INTRODUCTION

This paper describes a unique workshop structure based on the Montessori Method and utilizing both vertical and horizontal interaction in the training of systems programmers.

The scope of this paper covers the definition of the training need, the objectives of the workshop, the training curriculum, the selection process, the application of certain Montessori techniques in the training of programmers, the measurements for evaluating success, the results of the workshop to date, and the planned, continued upward development of programmers.

Training need

The California State Division of Highways' unsatisfactory experience with on-the-job and vendor training resulted in our decision to develop and implement our own in-house programmer training workshop. Prior to August, 1966, we depended upon on-the-job training supplemented by vendor courses.

On-the-job training was constantly interrupted by production schedules; production supervisors did not allow adequate time for training, and they were not necessarily qualified or motivated to train.

Vendor training objectives tended toward course completion and coverage of material. The courses were generalized, scheduled at inconvenient times and did not cover the total subject matter necessary to the training of a thoroughly productive trainee systems programmer.

Individual self-discipline, motivation and ambition for increased levels of responsibility did not result from these approaches toward training.

In July, 1966, we developed a curriculum for an in-house systems programmer training workshop, and applied certain training methods formulated by Maria Montessori. These methods are the right of the student to be active, to explore his environment and to develop inner resources through investigation and creative effort. The environment should be such as to give scope to the individual's inner resources, to direct these resources and most of all to call them forth. The instructor's task is one of assisting, watching, encouraging and inducing rather than interfering, prescribing or restricting.

To date, seventy-two systems programmer trainees have graduated from eight classes where these techniques have been applied.

A systems programmer trainee within our installation is one whose basic training includes concepts considerably beyond the fundamentals of applications programming. A graduate from our systems program, trainee workshop must also have a broad, basic understanding of the logic of the operating system-data management, system library functions, catalogue procedures, the sophistications and efficiencies of Job Control Language, the purpose of system control blocks, the various access methods, available utilities, and core dump analysis.

Graduates from our workshop are eligible for consideration for rotation into systems software programming after a minimum of six month's experience in applications programming. Upon rotation into the systems software programming unit, they receive additional training in operating system coding, teleprocessing access methods, system generation and details concerning the logic of the operating system under a multiple variable task environment.
The objectives of the workshop

The objectives of our programmer training workshop are:

1. The development of competent trainee systems programmers who can take immediate responsibility for complex assignments.
2. The development of individual self-discipline, professional motivation and endeavor towards reaching levels of increased responsibility such that it will continue after graduation from the workshop.
3. The assurance of a steady flow of capable personnel, qualified and trained, to fill programming positions which occur through promotions and attrition. We cannot afford to be dependent upon the indispensable programmer.
4. The programmer training workshop is only a first step in the system programmer's development; it serves to reduce attrition in the long run by providing opportunities limited only by the individual's interest and ability.

The training curriculum

The training curriculum is dynamic in nature. It is subject to revision based on our current hardware/software environment and advancements in the state-of-the-arts which have practical application within our installation.

At the present time our environment consists of a System/360 Model 65 with one million bytes of core storage, twelve tape drives, two 2314 disk storage devices, one data cell with another to be delivered in August, two high-speed printers, a drum, a Cal-Comp plotter, remote job entry with 2780's in eleven districts, six 2260 display units in the software and programming units, with additional 2260's on order for users.

We are under a full operating system with the Faster generalized file processor and are in a multiple variable task environment.

The curriculum developed for the workshop is divided into two parts. The first section covers programming concepts and the hardware and software of the system and comprises eight weeks. The remaining four weeks are devoted to a real, live production environment within the workshop in which each trainee is given a fairly complex production program to chart, code, compile and test, and which will actually be used by the installation. Each trainee is given a completely different program.

Implementation of the training workshop

Selection process

Candidates for our training workshop must pass a State version of the IBM Programmer Aptitude Test. The cut-off point is 70 percent. This test eliminates 50 percent of the candidate group. Successful candidates are further evaluated through in-depth interviews.

Questions are asked during the interview which will expose the candidate's self-image and professional goals. The candidate is asked to relate experiences which will substantiate the salient points of the self-image. The self-image and goals are then evaluated against the following behavioral criteria:

1. A strongly developed sense of personal motivation toward a long-term career in data processing.
2. A basic character that is conducive to developing a high degree of self-discipline.
3. An enthusiastic interest in working at a detailed level for long periods without frustration.
4. The ability to work under pressure to meet critical deadlines.
5. The ability to communicate.

Enrollment in our training workshop is limited to twelve trainees. This is the maximum number of students which we feel will allow us to maintain the desired flexibility of working conditions.

