ALGORITHMS AND COMPUTERS

David W. deM Alexander
University of Toronto Schools
Toronto, Ontario

There are several areas of study in the secondary schools which may be concerned with computers. In the technical stream, students can study circuitry, testing, and repair. In the commercial stream students can study the application of computer methods to business problems (data processing). In the academic stream, students can study computer science as a separate subject or can make use of computer methods to solve problems arising in present academic classes.

This paper is concerned with a course which makes use of computer methods to further the student's knowledge of mathematics. Two possible techniques can be used. The emphasis can be placed on teaching a computer language with little training on the basic techniques or problem solving which are adaptable to computer implementation; or the emphasis can be placed on the methods of problem solving, thus dealing only incidentally with the computer language. The first type of course has a place in the education of students who are interested in acquiring a skill which has vocational value. The second type of course is one that provides a skill but also has further academic value in that it gives experience in a different type of problem solving technique.

An experimental text for such a methods-oriented course was used in a Grade XII class at University of Toronto Schools during the 1965-66 school year. The text was the preliminary edition of *Algorithms, Computation, and Mathematics* by the School Mathematics Study Group. It is composed of three separate but related sections: a main text, a FORTRAN supplement, and an ALGOL supplement. This is an attempt to influence commercial texts to provide a similar format so that new languages can be included in a basic text without a complete revision being necessary. The class involved used ALGOL since they were using a Burrough's installation (B5500) made available through the generosity of Burrough's Business Machines Ltd. and K.C.S. (a Toronto firm of consultants in operations research and computer applications). Much of the key punching for the students was done at the Ontario Institute of Studies in Education.

The course is based on the study of algorithms. An algorithm is defined as any unambiguous plan for carrying out some process in a finite number of steps.

This is a familiar concept illustrated by (1) a recipe for a cake; (2) instructions for assembling a "prefab" cottage; (3) the procedure for subtracting a polynomial; (4) the formula for the square of a binomial; (5) the formal method for determining the square root of a number; (6) Newton's method for determining the square root of a number; and (7) the procedure for obtaining the product of two matrices.

A useful algorithm has certain special characteristics. Besides being unambiguous and having a finite number of steps, it should be applicable to a set of problems rather than an individual problem; and if it is to be suitable for implementing on a computer, it should involve a set of repetitive steps.

A clear method for describing an algorithm is a flow chart as illustrated below.

Example 1

A farmer sells a cow for \$75.75. Assuming that he receives this amount in 20 dollar bills, as far as possible, calculate the number of 20 dollar bills received.

