A WRONG IDENTIFICATION NUMBER, BUT HAS NOT YET ENTERED IT INTO THE COMPUTERS MEMORY:

It is very easy to change the number by just back spacing and retyping. However, if the wrong identification number is entered into the computers memory by hitting the Enter key, the computer will continue to recognize the student by the wrong number. It is our experience that asking a student to remember a wrong identification number does not work well. Remember that the program does not recognize students by their names, but only by their identification numbers.

Once a wrong identification number is entered, there are only two means of correcting the situation. Since the mistake will be made within the locked testing loop or RUN TEST section of this program, you will not be able to break out of this loop to enter the EDIT USER LIST section of the program where the problem can be corrected without either rebooting or shutting off the computer and starting again. Some operators find it easier to just execute the first test quickly and in a bogus fashion by pushing all A's or any other bogus answer combination just to finish and dispose of the test. Once the bogus test is failed and graded by the computer, the user can then sign on again under the correct identification number. The bogus test will remain in memory and can be deleted from the program at a later time.

Running a bogus test circumvents the problem of having to reboot and type in passwords in the presence of inmates. The Edit User List is another remedy for this same problem, but IT MUST BE ACCESSED FROM THE MAIN MENU. If this means of correcting the problem is chosen, the program must be rebooted or restarted in order to access the main menu. It is easier to take a bogus test, fail it and retake it again. The computer will have started a testing file on the wrong number the inmate entered the first time. It can be removed later through the main program menu.

THE DISPLAY/EDIT USER LIST FUNCTION:

Selecting this function from the main menu and hitting Enter will have the effect of showing you record by record each and every user contained in the memory of your system, one record at a time.

A count of the total number of users who are currently involved or have finished a particular program and have not been deleted can be found in the lower left hand corner of your screen. PAGE UP AND PAGE DOWN keys will allow you to move throughout the list at the push of the key until you find the file you are looking for to review, edit or delete.

Make sure that your number lock is off before using the page up/page down keys to review inmate test histories. This program function can give you at a glance all the information the system contains about each student's progress within a particular program. Each test file will contain the name of the student, his number, a record of all the tests he or she has taken and the date each test was completed along with each individual test score taken within that rehabilitation program. If you need a hard copy of

any file to verify full or partial program completion, you can hit the print screen key on your keyboard. If your computer is hooked up to a printer, the file was designed to take up just one page you will be printed out a full report on any particular student at the push of a button. Under normal circumstances it is not recommended that testing computers be attached to printers unless the lab is closed and the printer is moved from machine to machine to make hard copies of test records with no inmates present.

You cannot delete a user from the Display/Edit User section of this program. To delete a user, you must move to the Delete Users from User List section of this submenu.

DELETING USERS FROM USER LIST

DELETION OF INMATE RECORDS FROM THE PROGRAM: The Deletion of Inmate Records is not directly protected by a password, but relies on the initial program password for protection. For this and several other reasons, it is advised that inmates never be allowed to access or even view the main menu or any sub menu of this program. Having obtained the initial password, a malicious inmate could navigate to this section of the program and delete records.

The Edit User List section of the program allows the deletion of any inmate record out of your computer's memory. There are basically three reasons to delete a user from your memory banks: (a) After the user has completed the program, and has been given a certificate or some other kind of credit, and a record of program completion exists elsewhere. (b) If a student does not finish a program and has not requested to finish the program for over a year, the program manager can clear the computer memory bank of the students file. (c) Files which have been entered with the wrong identification number can be deleted.

Since the implementation of this system throughout a state prison system permits rapid standardization of programs throughout the entire correctional systems, it is NOT advisable to delete inmates who have not completed programs be deleted immediately. They may have been transferred to another unit where computer assisted labs exist. If this is the case, they can pick up with the program at the exact place they left off in their previous unit.

In time, program managers may find themselves receiving calls from program managers at other institutions requesting verification of a previous inmate's program scores. This was happening regularly at the time I retired from ADC.

Since computer assisted programs are quick and easy to implement, you may find another program manager calling for information on an inmate who started testing within your program and has since moved to the same program at another prison complex. Once you receive a call and confirmation that the inmate has matriculated into another

identical program at another institution it is safe to erase the incomplete inmate file. If not, the safest course would be to wait at least six months before erasing an inmate record. When the Delete User option is chosen, the screen will appear as follows:

DELETE USERS FROM USER LIST

Once you see this screen, you can page up or down to find the particular user you would like to delete. **MAKE SURE THE NUMBER LOCK ON YOUR KEY BOARD IS TURNED OFF AND YOU HAVE ACCESSED THE PAGE UP / PAGE DOWN FUNCTION ON YOUR KEYBOARD.**

When the DELETE USERS FROM USER LIST option is chosen a student test file will appear on the screen. On the bottom left hand corner of the screen will appear **(D) delete user**. This student file may or may not be the file you wish to delete. In order to find the file you want to delete hit the Page Up and Page Down keys till you find the inmate test file you wish to delete. The files are not in any particular order. Once you find the file you desire to eliminate, push the Del key on the keyboard. Another screen will flash up asking you if you are sure you want to delete this particular inmate file.

Review the file carefully and make sure this is the one you want to eliminate. You have the option of changing your mind at this point. If you push N for no indicating that you do not wish to delete this particular user, the program will return to the menu.

If you push Y for yes, the file you have targeted will disappear from both your computer screen and your test storage files and cannot be retrieved. You will be returned to the DELETE user file section of the program.

You can reaffirm that the file has disappeared by checking the number of users appearing on the bottom right hand corner of your computer screen. The total number of users should be one less than the previous count. When you are finished deleting your targeted files, press R to return to the EDIT USER MENU. If you desire to return to the main menu from this point, choose (3) EXIT TO MAIN MENU.

CREATING TESTS FOR YOUR OWN PROGRAMS

THE EDIT/ADD TEST QUESTION OPTION

One of the most unique and valuable features of this program is that YOU the ability to create multiple choice tests to your specifications for your own study booklets. This is a feature that some programs costing hundreds of times more reserve for themselves,

making you pay for programming costs if you want to customize your tests.

The Edit Test option allows you to both program this software to test over material you feel is particularly important and experiment with providing inmates an endless variety of program options. The number of rehabilitation options is only limited by your ability to dream up multiple choice test questions over material you can hand out to your inmate populations. If you think you have a better idea or find yourself getting a great inmate response from a particular group or program, test it!

In addition, this program gives group therapists and programs using a didactic delivery system a means of measuring how much material is actually being assimilated by inmates. The development of tests to be programmed onto this system forces thinking through of the material being presented. Program goals must be determined by the program manager, crystallized in his or her thinking and then programmed into the software for testing.

This option further gives the user extreme flexibility in the composition of test questions. With a few key strokes you can add a test question, eliminate one, or change any aspect of the question you do not feel is suitable. This program function provides this program with the ability to respond quickly to rapidly changing environments, program goals and information flow.

The testing material can be totally molded and formed by you to reflect what you think is important about a certain body of information. In addition, how you compose your test questions and the level of difficulty you build into fine tuning discriminations between the right and wrong answers allows you to change the entire nature of testing. You are in complete control of determining how fine a discrimination the student must make in understanding the material you deem is important. If only a general understanding is necessary, you may want to consider making the test questions easy. If you feel you want to set high standards, you may want to make them a little harder.

For the program to serve its intended purpose, students must not be able to guess their way through your test questions and must score 70% or more. In order to test the suitability of a test to perform for its intended purpose when we finish programming a new test we pull students in to take the test who have not been exposed to the study material. We then promise them a small reward if they can guess their way through the test questions and pass the test without studying. We do this several times to make sure the newly formulated test does indeed possess the ability to discriminate between someone who has studied the material and someone who has not.

ACCESSING THE EDIT/ADD TEST QUESTION MODE:

Scan down to option number three. You will see 3) EDIT/ADD TEST/QUESTIONS. It is quicker to move the highlighted bar down to option three and hit enter, but you can also

access the option by hitting the number 3 and then pushing enter.

Once you hit enter, the computer will bring up a security verification screen. This password can be changed only if you know the previous valid password. Inmates should never be allowed into this section or to even know the section exists.

Password Verification:

When you type your password into the computer, the password will not be displayed on the screen. The computer will warn you to remove any unauthorized personnel from the area until your password is fed into your system. If the correct password is given the computer, the system will then ask you if you desire to change your password. If you feel that your password may have been compromised, you should not hesitate to change it. After giving the computer the correct password to enter the Test Edit Section the program will ask if you want to change the password.

Password verification

Do you want to change the password?

