DOCUMENT RESUME

ED 112 889 IR 002 600

AUTHOR McClain, Donald H.; And Others
TITLE A Computer-Assisted Patient Simulation.
INSTITUTION Iowa Univ., Iowa City. Computer Center.
PUB DATE 4 Aug 75
NOTE 10p.; Paper presented at the Association for Development of Computer-Based Instructional Systems Summer Meeting (Portland, Maine, August 4-7, 1975)

EDRS PRICE MF-$0.76 HC-$1.58 Plus Postage
DESCRIPTORS *Computer Programs; Decision Making; Higher Education; *Information Processing; Interaction; *Medical Education; Medical Students; Problem Solving; *Simulation
IDENTIFIERS Computer Assisted Patient Simulation

ABSTRACT

A set of computer programs and data files have been designed to provide a simulated experience with patient-diagnosis problems for medical students. The programs, called CAPS (Computer Assisted Patient Simulation), aims to provide opportunities for making diagnoses and decisions; develop problem-solving skills; expand knowledge of specific disease entities and hospital resources useful in differentiating between them; teach cost-effective utilization of a pathology laboratory. The design of student interaction and the program and file structure are explained in detail, and a comparison made with the CASE system. In trials with 175 students, it was concluded that the system was a feasible approach to helping students in gathering and analyzing pertinent information, consulting supplementary materials to gain further knowledge, and arriving at a definite diagnosis and treatment plan. (SK)

ED112889

A Computer-Assisted Patient Simulation

Donald H. McClain
Thomas H. Kent
Scott C. Ludwig
Stephen W. Wessels

Introduction

Members of the University of Iowa Computer Center and the Department of Pathology of the College of Medicine have designed and implemented a set of computer programs and data files entitled CAPS (A Computer-Assisted Patient Simulation) which provides a simulated experience with patient-diagnosis problems to medical students. These computer based simulations of the physician-patient encounter are a learning exercise designed to: (1) provide opportunities for making diagnoses and decisions; (2) develop problem-solving skills; (3) expand knowledge of specific disease entities and the hospital resources useful and necessary in differentiating between these entities; and (4) teach cost-effective utilization of a pathology laboratory. The CAPS system offers the user, usually a medical student, the opportunity to apply acquired knowledge and practice learned techniques to solve various simulated, yet authentic, problems in differential diagnosis. (The term student as it appears throughout this paper also refers to an intern or a practicing physician concerned with continuing his/her education.)

Simulation

Simulation is a technique which has recently gained considerable attention and utilization in education as a strategy for teaching, eliciting, and evaluating higher-level problem-solving skills while providing realism, richness, and complexity. This technique has been described by Bobula and Page (1) as "placing an individual in a realistic setting where he is confronted by a problematic situation that requires a sequence of inquiries, decisions, and actions. Each of these activities triggers appropriate feedback which may modify the situation and be used for subsequent decisions about what to do next. The examinee's next action in turn may further modify the problem. Thus, a problem evolves through many stages until it is terminated when the individual reaches an acceptable solution or is faced by the unacceptable consequences brought about by his own choices and actions."

Because of: (1) the need to provide students with practice in sequential information gathering, diagnosis, and decision making and (2) the limitations involved in using actual medical patients (e.g., student time involved, availability of patients with specific diseases at the appropriate time, welfare of patients, staff time, etc.) simulations appear to be a promising technique for the presentation of patient-management problems. McGuire and associates to the University of Illinois' Center for Education Development at Chicago as referenced by Bobula and Page (1) have developed a model for the design of "paper-and-pencil" simulations utilizing a latent image method. Many clinical simulations for both instruction and evaluation have been implemented using this model. In another project also at the Center for Educational Development, Harless, Drennon, and co-workers (4) employed and computer as a media to provide simulations of the clinical encounter. Their system, termed CASE (Computer-Aided Simulation of the Clinical Encounter), permits information gathering in a natural language format. In addition to this study and CAPS here at Iowa, computer-based clinical simulations have also been developed at Massachusetts General Hospital in Boston, Massachusetts (5), and the State University of New York in New York City (3).

Design of a Student's Interaction

The student is assigned a patient by receiving a short printed history (in simulation terminology sometimes called "the setting"), a set of work sheets for the accumulation of further data concerning the patient, and a list of 66 possible diagnoses and 35 treatments. The student's task is to gather additional information about the patient until he/she feels confident to make a diagnosis and initiate treatment. This information regarding the patient's condition is attainable concerning: (1) past medical history; (2) physical examinations; and (3) laboratory tests or clinical procedures. To save time and avoid typing errors, a three-character (one letter and two-digits) code is associated with each of the pieces of information, and this code must be specified in requesting desired material from the computer. In the information-gathering process, the medical student is encouraged to alternately use reading materials, suggested algorithms, consultations, and the computer to gather and analyze information. Furthermore, all photographs or pertinent biopsy specimens and x-rays are available and referenced to appropriate cases.