Solution

Number of 20 dollar bills

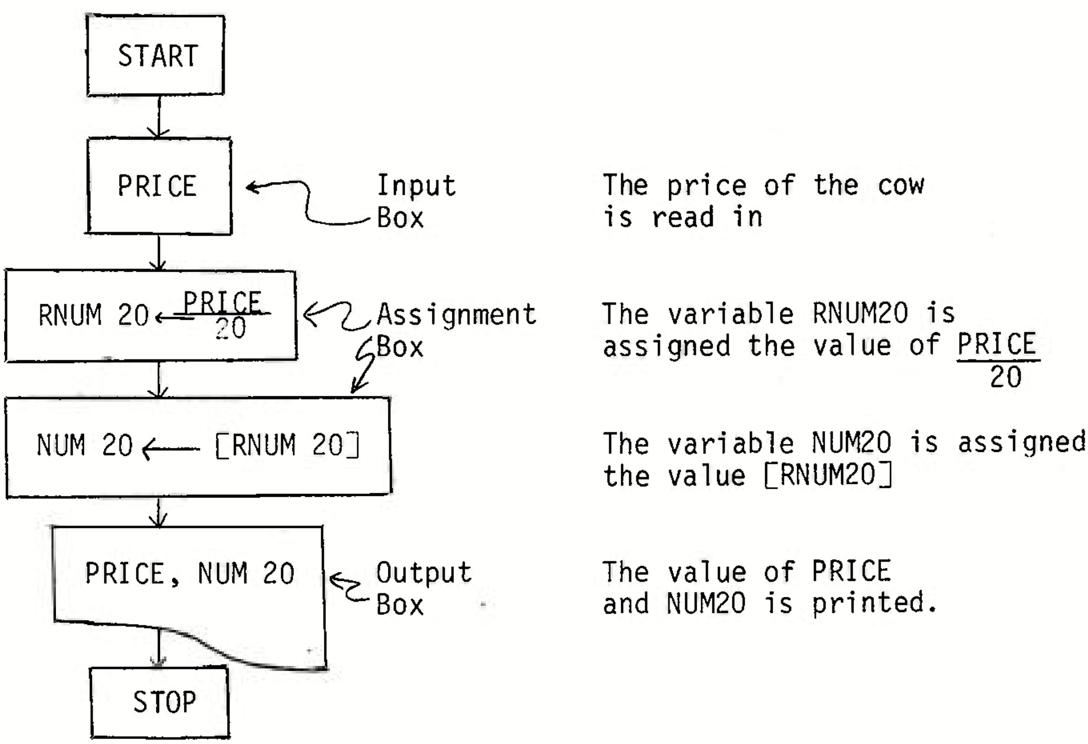
= Integral part of $\frac{75.75}{20}$

$= \left[\frac{75.75}{20}\right]$

Definition

Greatest integer function is the function such that $x \rightarrow [x]$, $x \varepsilon R$, where $[x]$ is the greatest integer which does not exceed x.

Flow Chart



The example illustrates the use of boxes of different shapes to represent different types of instructions in a flow chart and also the symbol $\leftarrow$ as a symbol used when assigning a particular value to a variable. From the flow chart above, with some instruction on the computer language being used, it is possible for the students to make up a program for running on a computer.

A major feature of the examples in the text is the way in which one problem is expanded upon in exercises from chapter to chapter. An illustration of this is indicated by the following example.

Example 2

A carnival wheel has 32 sections, numbered consecutively in a clockwise direction 0,1,2 ---, 31. The sections are colored blue, green, red, and yellow consecutively. The wheel is spun in a counterclockwise direction and points are awarded by the color of the section opposite the pointer when the wheel stops according to the following table.

Section color	Blue	Green	Red	Yellow
Points	-30	-10	10	30

Make up an algorithm to simulate this game.

Hint: Use S as original sector number

M as number of sectors spun through

K as a member of {0, 1, 2, 3}

where 0 $\leftrightarrow$ Blue, 1 $\leftrightarrow$ Green, 2 $\leftrightarrow$ Red, 3 $\leftrightarrow$ Yellow

P as the number of points.

Discussion

This problem requires several days to study and discuss in order to arrive at a satisfactory algorithm. A variety of interesting topics are introduced in the discussion: modular arithmetic, curve fitting; expectation. A study of particular values of S and M indicates that S, the sector number at the completion of the spin, is the remainder after obtaining the integral quotient of

$$\frac{S + M}{32}$$

This may be represented symbolically by $S' = S + M - \left[\frac{S+M}{32}\right] \times 32$

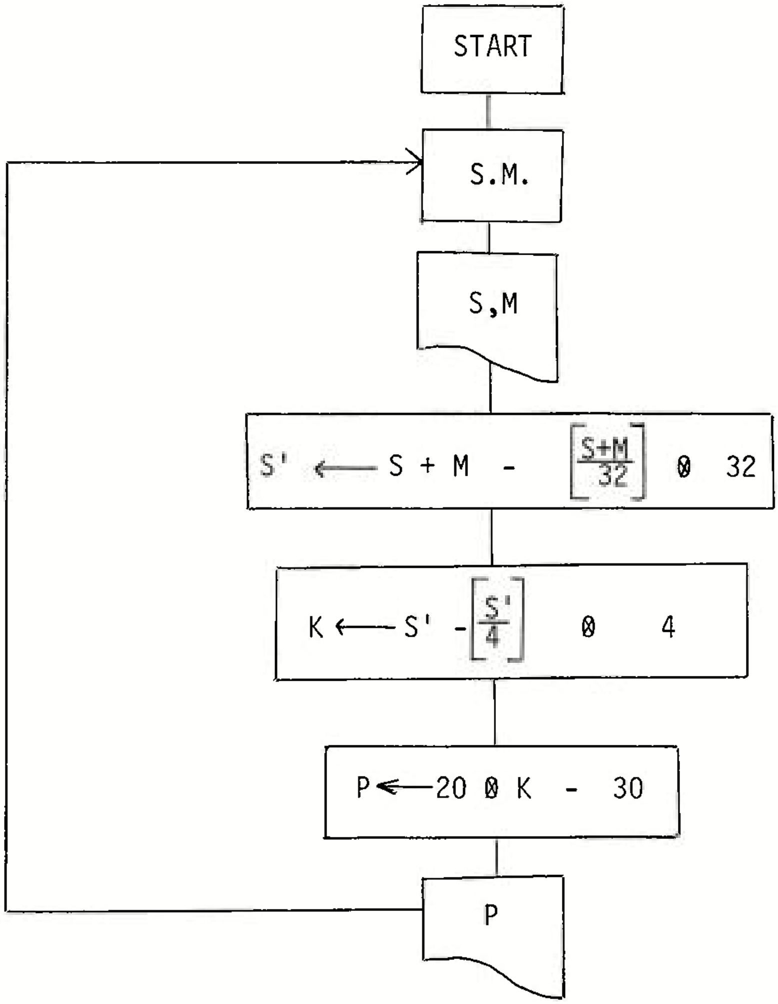
It is clear that the value of K is obtained by a similar process where 4 is the divisor. Thus