CHOOSING THE TEST YOU WOULD LIKE TO DEVELOP OR EDIT

After you give the computer the correct password and hit enter, the computer will ask you:

WHICH TEST WOULD YOU LIKE TO ADD OR EDIT (1-20)?

Each testing program is capable of administering up to twenty separate program tests over any program. EACH TEST MUST CONTAIN AT LEAST 20 QUESTIONS FOR THE PROGRAM TO WORK. The larger the number of questions a test contains, the more secure it will be and the better job it will do at testing. If the test only has the minimal number of questions (20) then you can be assured that each question will be chosen at least one time per test. They will be chosen randomly and the answers randomly scrambled, but each test question will appear on every test. The more test questions you have, the greater the probability will be that the answers will be different on each test and that no two tests will be alike. With enough test questions, there is a probability that no two tests will ever be alike. The program will not work with less than twenty test questions per study booklet; however, it is highly recommended that at least thirty to forty test questions be entered per study booklet if possible.

IF YOU ARE STARTING WITH A COMPLETELY NEW PROGRAM WITH NO TESTS INSTALLED, any option you choose will return the message, **Test___ not found. Create? Y/N**. If you choose Y or yes, the program will present you a blank test question skeleton for you to fill in.

IF YOU ARE EDITING A TESTING PROGRAM WHICH ALREADY HAS PREVIOUSLY BEEN PROGRAMMED IN: The program will give you the first question of the test number you asked for. At the bottom of either screen you will see the following editing functions:

PageUp/PageDown - Prev/Next Question:

This function allows you to page up and down through your test questions. **MAKE SURE YOUR NUMBER LOCK IS OFF** or the section will not work. By pushing the page up/page down keys, you can view or edit any test question contained in the storage bank for the particular test you requested.

If you know the test you are requesting has no previously programmed material and is an empty shell, you can start programming test questions into the blank screen presented to you. **DO NOT PAGE UP OR DOWN AND LEAVE BLANKS BETWEEN TEST QUESTIONS WHEN ENTERING NEW QUESTIONS**. The computer needs to have the test questions continuous as it pulls them up in line. If you leave a blank, that blank will be presented onto the screen at the time of testing. It will appear as a blank screen and the only option shown will be e). If you run your test after programming the data bank and find a blank, go back into the Test Edit Section, search for and delete the blank or it will continue to appear on the testing monitor along with your questions.

INSERT - ADD QUESTION: The next line along the bottom of the computer screen in the Edit Test Question Section will read, Insert - Add question. **THIS REFERS TO THE INSERT KEY NOT THE LETTER I ON THE KEYBOARD**. If you push the letter (i)l, the program will not respond, however if you push the INSERT KEY, you will be given an empty test question screen.

This means that a new space for you to write a test question has been created.

Note that when programming test questions into the computer, the correct answer must be placed in a particular position, the first. To move the cursor and start writing you must also push the E key for Edit. Consequently, you are first creating the space or skeleton for a new test question with the insert key. Secondly, you are editing that question to suit your desires with the E or edit key.

E - EDIT QUESTION: As mentioned above, the Insert key will only insert a space for you to program in a new question. You must push the E key on the keyboard to begin typing your question. This key activates the editor. The editor will work in a wrap around fashion and your cursor may disappear off of the screen for short periods of time. You will never run into a problem if your question remains within the confines of

all the boxes provided for the question and the answers. If you are not careful however, the capacity of each box can be considerably exceeded. When you go to run the program, the question will still come up, but if you exceed the size of the box while programming, the entire question will not be able to be seen on the screen.

The capacity of the programming box for each test questions is 256 characters. Your entire test questions cannot exceed 256 characters or it the entire question will not appear on the screen when presented for testing. There is some fudge factor built in that gives you a little leeway if you exceed the boxed in limits given for a test question. If you do exceed those limits however, you need to check when you run the test to make sure that particular question fits and is presented by in its entirety. If it is too large for presentation, you need to reenter the program and edit it down or it will continue to be presented in an incomplete state.

*******WARNING! THE PROGRAM WILL NOT WORK CORRECTLY IF YOU ONLY PUT DIGITS OR NUMBERS ALONE IN THE ANSWER SPACES. THE NUMBER MUST BE FOLLOWED BY A WORD OR A SYMBOL FOR THE PROGRAM TO PICK IT UP.**

For example you cannot put into an answer slot the number 34 alone, but you can put 34% or 34 beans. We recommend that after each test you program into the system, you run that test to make sure you have not exceeded space limits. left any empty spaces between your questions or typed in any lone standing numbers. When you page up or page down among your new questions, they should follow each other without a break. There should be no empty shells between your test questions. If there is a break where there is a blank test question, delete that space or question shell by pushing the delete key.

ONCE YOU PUSH THE E or EDIT KEY: The blinking cursor will appear within the QUESTION box, and you will be ready to type in your question and answers. The question will wrap around until the question fills the full block. It is strongly recommended that your test questions be kept short and concise.

Once you have finished programming in your question, hit enter and the blinking cursor will move down to the next box allowing you to type in a wrong answer. Remember, the correct answer **MUST** be in question slot number one for the computer to track it. Types in each wrong answer into its appropriate slot then hit enter until all the wrong answer spaces are full. Remember, the program needs at least 20 questions to operate correctly. Once you have completed a question with its associated right answer in question slot one and the other four wrong answers in the slots that follow, you can hit the **INSERT KEY** again and a new blank will appear for you to program in another test question.

While you are in the edit mode, you will be able to page up or page down to review the test questions you have created. When you are finished, count the number of questions you have programmed for a particular test to make sure you have at least twenty before

attempting to run the test.

TO FIXING YOUR TEST QUESTIONS INTO MEMORY:

To put enter your test questions into the computers memory is simple. Once you have entered your last test question with the editor, hit the Esc or Escape key. The computer will automatically place your new questions into memory in the appropriate test bank of the hard disk. The test will respond when called up from the main menu.

Once the escape key is hit, the computer will reset itself to the main menu. You can then run your test and see how it looks.

MAIN MENU

4) QUIT

The quit option on the main menu will do one of two things at this point. If the program is being operated from a menu, it will return to that menu. If the program is operated from a particular drive and is not attached to the menu, then it will return to the drive. It is recommended that the program be operated from a pass worded menu.

If you have not installed your program onto your hard drive yet, make sure to read the installation manual prior to attempting installation.

PROGRAM ORINATION AND HISTORY

SUMMARY

USERS AGREEMENT

COMPUTER ASSISTED SUBSTANCE ABUSE PROGRAMS FOR CORRECTIONAL POPULATIONS

By Jerry A. Marzinsky, M.Ed.

INTRODUCTION: HISTORY AND CHALLENGE

In the last twenty years prison populations throughout the United States have soared to over one million people. Overcrowding is epidemic with 44 states currently under Federal Court order to do something about the problem. (1)* For the tremendous costs involved, progress as measured by recidivism, is singularly unimpressive. Of the 108,580 persons released from prisons in 11 states in 1983, representing more than half of all released State prisoners that year, an estimated 62.5% were rearrested for a felony or serious misdemeanor within 3 years, 46.8% were reconvicted, and 41.4% returned to prison or jail."*(2) Inmates sentenced for a drug offense accounted for 44% of the increase in the prison population from 1984 to 1991 and 32% inmates committed their offense under the influence of alcohol.(*3)

It is common knowledge that a very large number of prisoners, estimated by some to be as high as 85%, are incarcerated for substance abuse related crimes. Archaic thinking in the past threw large amounts of money at this problem. The results were a failure. Traditional program delivery formats have proved extremely expensive, inefficient and totally inadequate to deal with the enormous number of inmates requiring substance abuse education and services within our prison systems. Ignoring this problem as prison systems are currently doing is akin to ignoring a festering sore.

In addition, many current programs are repeating past mistakes by not taking into account the unique programs encountered in attempting to provide valid, measurable programs for inmates. After close to twenty years working in programs within correctional environments, I've learned one thing. If you can't measure what inmates are learning, they are not learning anything. In addition, without measurable feedback, administrators find it much more difficult to fund programs. Historically these two very important, but largely ignored factors have resulted in adopting expensive and ineffective programs which broke down with exposure to the correctional environment.

The result was administrative disillusionment with program cost and effectiveness and the drying up of program funding.

What is needed is a low cost, high quality, measurable, large scale program delivery format which takes into account the numerous special problems unique to prison environments. The dilemma has been that no such customized program delivery format has existed, until now. When legislators demanded that something be done about the massive recidivism rates of prisoners, prison administrators put out bids for the cheapest contractors they could find to address the substance abuse problems of the inmate population, and they got what they paid for.