Realism of hospital practices is built into the simulation on the following manner: (1) Information is withheld from the student for the usual number of hospital days required to actually complete the analysis. (Historical and physical examination data are available immediately.) For example, if a biopsy is ordered by a student, he/she is directed to the appropriate histologic slide the following day and receives a written pathologist's report one additional day later. However, a simulated hospital day is not necessarily equivalent in time elapsed to a 24-hour day. Its exact length is under the user's control. Depending upon the student's time schedule and his/her desire to see a test result, a simulated hospital day might be as short as a few minutes or as long as several normal 24-hour days. But ineffective utilization of the laboratories caused by premature ending of simulated hospital days results in higher hospital costs for the patient. (2) Certain test and procedures which are incompatible cannot be scheduled during the same time period. For example, two tests involving radioisotopic procedures cannot be ordered on the same day. (3) A patient is charged current University of Iowa Hospital rates for both hospital room and laboratory tests. The room charge is calculated according to the total number of simulated days of the stay. (4) Laboratory results are varied according to a normal distribution determined from actual patient data. This means that if a patient's hemoglobin is requested on two separate days the two values obtained would most likely be different. Also, it means that students who request the same tests will be given different values distributed over a normal curve with a predetermined mean and standard deviation.

Upon completing the stages of acquiring information, applying previous classroom knowledge to the case, and examining relationships, a medical student enters his/her diagnosis and suggested treatment of the problem into the computer. The student does not have to formulate a statement of this diagnosis or treatment but simply selects the appropriate codes from provided lists. A pre-stated diagnosis is utilized to make the student's reply more definitive. Also, if so desired, additional diagnostic statements may be entered by the student for the case. Student analyses are compared with ideal approaches, derived from the conensus of a group of faculty members, and are judged on the basis of correctness of diagnosis, cost of hospital stay and of tests and procedures performed, and ordering of recommended tests and procedures.

Prior to receiving feedback on the correctness of his/her diagnosis and treatment, the student is given a short questionnaire to access attitudes toward the CAPS concept. The student is asked to rate his/her interest in CAPS and the difficulty of the case. He/she is also requested to evaluate the degree to which the simulation improved knowledge of labcratory tests, diagnostic skills, and understanding of subject matter.

Actual student interaction with the computer is kept intentionally brief, with the computer being employed to control and vary the information presented to the medical student and to record student decisions. At the same time, the student is encouraged to make use of resources in addition to the computer.

CAPS Programs and File Structure

The CAPS system is implemented on a Hewlett-Packard (HP) 2000F minicomputer utilizing BASIC programs and HP's Course Writing Facility (CWF), the latter a sequel to IBM's Coursewriter

III language. CWF was engaged primarily as an aid in the record-keeping function of the simulation by providing student course facilities (i.e., counters, buffers, switches, parameters, registers, and auxiliary storage) and sign-on return points. Also, in conjunction with HP's Instructional Management Facility (IMF), CWF manges student registrations and the sign-on/sign-off procedures. Because a great deal of the CAPS interaction involves information retrieval and file manipulation, the bulk of the CAPS system is comprised of BASIC language programs.

Two files which hold data concerning past medical history, physical examinations, and laboratory tests are associated with each patient case. One file defines normal results and the other file, abnormal results. The normal file, the larger of the two, is capable of being employed with many cases whereas the abnormal file pertains exclusively to a particular patient case and contains only data which is different from the normal values. The normal file consists of 39 standard history items, 13 physical examination categories, and 117 remotely relevant laboratory tests and clinical procedures available at University Hospital. The number of items defined for each abnormal file averages less than 35.

Case names are of the form "A11" where "A" is a letter and "11" is a two-digit number. The letter of the case name designates the name of the normal file (e.g., M for male, F for female, C for child, E for elderly person, etc.) and the number indicates which abnormal file is to be used for the particular patient. Currently, the system is comprised of two normal files (male and female) and eleven abnormal files. However, twenty-two additional abnormal files are about to be entered into the system, and data for a normal child's file is being compiled.