Unique Montessori environment

Our training environment is unique because it applies Montessori Methods to the training of systems programmers. These methods are the development of self-discipline through freedom, an environment which challenges and motivates on an individual basis and encourages individual growth toward full potential.

The workshop is structured to appear to be a free form in-group atmosphere. It is, in fact, a carefully controlled and disciplined environment where the ground rules for expected performance within the workshop emphasize the professional attitude expected in the production environment of the installation. The workshop stresses group interaction and individual self-discipline. It utilizes the unusual horizontal interaction of students training each other as well as the traditional vertical interaction of the instructor training the students.

Ground rules for expected performance

Ground rules for expected performance are outlined for the trainees.
It is emphasized that each trainee has been carefully selected for this workshop.

The workshop has been implemented to produce qualified, first level systems programmer trainees. It was not implemented to produce coders of higher level languages. It is expected that the trainees will keep this objective before them as they proceed through a difficult curriculum.

The curriculum requires an average of three hours of voluntary homework assignment each day. These homework assignments make it possible to cover an extensive amount of academic material within an eight week period, evaluate the professional motivation of the trainee, and apply pressures of motivation and self-discipline.

This workshop does not graduate a trainee who demonstrates a lack of ability or who fails to meet the expected high level of performance. The trainees are told that the expected attrition rate within the workshop is 10 to 20 percent and that there will be no exceptions to the standards of expected performance.

The graduates from this workshop are expected to be able to immediately assume responsibility for complex programming assignments with minimal supervision.

Application of the Montessori method

The application of the Montessori Method to our training workshop has been achieved in the following areas:

1. Intense self-discipline through freedom

   The trainee has been encouraged to discipline his own study habits and not be concerned with the pace at which other trainees may learn. Observation has shown that when a trainee attempts to emulate the study habits of a peer, he frustrates his natural mode of self-discipline. This results in reduced performance. When the trainee returns to his own specific form of self-discipline, his level of performance rises again.

   The trainee is required to demonstrate responsible initiative in coming to the training instructor for individual guidance and instruction. Every opportunity is given for special tutoring at a time acceptable to his schedule. He must, however, demonstrate a desire for help in achieving his goal. If the instructor wishes to remind a particular student that special tutoring is available, the reminder is directed toward the entire group; not toward the individual. This approach forces the student to reconsider the advantages of accepting help and the alternative of possible failure to pass successfully through the curriculum. It emphasizes to him the need for self-discipline through freedom of action on his part in taking the proper initiative. It is made clear to the trainee that he is responsible for properly interpreting and understanding the subject matter. Quizzes and examinations are open book to discourage memorization.

   The trainee is allowed considerable freedom in his individual ability to demonstrate self-discipline. A prime example of this freedom occurs during the study periods. The trainee is not actively monitored during these periods. He has the responsibility to be prepared to participate in the next scheduled group discussion. He cannot be a disruptive force to others who are studying. He is given the freedom to study by himself, study in a small group cluster where there is a horizontal exchange of ideas and a process of teaching each other, or he cannot study at all.

   During the study periods, the instructor will sometimes leave the environment for a half-hour or so. This is done to encourage a totally free environment for individual self-discipline on an active basis.

   The training instructor keeps a detailed, daily diary on each trainee. Particular attention is given to recording individual critical incidents. This diary serves as a reminder to the instructor during the weekly evaluation and counseling session with the student. It also is used to substantiate an outstanding performance report or as detailed documentation required for recommending rejection from the workshop.

2. Challenging and motivating the individual

   The trainee is given increasingly difficult assignments. He is told that the curriculum covers in-depth systems programming concepts. On the first day of the workshop he is given some thirty-five technical manuals ranging from basic programming through details concerning the internal logic of the operating system. He is told that he will be expected to understand and apply the technical subject matter covered within these manuals and that he has eight weeks to reach these expectations. The intense impact of receiving all thirty-five manuals at one time has the effect of challenging and motivating the trainee toward full achievement of these goals.

   The trainee is told that upon leaving the workshop environment he will be expected to be
immediately productive and contribute to the upgrading of the installation. His training has prepared him to move in several directions of individual development. Within a relatively short period of six to nine months he will be expected to become a responsible programmer of large Management Information Systems applications, a fully qualified software programmer responsible for the operating system or a technical specialist in teleprocessing, generalized file processors such as Gentry or Mark IV, Graphics or any other advancements in the state-of-the-art that the installation implements.

The trainee is encouraged to challenge his environment at any point. He is asked to become an experimentalist—to not be afraid to try a technical innovation which has not yet been implemented within the installation but which seems to be a practical approach and is proven to be theoretically possible. He is encouraged to be an innovationalist—to not be inhibited in making suggestions for improvement. The philosophy of the workshop is that the newest employee can possibly make a major contribution to progress within the installation.