The following paper summarizes what has been learned in a sixteen year struggle to address the unique problems associated with programming massive institutional populations.

For the last eight years I have worked in the Psychology Department of a large prison

complex in the western U.S. During that time we have struggled with the many perplexing problems involved with programming large institutional populations. After years of trial and error, we have finally hit on some inexpensive, workable solutions.

* Sources at end of paper

CRITICAL FACTORS WHICH MUST BE OVERCOME FOR LONG TERM PROGRAM SURVIVAL IN CORRECTIONAL SETTINGS

A large number of rehabilitation oriented programs which work very well in civilian settings will give the illusion that they are effective with prison populations, but in fact suffer expensive, painfully, lingering deaths when imported into correctional environments.

Having a keen interest in program development, I repeatedly found myself participating in program implementation. I watched with interest each time a new rehabilitation program was implemented at our prison complex. Over time, I saw program after program imported into the prison environment fail. Over time, I saw that they were failing for the same reasons. This long string of failures did not seem to stop prison administrators from making the same mistakes again over and over again. Over time, I drew some solid conclusions about the longevity and validity of rehabilitation oriented programs operated which were set up within correctional environments.

FACTORS WHICH MUST BE CONSIDERED FOR CORRECTIONAL PROGRAMS

SPACE REQUIREMENTS: The more physical space required for program operation within prisons and the length of time that this space is occupied could cause the demise of a prison rehabilitation program.

Space is at a premium in today's prisons. To survive, a program must be space efficient.

STAFF TIME: The more staff intensive a program is, the less likely it will be to survive long term. Staff are stretched extremely thin and staff turnover in most prisons is very high.

STAFF BURN OUT: Difficult or tedious or complicated programs dealing with repetitive presentation of materials or repetitive grading of tests will quickly burn out program staff. Staff who are ordered to or do not want to run programs are destined to provide loose security and do a poor job.

INMATE CHEATING/SUSTAINABILITY OF PROGRAM INTEGRITY: Inmate cheating

within the prison environment is a constant and if not taken into account will very quickly render a program totally useless or worse, reinforce criminal behavior. The potential for inmates to cheat on programs must be taken into account and prevented.

SUITABILITY TO THE CORRECTIONAL ENVIRONMENT: If a program does not meet the numerous overt and subtle requirements for operation within prison systems or provides a security threat, it will not survive.

FLEXIBILITY: The program must be easily adaptable to constantly changing operational times and information resulting from research and experimentation. The format must be flexible and adaptable to various physical sites.

QUALITY: Programs must be designed specifically for the educational level of inmates and must be interesting and directly pertinent to their lives. This system was written on a 7th grade level.

EFFICIENTLY AS AN INMATE MANAGEMENT TOOL: If a program can keep large numbers of inmates positively occupied for relatively long periods of time, the prison unit becomes much easier for administration and security to manage. In the prison environment the saying that “Idle hands are the devil’s workshop” is very true.

MEASURABILITY/ACCOUNTABILITY: If program output and inmate learning cannot be measured and evaluated, administrations may have a very hard time justifying continued funding and operation. In a worse case scenario, without measurable feedback one doesn’t really know if the program is actually accomplishing anything at all.

If what inmates are learning in a correctional program cannot be measured, rest assured that they are learning little to nothing. This however will not stop administration from turning in program participants to legislators giving the false impression that something positive is happening within prison systems. Turning in hollow numbers seems to be the norm.

RELEVANCY TO INMATE PROBLEMS: Wardens and prison officials are more willing to invest in programs centering on areas that address immediate problems and have the potential to reduce recidivism. If programs are not interesting and directly relevant to the inmate, inmate program participation will dramatically drop.

THE MINUTE A PROGRAM IS OFFERED FOR CREDIT, INMATE CHEATING CAN BE EXPECTED. If the program is not operated for credit, but operated on a voluntary basis, program managers for the most part do not have to worry about cheating. Program participation will be very low, but the inmates participating are more likely to be doing so because they want to learn about their problems.

However, managers should pay keen attention to secondary pay offs that inmates may be receiving. Even if a program gives out a certificate of participation, program numbers

will increase as inmates will want whatever they can get to show the parole board, even if they slept through the program they got the certificate from.

In addition, prison gangs will attempt to utilize voluntary programs as a meeting place where they can gather to give direction to inmates at places they cannot physically get to otherwise. A voluntary program meeting such as A.A. may serve as a place to pass drugs, plan scams or forward messages between gang members. Requiring that inmates pass a bank of computer assisted programs before being allowed to attend such voluntary programs will greatly reduce the number of illegitimate attendees.

As soon as a program is offered for system credit however, for example for parole board review or for any kind of reward, program managers should expect inmates to take the path of least resistance through a program. This often converts to inmate cheating on any kind of measured program. Extensive cheating on any program will render it useless and reinforce criminal behavior.

THE EVOLUTION OF A PRACTICAL PROGRAM

After watching a wide variety of commercial, home grown and contract programs fail at our prison complex; experimentation was started with a number of program formats which better fit our unique correctional environmental requirements. We started with the implementation of book, audio and video taped programs. These information delivery formats failed due to cost, requirements for expensive special equipment, inflexibility and difficulty testing over the material.

Study sheets were developed over already existing material. Inmates were required to study program packets and take tests over the study booklets in a monitored group setting once a week. This format allowed us to validly program more than triple the number of inmates we were able to run through traditional group or didactic delivery formats and yielded measurable results.

It was not long however, before recorders, tapes, books and converters began wearing out or were being stolen. The paper and pencil tests used to measure inmate learning were and distributed around the prison yard as currency within the inmate economy. Converters used to power tape recorded programs were stolen and utilized to power tattoo guns.

As each program lost materials and hard copy tests were stolen, program validity suffered and inmate cheating soared. Program costs and staff maintenance factors soared as ill fated attempts were made to keep manually tested programs valid despite increasing numbers of hard copy tests being stolen. With each stolen test, program validity went spiraling negatively out of control. At the end of six months, inmates had in one way or another stolen, copied or traded a sufficient number of hard copy tests to render testing useless. The inmates who had stolen the tests traded them for favors or

store items.

To circumvent expensive equipment replacement, stolen tape players which were turned into tattoo guns and stolen materials; we regrouped and developed our own comprehensive substance abuse study material. The new programs were centered study booklets which we could share with other institutions without having to worry about legal issues. There were several advantages. Such booklets were fully referenced and very inexpensive to reproduce. Home fashioned program study booklets proved to be the most successful and inexpensive means of conveying program information to the inmate population. Booklets replaced much more expensive equipment like tape recorders or video tape players and the overall operational cost of program operation dropped drastically.

Another advantage of the booklet format was that subject expertise was built into the study booklets which were written on a seventh grade level, completely eliminating reliance on expensive, highly trained staff to run equipment, lecture or run educational groups to keep programs consistently operating.

There were several additional advantages to utilizing study booklets to disseminate the massive amount of information necessary to run a large scale program within a correctional setting. The only materials required with this approach were the study booklets and tests. Both were very inexpensive to reproduce. When inmates started stealing and sending study booklets home to their wives and loved ones, we knew we were on the right track. Inmates don't risk stealing something they don't value. They were sharing with their significant others what they were learning, however, the constant theft of study booklets became a problem.

We addressed it by telling inmates that their study booklet was their ticket for admission to take a test. If they did not turn in their study booklets, they were not admitted. Study booklets long claimed lost began showing up on a regular basis. This requirement made the recirculation of these same study books a functional proposition. When covered with old manila folders that prison administration was going to throw out, these study booklets lasted three times longer than uncovered versions.

This once again cut the cost of programming inmates significantly.

The advantages of issuing study booklets to inmates to disseminate programmed information didn't end there. Meeting space within prisons is very limited and in high demand. The implementation of study booklets to disseminate information addressed the scarcity of classroom space. Space requirements for program operation centered on the study book format were virtually eliminated. Since this format eliminated the need for teachers, lecturers, counselors or other kinds of staff intensive educational group formats, valuable classroom space and time slots were not infringed upon. Inmates studied in already existing space, on breaks from work, during library hours or in their cells. The total number of inmates who could now be involved in substance

abuse rehabilitative programming was now limited only by the number of booklets we could reproduce and hand out to inmates. Study booklet based programs were on-going and never stopped like group or classroom educational programs taught by staff which ran for a few weeks then ended.