For each patient case, the abnormal file is run against the normal one in that the simulation program searches the abnormal file first for the desired result. If the result is not found, the program proceeds to examine the normal file. This procedure allows the easy proliferation of cases by merely defining multiple abnormal files to be run against any specific normal file. In fact, large numbers of new patient simulations can be easily created by using actual patients' hospital charts once a normal file is established.

A primary concern in the design of the simulation programs and the file structure is the program's ability to rapidly locate an appropriate result. Because input/output (I/O accesses degrade response time, the file is organized to minimize I/O accesses. This is accomplished by setting up a directory of the location of information within a file as the first three file records. The directory is comprised of a list of the letters of the results defined in the file as well as a pointer to the array which holds the file record number containing information pertaining to a particular result. During a student's session on the simulation, this directory is stored in memory. Thus, to find a requested piece of information, only one I/O access is necessary, preceded by a DO-loop to ascertain the appropriate letter and one calculation to locate the element of the array containing the file record number.

The file organization for the normal and abnormal files is identical except that the correct diagnosis and treatment for the patient constitute the fourth and fifth records of the abnormal files. Information for student access, with one result per record, makes up the remaining records of the files. For every result, the record has values for each or all (values appropriate or meaningful for the particular result) of the following data items: (1) name or descriptive statement of item; (2) time factor (number of days before information is available; if the number is zero, information is available immediately); (3) cost of test in dollars; (4) conditional option (allows student to withdraw order); (5) type of test (generator or regular--discussed in following paragraph); (6) test generated (a list of other tests to be included as part of current test); (7) test blocked (incompatible procedures prohibited for a specified number of days pending completion of test in progress); (8) numerical result (mean, standard deviation); (9) units for numerical results; and (10) written results.

Referred to in the preceding paragraph, regular tests provide the student with a static written statement and/or a computer-generated numerical value. The computer-generated numerical value is a normal random number selected from a normal distribution with a pre-specified mean and standard deviation. The normal random number is obtained by using Mueller's algorithm (6) and HP's uniform random number generator. A generator item merely consists of a group of legitimate tests or procedures which have a commonality. Each test can be ordered separately or all tests can be ordered as a package under the generator concept. The values for time required, cost, and any blocked tests are taken from the generator item itself rather than from the individual tests. For example, an SMA 6/60 test is composed of the results of the following six tests: sodium, potassium, chloride, CO_2 content, Urea-N, and creatinine. The cost of the SMA 6/60 test is $12 whereas the charge for six tests individually is $36 or $6 per test. The values produced either by the generator or the six individual items are equivalent, at least theoretically.

Computer programs to construct the files are written so as to prompt users for pertinent information during the building of a file. Because they are not stored in a sequential manner, the data items can be entered into the CAPS system in any sequence. However, in response to a program query, the user does establish limits as to the number of different letters possible and the maximum number of entries within each letter category. As the information for a result is incorporated into the file, the file directory is automatically updated to reflect the addition. Furthermore, the program allows the lists of data items to be expanded or changed for all patients.

During interaction with the simulation, the program presents a main option point to the student which permits he/she to select one of five options. The five option points are to: (1) select a desired piece of information; (2) conclude the current hospital day; (3) list all pending tests; (4) terminate the case by specifying a diagnosis and treatment; and (5) sign off the computer. After completing the option desired, except for number four, the student is recycled back to the main option point.

Initiating option one, the student responds to the program by entering the appropriate code for the desired piece of information. The program searches the blocked test list to assure that the results has not been blocked by a previous request. If not blocked, and depending upon the particular item, the program either displays the result to the student immediately or adds the test to the list of tests pending. The student continues ordering information until he/she wishes to return to the main option point, which is accomplished by merely pushing the carriage return mechanism on the terminal.

Option two is employed to begin a new simulated hospital day. When this option is specified, the computer program does two things: (1) removes any tests from the blocked test list which are no longer incompatible and (2) displays the results of any test which had been pending but is now available as of this new day. Both the blocked-test list and tests-pending list are updated and compressed by deleting tests no longer blocked or pending. The student is then branched back to the main option point.

For option three, the program displays the names of all tests which are pending completion and the day that each one will be available. Then, as before, the student is directed back to the main option point. If he/she selects option five, the student is signed off the computer and all data relating to his/her progress in the case is stored. When signing back onto the computer, the student is started at the main option point. The designation of option five does not orginate a new hospital day.

Upon concluding the information gathering and analysis process, the student specifies option four to state the diagnosis of the patient's problem and to begin a treatment. Once the student initiates the diagnosis, he/she cannot return to the main option point and so at this point is given a final opportunity to return. Upon proceeding, the student designates a diagnosis by typing an appropriate code number as specified in a handout. The program than permits the student to enter up to an including seven lines of additional description or comments. For the intended treatment plan, the student types in from one to ten treatment code numbers obtained from the handout. Next, the program requests the student to answer a short questionnaire which assesses the student's attitude toward the simulated patient experience. Finally, correct diagnosis and treatment for the patient is presented to the student.