3. Special emphasis on individuality

The workshop encourages the trainee to have a deep curiosity for exploring beyond the subject matter covered within the curriculum. The trainee is encouraged by his own motivation to do individual assignments beyond those required by the curriculum and to report back to the group. Examples of such individual assignments have been the internal logic manuals on the Compiler, Linkage Editor and Input/Output Control System. The IBM Systems Journal, Volume Three, Numbers Two and Three, 1964, with A Formal Description of the System/360 by Dr. Kenneth Iverson, has been given as a special reading assignment to several trainees who demonstrated interest and the ability to absorb the material. One trainee, especially qualified by his background, was encouraged to give a session to the group on vector analysis.

This initial interest on the part of the trainee has usually resulted in identifying a direction for continued self-development which has been beneficial to himself and to the installation.

Horizontal and vertical interactions

The physical environment of the workshop is made conducive to horizontal interaction in the exchange of ideas among the trainees. The desks are placed in a conference arrangement so that the students face each other. One desk in the group is reserved for the instructor, who, while sitting at this desk, enters the group involvement on an active basis and in fact becomes a member of the group.

The discussion periods are a complex environment utilizing the Montessori Methods of individual self-discipline, motivation and endeavor. Superimposed is a group dynamic structure which is best emphasized by the fact that a cooperative rather than competitive attitude is developed among the group. Group motivation is stressed.

At times the group actively takes over and runs its own training. It is they who select a peer to go up to the blackboard and summarize the key points of their discussion; and it is they who monitor their own horizontal interaction. At such times the instructor remains outside the group and does not become actively involved unless the group loses control over the discussion.

At other times the vertical interaction is evident between the instructor and individual students or between the instructor and the group as a whole. The instructor may have an open discussion with one member of the group. He may by his singular attitude indicate to the rest of the class that he does not want any group interference in this vertical interaction with a selected member from the group.

The group may "go critical" with respect to vertical interaction with the instructor. There has been in every workshop so far a crisis point about half way through the curriculum where the group attempts to resist learning any more, applying any further self-discipline and individual motivation. This critical period is usually of short duration, about a day, and marks the point at which the group learning can go in two directions. The whole situation can fall on its own weight and from then on only desultory learning and results can be expected. Or, the environment—people combination can go "critical" and a kind of self-sustaining group reaction is initiated which releases enough motivational and learning energy to carry the group through the increasing but controlled pressures of the workshop. This occurs primarily because of the unifying force of the instructor's own intense enthusiasm and motivation. In all eight workshops so far the students have reacted with greater self-discipline to the challenge of completing the course and becoming productive systems programmer trainees.

Measurements for evaluating success

The following measurements have been applied in evaluating the success of our workshop:
Measurements within the workshop environment

1. Observation of individual motivation and self-discipline in the mastery of the subject matter. These observations are recorded in the daily diary of critical incidents and are periodically evaluated by the training instructor.

2. Successful development to the level of systems programmer trainee.

3. Performance in written quizzes, examinations and case studies.

4. Evaluation of the trainee's ability to relate academic subject matter to the first production assignment.

Long term measurements

The long term measurements of the individual are:

1. A continued cooperative attitude among the peers working in a production unit.

2. Continued motivation toward self-development. Each employee has an individual Employee Development Plan which is kept current by his production supervisor.

3. The ability to handle complex systems assignments in a timely and efficient manner and to meet critical deadlines under pressure with few compilations and tests.

4. Self-discipline which results in minimal supervision and is demonstrated by adherence to standards and guidelines of the installation. This measurement can best be taken in terms of adherence to the requirement of complete documentation of a given system.

5. The continued advancement to ever increasing areas of responsibility within the installation and success in passing examinations for higher work classifications.

6. The periodic, written evaluation of the individual's performance by the production supervisor.

Results of our training workshop

The results of our training workshop for systems programmers has been twofold. There has been an immediate upgrading of programming performance and we have identified a strong need for individual development and continued training for all 209 persons employed within our installation.

The upgrading of programming performance by graduated trainees has been demonstrated in an outstanding manner. The trainees have continued to demonstrate a high motivation toward long term professional careers. This motivation is partly due to basic personality characteristics which were carefully selected during the in-depth interview and to the fact that the training workshop strengthened this natural motivation.

The supervisors have observed that the expected level of accuracy and self-discipline is being maintained. These programmers require minimal supervision.