Unlike classroom or group delivery formats, the study booklet format permitted inmates to move through programmed material at their own rate. Inmates did not have to show up for testing every week if they were not ready or their jobs interfered. This brings us to the testing of the programs. Remember, if inmates are not being tested as to what they are learning in programs, you can pretty well bet that they are learning nothing, just taking the path of least resistance.

No matter what testing format is being utilized, it is necessary to have staff monitor the testing site. The study booklet format disseminating information over many different program topics permitted inmates to converge at one time and place to test over programmed material. For example, inmates participating in the self study program on alcohol, drugs and co-dependency could all take different tests over those various programs in the same monitored testing room at the same time. If additional testing time was required, inmate attendance could be staggered by time to keep the testing room full and a constant flow through the testing site. We issued passes so inmates would show up at fifteen minute intervals. By the time some were finished testing and leaving, others were coming in. This provided for the most efficient utilization of classroom space during a scheduled testing session. Before the development of Computer Assisted Testing, program validity was assured by staff test monitors who watched for cheating.

The tested, take home study booklet system allowed for very large numbers of inmates to move through many different program formats in a short period of time while keeping the overall number of inmates present at the testing site at a manageable level. The staggered testing times achieved this. One thing security is not going to be happy about is large numbers of inmates congregating at any one place without strict monitoring.

Adopting the study booklet format kept the overall cost of program operation extremely low and the return on the amount of staff time spent versus the enormous amount of time inmates were kept productively occupied at a very favorable ratio. This format allowed us to keep large numbers of inmates moving through a variety of validly tested programs at a very low cost in terms of staff time and physical resources. In the old days, utilizing the study booklet format and monitored paper and pencil testing of programmed material, we thought we had finally hit upon a solution to the programming problems associated with the operation of large inmate programs for even larger inmate populations. With time however, two major and seemingly insurmountable problems reoccurred. We had seen them before, and they stood like a stone wall before us.

Incessant inmate cheating, copied answers and stolen tests were constantly leaking into the inmate population, slowly but surely invalidating programs. Predictable paper and pencil tests were extremely vulnerable to various kinds of cheat sheets. The proliferation of these simple but effective cheating devices began to compromise even the most strictly monitored of programs. If programs awarded any kind of credit at all, any inmate who could obtain the answers to tests could make a good profit selling them. Tests and answers to tests eventually found their way into the inmate economy and became a significant part of it. Copies of stolen tests were passed around the prison yard and traded for cigarettes or food items from the store. The number of inmates failing tests drastically dropped. The compromise of paper and pencil tested programs meant we were reinforcing criminal behavior by running compromised programs where inmates were being rewarded for cheating. As it turned out, many wardens and correctional supervisors cared less as long as the programs were generating participation numbers. They cared less whether or not inmates were actually learning anything in those programs, only that the director kept getting high program participation numbers that could be shown to the governor and law makers.

In addition to the theft of hard copy tests and the part they were paying in the inmate economy, there appeared another serious problem. Staff burnout was eating up correctional staff and causing them to take assignments elsewhere within the prison. The manual grading of thousands of paper and pencil tests and arguing with inmates about the scores they earned proved extremely draining. There were many times when paper and pencil tested rehabilitation programs would go down as staff resisted the boring task of grading tests and putting up with inmates attempting to manipulate a higher score out of them than they deserved. Inmates appreciated instant feedback as to their test results; however the down side of grading tests for them before they left the testing room was that they were constantly arguing with program managers over questions and answers. Although their points were often bogus, their confronting staff trying to get a higher score by arguing over a specific question wore down program managers.

AUTOMATION - PUTTING PROGRAMS ON AUTO-PILOT

After repeated failures with various paper and pencil methods of scoring tests, including having several different versions for each study booklet, we came to the disappointing conclusion that large scale programs operated within correctional settings could not be kept valid through paper and pencil testing. Time after time after developing different tests, it took an average of about six months before the failure rate began to drop which was a strong indication that the tests were out in the inmate population. Periodic room searches by security verified that inmates were in possession of stolen tests and our programs were compromised.

After much thought, it appeared that the only feasible solution around all these problems was to somehow program a computer to do the testing, data banking, scoring and

presentation of test results. The computerization of testing appeared to be the only hope of solving the many difficult problems we had encountered in keeping correctional programs valid.

A search was begun for software which would address the many problems we had to solve. Nothing was found which was affordable enough for prison administration to pay for and none of the testing programs on the market were developed specifically for use with insidious prison populations. With time, it became clear that we would somehow have to develop our own software.

I turned to my wife, Maria, a programmer at the University of Arizona with a list of far fetched requirements. After much work and coming close to a divorce a few times, in less than a year we had a very basic software program which promised to do the job. The program randomly selected 20 test questions from a large data bank of questions and displayed each randomly selected test question onto a computer monitor. This function was designed to defeat the infamous inmate cheat sheet which relied on the predictable order of questions on paper and pencil tests for its success. Since inmate cheat sheets had defeated every previous program format we had implemented for inmate populations, we wanted to make sure this simple but devastating device was knocked out as a threat to inmate programs.

The inmate cheat sheet proved so effective against standard hard copy test that we went the extra mile and designed into the computer program another subroutine which d randomly scrambled the order in which the answers to each test question were presented also. The successful development of these blocks of computer programming meant that even the brightest of inmates could not predict which questions would be asked, the order of those questions or the order of the answers.

The probability of any two tests being the same was minuscule. Once this computer code was developed and put to work on the front lines, the computer rendered the inmate cheat sheet useless.

We thought we had the problem solved after more than a year of development and testing only to discover another problem standing before us. This was the inmate hacker.

Any inmate who knew anything at all about computers was prone to try to break into the system. The battle had now shifted from programs that any inmate could cheat on through the use of simple, miniaturized cheat sheets to a high tech area where only a very few inmates had the knowledge necessary to threaten the validity of tested programs by hacking into the computer. This eliminated ninety percent of the inmate population that had previously posed a threat to the operation of valid programs. The result was that inmates were learning more from tested programs than ever before, and actually began to appreciate what they were learning. They were assimilating more information than ever before because they actually had to study and learn the material.

We had succeeded in making it significantly more difficult for the average inmate to cheat. We thought we had only to face off with those inmates with the limited number of inmates who had the technical expertise to break into a computer. We were wrong. Many of the inmates with such expertise wouldn't have attempted to violate computer labs on their own. We soon discovered they had additional motivation provided by the gangsters on the prison units who ran businesses selling illegal tests in the past and wanted to recover those businesses again. The battle now turned high tech.

We found ourselves concentrating on the development of computer security measures which would block the small percentage of inmates who were knowledgeable enough about computers to break into the program or the computer's operating system.

The first measure implemented was the development of several encrypted security passwords. A test timer was added which informed the program manager if an inmate had exceeded specified testing time limits. An alarm was added that informed the program manager when an inmate had completed his test.

One of the first things the inmate hackers went through to break into our computers were the floppy drives. They stole floppy disks from administrative offices and loaded the basics of an operating system onto the disks. Lame program managers who left the labs in the hands of inmates and disappeared gave hackers the exact opportunity they needed to reboot the testing computers with their floppy disks. In response, we either removed or disabled the floppy drives to prevent such violations. We had no control over lab managers who were assigned to operate the labs.

Once we thought we had done as much as possible to keep the system secure programmatically, we concentrated on making the software as user friendly as possible. We made the system so easy to program that virtually anyone can program into it any multiple choice questions that they can dream up.

Unlike some of the commercial packages which require the company to program your test questions into the software, this system was designed to make it easy for the program manager to program in any test questions over any program social workers might develop for prisons. Functionally this meant that any program or group, over any topic, at any correctional institution in any prison utilizing our alphabet could now be validly tested by computer and made to yield measurable, valid program results.

The system not only assures that inmates must know the presented material for any group or program, but the software automates verification for any program, securely, with minimal staff involvement and at very low cost.

We had taken the system security measures as far as we could programmatically but in lab after lab we found correctional caseworkers giving more and more responsibility to inmate trustees for the operation of these programs assuring us that they trusted this or that inmate with such increased responsibilities despite the fact that these systems

required very little on the part of the program managers. We approved inmate clerks handing out and collecting study material, but they progressively pushed for increased responsibility in the labs and acted insulted if they were not given.

We had taken the software security measures as far as we could and next set ourselves the task of procuring hardware. The prison system was constantly updating its computers and we made arrangements to procure many outdated machines that would have been hauled off to government surplus. Despite this, we could not keep up with the explosion in program growth as the CAP concept caught on. With prison operating funding constantly strained, even used computer shops were too expensive for the number of computers we needed to program the entire complex and other prison complex's requesting computers. We worked around this problem by visiting the local state university surplus warehouse and found an entire graveyard of old, retired and very inexpensive 8088 computers. Although they were not much good to the university, they were perfect for our needs. We were able to pick up an entire computer system for between \$100 and \$150 including monitor, hard disk, hard drive and key boards. Each single ancient computer was capable of validly testing 48 inmates per 8 hour shift. One psychologist in the Nevada prison system did just that. He ran several inmate programs on a single old IBM 8088 computer stationed in his office.