The program accumulates a sequential day-by-day record of the student's ordering of tests, the total number of hospital days, the entire charges of the hospital stay and tests, the diagnosis, and the treatment. Upon completion of the case, this data is stored in a file, and the student's course area is readied for another patient.

A Comparison with CASE

As the CASE system is familiar to most medical educators concerned with the teaching of clinical skills, it is worthwhile to make some comparisons between CAPS and CASE. Technical and design differences will be noted but no judgment will be made as to which system is the more effective.

A very noticeable difference is the fact that the CASE simulations utilize a natural language format in which the user constructs an information request in his/her own words. On the other hand, CAPS employs a code system to represent requests. This feature of the CAPS design permits simpler student input, demands less student terminal contact time, and allows an unsophisticated answer-processing capability. Another factor influenced by the uncomplicated CAPS design philosophy is the amount of time involved in implementing the system. The CAPS system was programmed in fewer than three months, considerably less time than was required to implement the more complicated CASE system.

The CAPS simulations necessitate much less direct access disk space than the CASE patients, another significant difference. This smaller storage requirement is due partly to the CAPS feature of sharing files rather than duplicating common information and partly to the complexities of the CASE answer-processing procedures. Furthermore, the two-file design features gives the CAPS system a great deal of flexibility. New cases can readily be added to the system by simply creating an appropriate abnormal file for the patient. Although CASE operates on a similar principle, the method of adding new cases appears to be much more involved. CASE literature (2) states that cases can be easily defined and require only a small amount of the physician's time. However, after several years of operation, the few cases produced seems to indicate otherwise.

Realism is incorporated into the CAPS simulations in the following ways: (1) Test results are available only after the appropriate number of days required to complete the test have passed. (2) Certain tests cannot be run while other incompatible tests are scheduled. (3) Charges are accumulated for both the tests run and the number of patient hospital days. (4) Test results are not static but are generated each time a test is requested. In the CASE system the static results of a test are available immediately, with no charge and without reference to previously ordered tests.

Both CAPS and CASE provide feedback to the student concerning the correct diagnosis and treatment. However in the CASE system, the diagnosis and treatment can have an effect on the patient, or the patient's symptoms may change during the case.

Conclusion

The CAPS system was tested on 175 medical students Fall Semester, 1974, with each student completing at least one patient case. For this trial run, eleven cases were developed dealing with the malabsorption syndrome. In general, the students were not as efficient nor effective in regard to the time and laboratory expenditures nor the completeness of the laboratory-data acquisition as was the faculty consensus. However, student reaction was highly favorable. CAPS has been deemed a success by both the faculty of the Pathology Department of The University of Iowa College of Medicine and the students, with additional cases being made available and greater utilization of the system anticipated in the near future.

It is believed by many medical educators that medical students need to be assisted in developing skills and techniques in: (1) gathering and analyzing pertinent information about the patient; (2) consulting supplementary materials to gain further knowledge; and (3) arriving at a definite diagnosis and treatment plan. CAPS appears to be a feasible approach to facilitating such learning by providing real patient cases within realistic hospital contraints.

List of References

1. Bobula, J. A. and Page, Gordon G., Manual on Construction of Written Simulations, Center for Educational Development, University of Illinois College of Medicine, Chicago, Illinois, 1973.

2. Computer Systems Section, Documentation For Computer-Aided Simulation Of The Clinical Encounter (C.A.S.E.), Center for Educational Development, University of Illinois College of Medicine, Chicago, Illinois, April, 1974.

3. Diamond, Herbert S.; Plotz, Charles M.; and Weiner, Max, "Computer Assisted Instruction In Medicine: A Report On Course Development", ADCIS Conference, Ann Arbor, Michigan, August, 1973.

4. Harless, W.G.; Drennon, G.G.; Marxer, J.J.; Root, J.A.; and Miller, G.E., "CASE: A Computer-Aided Simulation of the Clinical Encounter", Journal of Medical Education, 46:443-448, 1971.

5. Laboratory of Computer Science, Computers in Patient Care and Medical Education, Massachusetts General Hospital, Boston, Massachusetts, 1974.

6. Muller, M.E., "Comparison of Methods for Generating Normal Deviates on Digital Computers", Journal of Association of Computing Machinery, 6:3: 376-383, July, 1959.