The purchase of a consumer item usually presents the buyer with a number of choices in a range of products. So it is with computers and, for those that are applicable to office practice, most physicians have no basis or experience for judging the systems available. Computer hardware is new and mysterious to most of us and the problem of selecting systems is all the more difficult because hardware provides only the beginning—the potential. Software, the series of instructions that tell the computer what to do, is required to actuate the potential; so the choices multiply.

A major purpose of *M.D. Computing* is to provide physicians with a basis for judging medical computer systems. Judging a system on the basis of the availability of specific functions, e.g., fee slips or recall capability, is simply a matter of comparing your wish list against the advertised features of the system. However, one must also ask more general questions about these systems. Are they easy to use, fast, reliable? The answers to these questions depend upon more general characteristics of the system, for example, the manner in which it requests information or how it identifies patients, and can make all the difference between the success or failure of an office practice systems. In this series of reports, we will describe how various systems differ regarding these general characteristics, and recommend those characteristics to choose and those to avoid. In this first of a series, we discuss the difference in the approach of various vendors to data entry.

In this first review, approaches to data entry and the features that vendors provide to assist these processes will be discussed.

**TWO KINDS OF USER/COMPUTER DIALOGUE**

Data are entered into the computer by means of a dialogue. The computer prints a question and the operator types in a reply. Two major modes are available for accomplishing this dialogue: one is screen oriented and the other is line oriented. In the latter, the computer prints one question at a time and expects one answer. After the answer is entered, the computer prints the next question on the next line and again waits for the reply. If more questions are asked than there are lines on the screen, the text obligingly moves upward, like a scroll. The top line disappears from view as the newest question appears on the bottom line. In the early days of the computer, the line-oriented mode was the only one possible because the teletypewriter—the primary device for entering data—had
no screen.

Screen-oriented dialogue became available with the development of economical cathode-ray tubes (CRT), which display the typed characters on a video screen. The computer can ask many questions at once by displaying a “form” on the CRT screen. The form contains blanks that are labeled to indicate what information is to be entered. A blinking marker, called a cursor, appears on the screen to indicate where the computer requests information. (Note that most software systems use some kind of cursor or pointer, regardless of the input mode.) The computer positions the cursor at the first input position on the form and the operator responds by typing in the answer. Then the cursor moves on to the next input position and again the operator replies. And so on. In contrast to the line-oriented style, the location of the form in the screen-oriented mode remains constant; only the cursor moves.

The screen-oriented mode of input, in general, is better. It provides rapid visual orientation, makes it easy to correct answers, and is less distracting since the form on the screen remains constant while “you” move around in it. However, these same advantages can be nullified by a screen system that is slow, poorly organized, or contains a cursor that jumps distractingly on the screen while the data are entered.

A good line-oriented dialogue that also permits the operator to back up and change data that have already been entered is a close second. In the end, work with the system and trust your own impressions. Can you appreciate what the computer wants? Is it easy to recognize where you are in the program? Does the process flow smoothly and clearly?

INPUTS: DIRECT ENTRY VS MENU SELECTION

Whether the dialogue is line oriented or screen oriented, there are two ways to answer the computer’s questions. The first alternative is to type in the data directly; the second is to select the answer from a menu—a list of choices. The most common menu display presents two or three columns of numbered choices. You choose the item by typing in the number. For example, if the practice has four physicians, a menu for identifying the attending physician might appear as follows:

| 1. DR. BROWN | 3. DR. WHITE |
| 2. DR. BLACK | 4. DR. GREEN |

ENTER ATTENDING PHYSICIAN

In this example we would identify Dr. White as the physician by entering the number “3” on the keyboard. Menu selection technique should be used whenever the possible entries are small in number. Almost all systems employ menus for choosing major system functions such as patient registry, charge entry, or payment posting. Unfortunately, few systems use menus as extensively as they should.

Overflow Menus

There are a number of aspects to consider about menu selection. First is the question of fitting the number of available choices on one screen. A typical CRT has 24 rows, each able to accommodate 80 text characters. But since all the space cannot be devoted to the menu—room is needed for dialogue and “fixed” information—the number of menu choices displayed is more realistically between 15 and 20. However, there are ways to deal with the excess. Overflow screens are one such solution. They enable the computer operator to ask for the “next” screen if the current screen does not contain the choice he seeks. Overflow screens, like pearls on a necklace, are chained together and thus require the operator to view every screen in sequence if the choice of interest is on the last screen.

Hierarchical Menus

A hierarchical structure of menu screens is preferable to the above if the choices fill more than a few screens. In a hierarchical menu display, the first screen may list 15 categories of procedures and the other screens would list the actual procedures in each category. In this case, the user would only have to view two screens to get to any of the 15 menus.

Suppressable Menus

Also look for the option of turning off the menu display. Menus are a boon while you are learning to operate a system, but they can become an annoyance once you have become proficient, because of the delay inherent in generating the menu display.

Selection Devices for the Future

A number of pointing mechanisms are available which permit one to choose options in a menu without using the keyboard. A pen-like device with a small light at the end and a switch for turning it on is probably the most widely used such device at the present time. The computer responds to the “light-pen” signal by beeping or highlighting the selected choice. A touch screen is an even more direct mechanism for selection. The operator simply touches a finger to the screen to elicit the desired response. The “mouse,” a flat plastic device about half the size of a cigarette pack, is the newest selector. The mouse is placed on the table to the right side of the terminal and connected to it by a small wire. When it is moved on the table, the cursor moves concurrently on the screen in the same direction and for the same distance—remote control from table to screen. The mouse has 1-3 buttons on its surface that are used to select or activate the option that lies under the cursor. The Apple Lisa comes with a mouse, and a number of vendors have announced “mice” that can be added to existing computer systems. In tests, users could select segments of text much faster with a mouse than with the use of key sequences [1]. Because of this and its low price, pundits predict that it will become the most popular selection mechanism.

With very high resolution terminals, pictures can be used instead of text to represent menu choices. For example, on the Apple Lisa, a “garbage can” identifies the process of discarding a file. By moving the cursor to the garbage can, and then pressing the appropriate button on the mouse, the data being worked on by the operator are “thrown away.” Whether the above pointing mechanisms nor graphic icons