We sent other program staff out to the local army and air force bases to auctions and purchased many more old 8088 computers, some at prices or below what we paid for them at the university. In a short time between these various sources we had a five computer lab up and operating allowing us to potentially program 240 inmates in an eight hour period which would account for up to 700 hours of waking inmate time invested in positive, rehabilitation oriented programming in a single day. This meant that these systems that would only be set into operation a few hours a day were proving to be very powerful inmate management tools. Despite the fact that the computer labs themselves were only being run for a few hours a day, a survey of inmates moving through the labs revealed that between travel to and from the testing lab and studying for tests, inmates were expending an average of three waking hours per each fifteen minute test they were taking on the computer. As far as inmate management statistics go, this exchange rate of inmate time expended in productive programmed activity versus the amount of staff time expended to obtain these results could not be matched by any other correctional program in existence that we were aware of.

We could now powerfully begin to address many of the problems inmates were being sent to prison with valid, fully measurable rehabilitation programs and accurately measure not only how much these inmates were learning but how fast they were learning.

Once a five computer lab was set up and operating smoothly on the Santa Rita Unit, we wanted to see how the system would work on our highest security unit, Cimarron, a level four/five facility containing some of the nastiest inmates on the prison complex.

With the full support of the deputy warden, a computer assisted pilot program was initiated at Cimarron. We started with two computers which were programmed to test over four different rehabilitative programs. Despite having only a few computers, inmate participation quickly reached previously unattainable heights. As a result, we expanded the lab with a few more computers. Student involvement peaked with inmate participation leveling off at 25% of the entire unit of 750 inmates over a one month period. We ran the systems ourselves until the chief psychologist announced that we needed to be spending more time on clinical issues with the chronically mentally ill. It appeared that one of the community organizations had filed a lawsuit and the prison is driven by law suits.

Before we had to pull out and turn the system over to prison caseworkers, we had cheating eliminated. Since the computers were doing all the work, staff burn out factors were vanquished and the monitoring of the programs testing lab became one of the easiest of correctional counselor assignments. Required staff involvement, and physical space requirements were so minimal that a caseworker could now run a large scale substance abuse program utilizing only a few hours of his time a week in order to program 25% of an entire prison unit. This small amount of time detracted little from their caseload responsibilities and benefited all counselors on the unit who went inmates on their caseloads to be involved in substance abuse education.

The ongoing operational cost of these systems proved far below that of any previous rehabilitation program the prison system had ever initiated. The cost of electricity to drive a computer was approximately that of a 200 watt bulb. There was no more staff efficient program as the computers and the inmates themselves were doing 95% of all the work. In effect, inmates were going a long way toward rehabilitating themselves with the guidance of program study booklets and computer assisted testing to verify they were learning the material.

Over time, many unexpected benefits of CAP's also popped up. Although the vast majority of inmates belly ached and complained at first exposure to CAP's as the program required much more from them than any other correctional program ever implemented, we found that past the first three tests, inmates actually enjoyed taking program tests on the computers and getting immediate feedback.

For some, this was the first time they had ever worked on a computer and the closest thing to a video game they had encountered in prison.

Over three years of continuous operation, the Computer Assisted Program format clearly demonstrated that this program had what it takes to stand up to correctional environmental pressure and con-artist scams when operated in accord with its design parameters. Although the program required very little staff time, the little it did require had to be available and the lab managers needed to be volunteers. Time after time, staff forced to run the program against their will let inmate clerks take more and more responsibility for program operation until the system was compromised and rendered

useless.

Since the first days of operation with a single obsolete computer in the corner of my office, and the time our first inmate approached a testing computer with cheat sheet in hand, we had come a long way. Computer labs containing as many as ten computers had been set into operation capable of testing every inmate who desired programming in the period of a week.

Since the computer testing program is easily programmable some staff began to develop additional programs they felt important to the rehabilitation of inmates, programs of their own design.

Despite being a major advance in permitting correctional and educational programs staff to validly measure learning, the establishment of computer assisted correctional programs proved to have many other beneficial aspects. One of these facets was that the system lent itself to the standardization of programs throughout correctional systems. Such standardization allowed inmates to begin a program at one institution and complete at another. The computerization of Programs also allowed them to be established quickly and easily without the extensive staff training required of other types of programs. Once program study booklets had been designed and test questions programmed onto this testing software, it became a very easy matter to add that very same program to all testing computers operating throughout the system or company. Since inmate movement in prison systems is constant and frequent, standardization of programs saves a lot of wear and tear on everyone's nerves and allows the Department of Corrections to take a systemized approach to the delivery of psycho-educational or substance abuse services.

Since inception, throughout our entire state correctional system, a whole battery of motley, inexpensive, IBM compatible 8088 personal computers have administered over 10,000 successfully passed substance abuse education tests accounting for close to 20,000 hours of waking inmate time in our state alone. In addition, the Nevada Department of Corrections evaluated the concept and adopted the concept running labs successfully for several years before a non-progressive warden was appointed.

Lacking understanding of the system and not wanting to put forth the effort to understand, he arbitrarily declared that during his tenure, no inmate in his prison system would be permitted to work on any kind of computer.

Unfortunately, people with such Neanderthal mind sets are all too often appointed to manage prison systems in the U.S. Citizens don't care and don't want to hear about prisons and prison systems. Unfortunately, thousands of prisoners are being released into U.S. cities and towns every month and are coming out of our prisons in much worse shape than when they went in. They are not being afforded counseling or programs as prison systems are reporting that they cannot afford the money or staff time to run rehabilitative programs. The sad truth is that they don't care and don't want to make the

effort or take the time.

As a result, this system which cost in the neighborhood of \$50,000 of programming and development time and years of trial and error experimentation with what worked and what didn't, is being offered to the public free of charge. This will serve to fly in the face of Wardens and corrections officials who constantly report to the public that they cannot afford programs, space or staff time to run programs to rehabilitate inmates.

There is no more staff efficient, economical, completely measurable system on the market developed specifically for the large scale programming of inmate populations. There are no programs of this complexity and magnitude being offered free of charge to any corrections system who will pledge to operate the system strictly according to the program's design parameters. There is no more efficient program available for the programming of large populations of inmates. There is no program which reverses the costs with the more inmates programmed, the less expensive the system becomes per inmate programmed.

Unlike many other programs, the opinion of large numbers of inmates was taken into account in the development of this system. It was designed specifically for the inmate who during the development of this system was considered the consumer. It was developed specifically for the programming of inmate populations and took many years to develop and test. Inmates were recruited to evaluate the program and their evaluations upon program completion were glowing. Virtually all inmates who fully completed at least one computer assisted program gave the system high marks. A number of these inmate evaluations will be presented on the Patrick Website for examination. These evaluations are randomly selected and represent the positive attitude of inmates toward this program upon completion. As mentioned earlier, inmates were stealing program booklets and sending them home to their loved ones to show what they were learning and to help them understand some of the problems they were having with alcohol, drugs and parenting. Inmates don't steal what they don't value.

Because the mind set of correctional officials at this time is punishment and vengeance oriented, it is not expected that even offered free of charge that this system or any system that has proven itself to be effective in the rehabilitation of inmates will be adopted by prison systems who are more into punishment than into the helping inmates understand what has gone wrong in their lives and how to begin fixing the problems.

This is despite the fact that research has shown that punishment only temporarily suppresses unwanted behavior which will again emerge when the punishing stimulus is taken away. Look at the last speeding ticket you got. You got a hefty fine, it slowed you down for a few months, but now your back to your normal driving habits.

If nothing else, it is hoped that what we have learned from an entire career working with large institutional populations of prisoners will prove valuable for upcoming generations

of social workers and counselors who will be working in correctional settings. Our prison systems are out of control monsters that are festering sores in our society. They are costing taxpayers billions of dollars a year and breeding generations of criminals. They are graduate schools for criminals where the number of our citizens being admitted is now about equal to the number of our citizens going to college.

With the offering of this system and the written programs that go with it free of charge, let it be known that there is absolutely no reason and no merit behind the claim of corrections officials that they cannot afford rehabilitative programs for their prisons. Their only reason they now have for not providing programs is the bottom line truth that they do not want to, and that is the truth.