Sophisticated production assignments normally given to experienced programmers have been given to the systems programmer trainees soon after graduation from the workshop. These are often large scale, complex programs involving 50 to 100K core storage, intricate relational editing techniques, table searches, complex mathematical calculations and dense reporting formats. Our applications involve the construction of large information systems within the areas of traffic control, urban planning, fiscal management, administrative support and engineering project control as well as complex programming in the areas of bridge, vertical alignment, traverse and earthwork. Graduates from our trainee workshop are expected to become immediately productive in these areas. Those who rotate into software programming have responsibility for the operating system, the writing of all in-house utilities and subroutines as well as being responsible for maintaining an in-depth understanding of remote job entry and teleprocessing capabilities and the software capabilities of any large generalized file processors currently in use.

The programs written by these former trainees have been logically planned using truth tables and modular design and are fully documented. The number of compilations and tests has been greatly reduced.

These graduates from the training workshop have consistently demonstrated competence and understanding of the complex capabilities of the system. They have been articulate in expressing knowledge of the system and they have demonstrated ability to explore and implement those areas of software which result in the saving of machine time.

A typical example of such a contribution can be illustrated in the fact that a graduate with less than six months of production programming experience researched the feasibility, recommended and then implemented the conversion of a large sequential data base to indexed sequential access at a time when ISAM was a relatively new and untried quantity.

These systems programmers have demonstrated ability to maintain and modify extremely large and
complex systems and the ability to consistently produce under critical deadlines.

Their continued interest in the field of data processing has been evidenced by their active participation in professional associations and their enthusiastic attendance at seminars and symposiums which have covered advancements in the state-of-the-arts. They have continued to demonstrate highly developed study habits which have resulted in their being thoroughly familiar with the subject matter presented in new manuals and publications. All of the former trainees have clearly defined goals for their own long term professional development. Of the seventy-two systems programmers who have graduated from the training workshop, approximately two thirds have returned to take evening courses at college toward a higher degree.

Four new programming units have been created entirely from graduates of the training workshop. These production units have maintained highly satisfactory levels of programming performance.

Ten programmers, who received their training in the workshop, have become part-time instructors in advanced techniques sessions and workshops for programmers.

Ten of the graduates from the training workshop are now highly qualified software specialists responsible for all aspects of maintaining our operating system and implementing teleprocessing, partitioning under multiprogramming, Faster and the system utilities.

The lead systems programmers in every production unit have graduated from our training workshop within the last year and a half. Two graduates have underfilled temporarily as supervisors of programming units.

Attrition has been noticeably reduced since the implementation of the training workshop from 27 to 14 percent in the first year after implementation of the first workshop. At the present time our attrition is so reduced that we do not anticipate the need for an entry level systems programmer trainee workshop before July, 1969.

The areas of continued training and self-development have been identified in the following ways:

1. The supervisors and experienced programmers have requested continued training for themselves and these training programs are currently being developed for them.
2. We also have an individualized employee development plan which includes every member of our staff and involves 209 employees in 41 computer systems classifications. The basis of this plan is: (a) A determination of skills and knowledges required and desired for each and every position and (b) each employee's background of education and experience. The differences between these two elements becomes a base for our individualized development plan.

This is not a "one-shot" or lightly considered plan. The dynamic nature of both computer technology and information usage necessitates continuing review, updating, and planning. In effect this plan constitutes a contract with each employee outlining management responsibility and the employee's responsibility for his own self-development. The plan may include anything from college courses to management development, depending upon a mutual agreement as to need.

3. An up-to-date library of current periodicals, journals, and books is being instituted by our management.
4. The need for a full-time training program for Computer Systems has been identified. We have developed and implemented a training workshop for systems analysts utilizing the same involutional group dynamic structure training techniques. To date, four such systems analyst training workshops have been implemented since September, 1968. We are currently developing several workshops for programmers with 12-18 months and 18-24 months experience as well as a series of workshops for computer operators.

CONCLUSION
Our efforts in systems programmer training have had immediate and long-range results. We have achieved the level and quality of programming skills necessary for successful operation in our third generation environment. We have recognized the need for both initial and continued training.

We have recognized immediate results from the application of the Montessori Methods emphasizing individual self-discipline through freedom, individual motivation and endeavor towards reaching levels of increased responsibility. Our present and future workshops will continue to utilize these training techniques which are emphasized by the Montessori Method of teaching.

A formal individual development for each of our 209 employees has been implemented. Most important, management has recognized that the development of personnel on a planned, continuous basis is the key
Montessori Techniques in Workshop Environment

...to successful achievement of our long-range operating goals.

REFERENCES

1. M. MONTESSORI
   * The Montessori method
     Second Edition Frederick A Stokes Co New York 1912

2. E. R. ALEXANDER
   Third generation programmer training—the workshop

3. A. D. FALKOFF & K. E. IVerson
   A formal description of systems/360
   IBM Systems Journal Vol 3 No 2 and 3 1964

4. E. R. ALEXANDER
   A working measured development plan for computer personnel
   Proc of the Sixth Annual Computer Personnel Research Conference 1968