The only thing that will be requested in return for the download of this program and its associated written study booklets is a signed statement that any system downloading this program will operate it according to the operators manual and its design parameters and the assurance that the program will be operated exclusively by volunteer staff and never turned over to inmates to operate as has happened far too often in the past.

Jerry A. Marzinsky M.Ed., C.P.C.

**PROGRAM OPERATING ENVELOPE
CRITICAL OPERATING PROCEDURES YOU WILL BE EXPECTED TO
ADHERE TO IF YOU DOWNLOAD THIS PROGRAM**

**IF YOU ARE DOWNLOADING THIS SYSTEM FOR FREE, YOU ARE
EXPECTED TO SIGN A CONTRACT WITH THE PATRICK WEB SITE
THAT YOU WILL MAINTAIN EACH OF THE FOLLOWING
OPERATIONAL PARAMETERS.**

There is no computer program which cannot be broken into by a knowledgeable enough computer hacker with enough time.

Unless staff are present, testing labs should be locked and secured when the room is not in use and inmates should never be allowed into the testing lab without staff being present.

During down times, computers charged with providing programs should be locked in a secure setting which does not permit inmate access. The following list of requirements is intended to prevent the compromise of these systems. Each has proved to be a

problem in actual operation and they are not listed lightly.

Violation of these stipulations has led to compromised systems at different prisons.

This system is a tool, a very powerful tool, but it will not operate any better than the craftsman who uses it. It is highly recommend that within correctional settings, *volunteers* be trained and utilized to operate Computer Assisted Programs. When non-motivated staff who are not interested in inmate programs have been order to monitor a lab, a whole plethora of problems resulted. These people are not going to make your best pilots. Although the operation of a valid program does not require much time, it does require desire and competence. Program managers should be fully knowledgeable about program operation and able to tell at a glance whether or not the system is being operated as intended by glancing at the monitor. This is not difficult since the system operates in a closed testing loop once set into operation. The number of screens available for the inmate student is very limited. The rules listed below have been learned through the college of hard knocks and direct experience and must be adhered to in order to maintain program validity.

If credit is offered to inmates for any of the programs being completed, their efforts to break into the system will be greater. If the following guidelines are adhered to, your system will remain secure even after years of operating within correctional environments.

1. PROGRAM MANAGERS SHOULD ALWAYS BE ABLE TO CLEARLY SEE THE COMPUTER SCREENS DURING PROGRAM OPERATION AT ALL TIMES.

Inmates should never be permitted to maneuver computer monitors into a position where staff monitors cannot clearly see the screens. This is a common maneuver by inmates who then try to crack the system, copy test questions or otherwise try to compromise the system. With little practice, staff will be able to assess with a split second glance whether or not the program screen in front of the inmate is legitimate. Above the screen for testing is the name of the inmate who should be on the machine. If this name does not match the inmate ID, then another inmate is taking a test for one who has not studied and is paying the testing inmate off. NEVER ALLOW INMATES TO TURN THE COMPUTER MONITORS TO A POSITION WHERE THEY CANNOT BE CLEARLY SEEN BY THE PROGRAM MANAGER.

The locked testing loop feature is designed to block any input or shutdown of the computer from the keyboard, other than the few responses necessary for the student to take a test. This means that if a break-in is to be attempted during the operation of a lab, the student or inmate is going to have to physically shut down the computer and bring it back up again with another operating system. If a computer screen goes blank for ANY reason during a lab testing operation, this is a security violation. Any such

unauthorized shutdown should be quickly investigated by the program manager.

Sanctions should be taken against any inmate who brings down a computer for any reason.

The software program itself is set up to sound an alarm at points where security might be compromised if the program manager is paying attention. An alarm will sound if an incorrect password is fed into the machine more than three times. This alerts the program manager to potential illicit activity on the computer.

In addition, each time an inmate finishes a test, the program will sound a six second alarm indicating that the inmate no longer has any business on the machine. Inmates should be told to leave the computer as they hear this alarm. Their test score will be flashed up on the screen simultaneously. They have no further business on the computer past that point.

By notifying the program manager of an inmates test completion, the computer leaves further decisions about that particular computer's use to the manager and not the inmate. Inmates who fail tests if permitted will sign on several times trying to guess their way through a test blocking the use of the computer for legitimate students who have studied and need to test. They don't care if they block the flow through an entire lab. It is the program manager that must make sure such inmates are moved off testing machines to make room for the next student to test. Inmate clerks cannot do this as they are subject to pressure from the inmate population and gangsters who will manipulate them or threaten them. The bottom line is that if lab management is left up to an inmate, the lab will not operate as designed and will be compromised within a short time. If the completion alarm sounds and the inmate remains seated at the computer, the program manager should ask him to move away.

TESTING COMPUTERS SHOULD ALWAYS BE MONITORED BY STAFF AT ALL TIMES DURING TESTING OPERATION.

2. THIS PROGRAM SHOULD NEVER BE OPERATED FROM A FLOPPY DRIVE.

The program should be loaded onto a hard drive and operated from that hard drive. If possible, the execution files should be hidden using the DOS attributes.

This is not a necessity for the program to operate, but is an added security measure. There are a number of reasons that operating this program from a floppy drive is not desirable. A floppy can be easily stolen from a disk drive and along with it will go your testing program. If an inmate gets hold of an unsecured disk, your program is toast.

In addition, running this testing software from a floppy drive is very slow and inefficient in addition to being a grave security risk. Since a number of these programs are being operated on old, inexpensive 8088 machines, it could take a painfully long time to move around between different facets of program operation.

3. INMATES SHOULD NEVER BE PERMITTED TO ENTER A TESTING LAB WITH ANY KIND OF COMPUTER DISK.

In most institutions, inmates are not permitted to have computer disks. When an inmate shows up for testing they should have nothing with them except their study booklet which they should turn in before entering the lab. Inmates who enter a lab with a computer disk pose a serious security risk to program operation and validity.

4. IF NEWER COMPUTERS ARE PURCHASED TO OPERATE THIS PROGRAM, THEN PASSWORD THE CMOS. THIS WILL ASSURE THAT INMATES CANNOT ENTER THE COMPUTER WITHOUT THE APPROPRIATE PASSWORD.

Older 8088 computers are very inexpensive and can often be purchased for as little as \$100 each or less. These older model computers do not permit the user to password the CMOS. This makes these older models more vulnerable to inmate hackers. If however you are getting newer computers from surplus, you should take the extra step of setting a password on the CMOS. Passwording CMOS will prevent unauthorized users without the password from starting up the computer. As a consequence, computers with password protected CMOS features are much more secure during periods of inactivity. It would take a computer technician with tools and the time to open the back of the computer to manually reset the CMOS and break into the computer. Even if they could accomplish this, the tests are still encrypted.

5. UNLESS USED BY STAFF AND CLOSELY MONITORED DURING OPERATION, EACH COMPUTER SHOULD BE STAND ALONE AND CONTAIN NO OTHER SOFTWARE THAN THE TESTING PROGRAM.

In short, free standing testing computers should be dedicated to testing alone and have no other function as far as inmates are concerned. Inmate clerks have made excuses to run other types of software on testing computers, most notably, word processors. While using the word processor, they have utilized the opportunity to attempt to break into a testing computer.

Although staff may want to install programs and use the machines for other purposes, inmates can and will then utilize the opportunity to attempt to break into the testing machines. Most computer break ins in the past were the result of inmates being permitted to use a testing computers to do clerical types of work such as formulating

store lists, or word processing. If inmates know that there are other types of programs installed on testing computers, they will make a play to access that other software and have brought down programs more than once utilizing this tactic.

6. STUDENTS SHOULD NEVER BE ALLOWED TO APPROACH A TESTING STATION FOR THE PURPOSE OF TAKING A TEST WITH BOOKS, NOTEBOOKS, PACKAGES OR ANY OTHER SUCH MATERIALS IN WHICH COMPUTER DISKS COULD BE HIDDEN.

They should surrender these materials prior to entering the testing room. They should have nothing in their possession when they sit down at a computer. Since tests are administered electronically, requiring no paper or writing implements, inmates have no need to have anything at all in their possession other than their ID when they go to take a test. The inmate's ID should always be checked against the inmate number at the top of the testing screen.

7. STUDENTS SHOULD DISPLAY A PICTURE I.D. PRIOR TO STARTING A TEST. THE STUDENT ID SHOULD MATCH IDENTIFICATION NUMBERS PROVIDED BY THE COMPUTER AT THE UPPER TOP LEFT CORNER OF THE MONITOR.

The identification number on the ID should match the student number on the screen, and the picture on the ID should match the students face. If program managers are not on the ball, one of the ways inmates commonly can cheat this program is to pay a confederate to take the computer administered test for them. The confederate signs onto the computer with another inmate ID and takes the tests for him. The test results are then recorded under the scammer's ID since the computer has no way to actually check on which inmate is signing in to take a test. Although we have had inmate clerks schedule other inmates for testing and usher them onto the machines, a staff member needs to check identification to make sure the screen matches the inmate who is actually sitting at the computer. If this is not done, you may have a rip roaring new inmate business on your hands featuring a whole lineup of confederate test takers being paid off with everything from candy and soda to drugs and cigarettes. This would be especially true if the programs you offer are for credit within the correctional system. Inmates will attempt take over these systems very quickly if they see any profit at all for doing so. It is critical that they be operated validly or all you are doing is reinforcing inmates for criminal activity. Unfortunately, many correctional staff don't care. They only want to give the impression that inmates are receiving services valid or not.

8. MANAGEMENT OF SECURITY PASSWORDS FOR COMPUTERS

All security passwords for program access should be punched into the computer before students or inmates are allowed into the testing room. Computers should all be brought

up to operational status before any inmate clerk charged with collecting and distributing study booklets are permitted into the lab. No inmate should be present when encrypted passwords are being fed into the computers. Inmates are notorious for being able to read upside down documents from distances that would be almost impossible for the average person. If inmates do manage to gain access to your testing room and have your passwords they will be able to obtain and sell your tests. Security passwords should be changed periodically. This program provides a feature for doing that. Access passwords should never be written down on anything nor should any word in a dictionary be used. There are all too many inmates who will try to crack this system just for the challenge and notoriety of doing so. Others will want to turn the sale of tests into a business. One thing you can count on, is that there will be a steady flow of inmates constantly challenging your system as long as it is being operated within a correctional environment. You can count on this. It will never stop. If you leave a crack in your operational security, you can bet that inmates can and will take advantage of it. They have all the time in the world to exploit any mistake you make and they will.

9. DO NOT SWITCH TO A DIFFERENT TESTING PROGRAM WHILE THE LAB IS IN OPERATION WITH INMATES PRESENT.

With the automated menu which comes with this system, it is possible to switch from testing one program to testing an entirely different program in a matter of a few seconds. For example, you can switch from testing the Alcohol Education Program to the Parenting Education program by feeding into the computer a series of encrypted passwords. In order to do this, the first program has to be closed out and another brought up through the menu. To do this requires that a series of passwords be fed into the computer. If this switch is made while inmates are actively using the lab, you can be assured that they will be watching to scarf up any password which is typed in while they are in the area. In order to prevent inmates from obtaining passwords in this manner, the program operator should not be switching from one program to another while students are in the area. If you are running more than one testing computer you should set them to test different programs before any inmate is permitted into the lab. For example you might want to set up two of the computers to test Alcohol Education, two to test Drug Education and another two to test Parenting or Aids Education. There should be no changing of programs while inmates are present. If you only have one computer and the switching of programs is absolutely necessary, all inmates should be required to leave the testing room or office to eliminate the probability of them viewing the keyboard as you type in the passwords. Careless or lazy correctional staff have compromised more than one otherwise valid and prolific program by ignoring this rule.

If you are limited to the number of computers you are able to obtain, different sets of programs could be utilized to run different programs on different days of the week. Say if you only have three computers, all of them could be set to test over Alcohol Education for an hour or so on Monday. The same three computers could be set to test over

Parenting for an hour on Tuesday etc. once again eliminating the necessity to type in passwords while inmates are present. Inmates will always be watching for an opportunity to break into these systems. They MUST be run strictly according to these stipulated operational parameters or these programs will be worse than useless, they will serve to reinforce criminal behavior on an unprecedented scale. We can't emphasize this strongly enough, and the main source of compromise is lazy, incompetent or non-caring correctional staff. Again, these programs should only be run by dedicated volunteers who agree to operate these systems in accordance with these operational parameters.

10. NEVER PERMIT AN INMATE CLERK TO ASSUME RESPONSIBILITY FOR ANY LAB OPERATION OTHER THAN COLLECTING AND HANDING OUT STUDY BOOKLETS THAT SERVE AS TICKETS FOR ADMISSION TO TESTING.

This would seem to be common sense, but unfortunately for a large number of correctional case managers, it isn't. More than a few lazy, incompetent, non-motivated correctional staff lab operators have found it easier to turn the operation of their testing lab over to an inmate clerk who they maintained they trusted. THERE ARE NO INMATE CLERKS THAT CAN BE TRUSTED TO OPERATE THESE SYSTEMS. THEY ARE ALL SUBJECT TO THREATS AND PRESSURE FROM THE INMATE POPULATION. The results were that the inmate clerks had at different institutions either themselves broke into computers and messed up the software or permitted one of their cronies to do the same. When given access to the programming of the tests by some of the unbelievably incompetent morons who had run this program, inmate clerks broke into the edit the test question section of the program and would put a period next to the right answer, a mark almost unnoticeable. Randomization of questions and answers were completely defeated and inmates were selling the secret on the yard for quite a profit. Never, under any circumstances allow an inmate or a trustee to operate this program. IF there is any kind of credit given for completion of the program you are testing the effort to break in and compromise this system will be intense. Any crack in your security can and will be exploited

11. NEVER USE YOUR COMPUTER TESTING LAB AS A WAITING ROOM FOR INMATES WHO WANT TO TAKE TESTS.

The reason for this should be obvious. Inmates waiting around inside of the computer lab create too much commotion and are difficult to monitor. This only increases the difficulty of the program manager's job.

The only inmates who should be in a lab are those actually taking tests and perhaps one clerk at the door who is collecting and handing out study material. All others should be asked to wait outside and admitted one at a time as computers open up for use. In

addition, if inmates are permitted to wait inside the lab, in many cases they will try to get a look at the test questions while others are using the computers. They will attempt to write down the questions or work in teams where each memorizes one or two questions. Inmates can and will constantly be attempting to crack this system. All these mistakes have already been made for you to learn from. Inmates will bring down or compromise these labs if at all possible. It is the human operator who has to hold the line on security and no inmate can be trusted to do this for them. In all cases, if a lab is allowed to get disorderly and too many inmates are admitted, the extra noise and commotion will create a distraction for other inmates trying to test lowering their scores and increasing their frustration.

In normal lab operation, there will be a certain amount of test question leakage due to inmates remembering and carrying test questions out of the lab by memory. This method of loss can be rendered ineffective by increasing the number of test questions which are used in the testing data base for each test. This normal leakage has proven to be inconsequential and is not near as serious as permitting inmates to copy test answers off of the computer screens because of shoddy or incompetent operational procedure.

14. DO NOT ALLOW INMATES TO RETAKE A TEST THE SAME DAY AFTER THEY FAILED IT. THIS IS SOMETHING INMATES WILL BE PUSHING FOR. THEY WILL ONLY FAIL IT AGAIN AND BURN UP VALUABLE COMPUTER TIME.

We have done some research on allowing inmates to retake the same test again the same day. In most cases, they do not go and study more; they just try their luck again like playing a slot machine and end up failing the test once again. In the process they burn up valuable computer time that could be utilized for those inmates who did study and are prepared.

At the completion of each test, this system is designed to automatically reset for the next test. If the program manager is conscientious and keeps his eye on his lab, the alarm that each computer sounds when an inmate has finished testing will be heard. The manager is then responsible to moving that inmate off the computer and another inmate on.

If this audio alarm is ignored by the program manager, and the inmate is left at the computer station, the inmate will sign on again. If he has failed his test, he will attempt to take it again, pushing different test answers in an attempt to pass.

If he passed his current test, he may sign onto the next test to get a look at the questions, or take a guess at the answers in an attempt to pass. He will fail, but in the process he is hogging the computer. Most inmates don't care that others are waiting in

line to test if they can get control of a computer and not be pressured to get off when they have finished testing.

If a lab manager is lame about moving inmates off his computers when they are finished testing, the Statistical Package incorporated into the program will reflect its occurrence by reporting abnormally large numbers of failed tests. A valid program run for correctional populations and written at about a seventh grade reading level will indicate normal failure rates of about ten percent once inmates get past the first three tests. If the Statistical Package starts to display failure rates approaching pass rates, it is most likely due to the program manager not assuring that the inmate leaves the computer when he is finished testing or the lab has been turned over to an inmate clerk who is letting his friends, cronies and gangsters spend as much time as they desire testing at the expense of other students.

The exception to this rule is that there are some inmates who are more motivated than others and will want to take more than one test at a time. The testing system will easily accommodate these kinds of requests. These requests should be brought before the program manager and he or she should be aware that the inmate is planning to take more than one test at a certain sitting. We have found it sound procedure to require an inmate who is requesting to take a second test score at least a 90% on his first test. The scores obtained on second tests are often significantly lower than those obtained on first tests. If the inmate makes a 90% or better on his first test, there is a good probability that he will at least pass a second test taken at the same time. Anything below a 90% will yield an increased failure rate on second tests and may very well be a waste of time for the program manager. Again, an inmate who passes a first test with a 90% is usually going to pass a second with at least a 70%. These inmates can move through a program in half the time of the average inmate who takes one test a week.

15. GIVING INMATES CREDIT FOR TESTED PROGRAMS CHANGES THE ENTIRE BALL GAME. INMATES WILL THEN BE ESPECIALLY VIGELANT FOR A MEANS OF BREAKING INTO THE SYSTEM AND FINDING A WAY TO CHEAT.

Program managers must be aware that any "for credit" program will change not only the inmate's perception of the program, but will also drastically alter the manner in which they behave toward that program. The more credit inmates are given for the completion of any program, the more motivation they have to both take the program and to cheat on it. Many inmates will take the path of least resistance if they can find one, and if it is easier to cheat their way through a program than it is to study, this will most certainly be the path that most will choose.

We have seen that in a number of cases with paper and pencil tests that they worked harder at cheating than they did to legitimately study and take the tests. When tested

programs being offered "for any kind of system credit" program managers will have to be much more vigilant to maintain strict operational security measures. Credit can be considered anything the inmate might value, including privileges, restored visitation, hobby craft, certificates or computer records for the parole board etc. If inmates get credit for the completion of a program they will constantly make attempts to cheat if it is the path of least resistance. Unfortunately, all too many programs offered by correctional facilities at this time are extremely easy to cheat or, or have no measurable output at all, but on the surface look like they are doing something. This is especially true in group centered programs where inmates come out having learned nothing that they can put into words, but are given credit merely for showing up and sitting there with their eyes open. Inmate cheating can range from using cheat sheets to beat tests to sitting around and daydreaming in a lecture or a group in order to get credit without paying attention.

HARDWARE SECURITY CONSIDERATIONS AND OTHER OPERATIONAL CONSIDERATIONS

16. If inmates have access to floppy drives and have any kind of a computer disk, you can count on them sticking it into one of your computer drives. One method of maintaining computer security is to remove or disable your floppy disk drives once your computer is set up. This has both advantages and disadvantages. Removing the disk drives once a computer is set up for testing is probably one of the better ways of protecting your computer from hackers and break ins. Once the drives are removed, nothing can be put onto the computer, and nothing can be taken off. With the disks missing, pirated software programs cannot be used to hack at the system or leveled at the programs security features. There are however obvious drawbacks to removing the drives. It makes the installation of newly developed testing programs difficult. It also makes the back up of the tests for a program almost impossible. In addition, it negates many of the nicer features of this system which allows for immediate alteration of test questions for any one of a number of different reasons. Running your lab with computers whose disk drives have been removed is probably one of the most secure methods of operation, but it is very inconvenient and rigid negating much of the flexibility this system has built in. However, operating without disk drives may be the only way some prisons will permit these systems to operate. Operating in this manner is still entirely possible, though slightly more inconvenient.

In these cases, since floppy drives are not available to make back ups of inmate testing progress it is highly recommended that some hard copy method of tracking inmate progress be developed in case of hard drive failure. The program permits downloads of inmate scores if the floppy drives are operational. If the lab is monitored adequately and competently, inmates will not attempt to bring floppy drives to bear. If the monitoring is sloppy and they have a disk, the program is at risk.

17. Unless operated from a monitored LAN system, each program dedicated to testing should be operated on a stand alone basis with no other program installed onto the hard drive other than DOS. All extraneous DOS files should be removed and the system operated only with the boot record files. For security purposes, all other DOS utility files should be removed. In earlier versions of DOS, (prior to DOS 6.0) these three files consisted of IO.SYS, MSDOS.SYS, and COMMAND.COM. These are the only DOS files necessary for this system to operate. These are the only system files which should be on a testing computer using DOS versions below 6.0. For versions of DOS 6.0 and above you will need all of the above mentioned files, plus DOUBLE SPACE. The file name for DOUBLE SPACE is dblspace.bin. So in the case of DOS operating systems versions 6.0 and above, four boot record files are needed. I've been told that these operating systems are now so obsolete that they can be used for free if you can find them.

Any of the additional utility DOS files placed onto the hard drive could give a knowledgeable computer hacker some of the tools needed to affect a break in to your system. Inmates must be kept out of the computers operating system at all costs. If you format an empty hard drive with a system disk, these extraneous utility files will automatically be placed onto your hard drive. The command for formatting an empty hard drive with a systems disk is FORMAT /S. This command will give you the bare bones number of files needed to operate the User Programmable Testing and Scoring System. Do not permit any utility DOS files to reside on the hard drive of your testing computers. The fewer the number of operating system files used, the more secure your machine will be.

18. Hooking up a printer to your testing computers presents obvious problems. An inmate hacker who may be able to gain access to your lab, operating system and passwords will have the means of quickly printing out your test questions. It is safer not to have a printer located in a testing lab operating with correctional populations at all. In any case, a printer should not be hooked up to a testing computer during normal operation. If the testing monitor is distracted inmates could conceivably print screen questions right from the computer screen. Any printer residing in the same physical space as your testing lab is a potential security threat.

19. If you are designing your own programs, you should have at the very least thirty or more questions in each test data storage bank. The system requires a minimum of twenty.

However, if only the minimum number of test questions is programmed into the system, the computer will present each one of those test questions each time a test is given. This means that the same questions will be given each time that test is given. The same twenty will be chosen repeatedly making it far easier for inmates to obtain the test questions by memory alone.

Although the order of both the test questions and answers will be scrambled by the computer irrespective of the number of questions you have, program validity will be much more at risk if only the minimum of twenty questions is used. On the other hand, the more test questions programmed into your system for any one test, the more secure your system will be. Once you get up to about sixty test questions, it is possible that inmates taking the same test next to each other will not receive any of the same questions. In the worse case, they will receive very few of the same test questions, and for all practical purposes although the computer is testing over the same programmed material, the tests will be very different.

20. Provisions must be made for backup of inmate test scores in event of computer failure. To keep costs down, some correctional systems are purchasing used computers from University Surplus or Military Bases. Although these older computers do a perfectly adequate job of running the testing software and are plenty fast for testing purposes, they are more prone to physical failure and parts may be difficult to find depending on how old the computers you are using are. The CAP software package comes with an automatic back-up system which will appear on the main menu. This feature will automatically back up each program at the completion of testing if used correctly. Backing up inmate test scores may save some wear and tear on your nerves in the event of computer failure. This system forces inmates to legitimately study for rehabilitative programs. In many cases they have to spend a great deal of time studying. As a result, you will find that if a computer goes down, inmates will become very irate if their test scores are lost and cannot be recovered for any reason. If you are not using the automatic back up feature which comes with this program, it is suggested that you formulate a manually operated accounting system to keep track of student scores. Inmates should never be trusted to do manual backups of scores, never.

USE YOUR STUDY BOOKLETS AS TICKETS FOR ADMISSION TO PROGRAM TESTING: At this point, we would like to recommend that program operators who are testing over study booklets or study sheets given to inmates for study in their cells, seriously consider using these as tickets for admission to testing. This serves a number of purposes. First of all, you will get your study booklets back allowing you to re-circulate them. This will go a long way toward allowing you to keep your copying costs down. Secondly, if you are using study sheets in conjunction with books or tapes, the process keeps filled out study sheets from floating around the yard and being put to illicit use. If a group or didactic program is being tested, it is best to have an attendance list available at the time of testing to assure that the inmate has actually attended the group or program and has not just copied someone else study sheet.

We hope you enjoy learning how to operate the C.A.R.P. program and helping others beat their addictions, not matter what those addictions may be.

You are doing a tremendous service for the addict and for the public as well.

REFERENCES

- (1) NBC News Broadcast on Saturday 9/25/93 at 5 PM. They report this information was cited from FBI statistics.
- (2) Bureau of Justice Statistics, Special Report, Recidivism of Prisoners Released in 1983. U.S. Department of Justice, Office of Justice Programs, Bureau of Justice Statistics.
- (3) Bureau of Justice Statistics, Survey of State Prison Inmates, 1991. March 1993, NCJ-136949.