$$K = S' - \left[\frac{S'}{4}\right] \times 4.$$

To assign point values corresponding to a specific value of K, it is necessary to express P as a function of K (if a condition box is not to be used). A study of the graph of {(K, P)} suggests that the function is linear. It follows that P = 20 X K - 30.

Flow Chart

This simulation of the game provides a loop so that successive "spins" may be considered. Obvious extensions of this problem are dealt with as the students learn more programming techniques.



1. Point values are introduced which are not determined by a simple function. This requires the introduction of condition boxes.
2. Point values are changed for different sets of spins. This requires the introduction of subscripted variables.
3. A cumulative total of points scored on successive spins is kept. This requires the introduction of a loop or iteration box.
4. A count is kept of the number of spins required to reach a certain value for the total points. This requires an iteration box.

The extensions of the problem open various topics of discussion and provide for experience in the concept of the probability of an event. The students enjoyed this simulation type of problem and went far beyond the required assignments in building up better simulations of the game.[1]

In Ontario, action is already being taken at the departmental level on the question of how the curriculum should be modified to recognize the impact of computers on society: (a) A revision of the business and commerce curriculum has modernized this course, introducing a study of data processing. (b) A committee is producing a computer language for high schools. (c) A committee of the Curriculum Branch has been set up "to study the impact of computer science on the curriculum".

This committee has prepared guidelines for the type of computer suitable for use in secondary schools; sponsored an experimental use of the SMSG material in 20 schools; and entered into a study of experimental programs being carried on in the US and Canada.[2]

From our own experience with this course, we feel that for many students such a course can give an added dimension to their study of mathematics. It would be a mistake to say that all students will enjoy such a course. Even some students who enjoy mathematics find the attention to detail demand in this computer work to be discouraging.

Any teacher who contemplates introducing a course in computer methods should consider the following:

1. What computer facility will be used?

To avoid long turn around times, it is essential that the installation be powerful enough for the problems you wish to do. A small installation will limit the work to such simple problems that the students will feel the additional work involved in programming is not worth the trouble. It is doubtful if any computer which does not use punched card input and line printer output will be versatile enough for any more than an introduction to the techniques of programming. If it is expected that the students should have a chance to actually work out problems of some depth using computer methods, then a fairly powerful installation is necessary. It is not essential that students see every one of their programs processed. We found that having programs run overnight was a very feasible method.

2. What language will be used?

This will be determined by the installation used. Some variation of

---

[1]For further details of the course see School Mathematics Study Group, *Algorithms, Computation, and Mathematics*, (Montreal: McGill University Press, 1961.)

[2]Further details of the Ontario studies can be obtained from G.A. Kaye, Curriculum Branch, Ontario Department of Education, 44 Eglinton Avenue, W, Toronto, Ontario.

ALGOL has the advantage over present forms of FORTRAN that the position of statements on the cards is not important. This allows corrections to be made fairly easily, an important consideration when dealing with students who are not adept at key punching.

3. What special procedures must be followed to run a program on the particular installation used?

Different installations have different procedures. It is essential that the teacher work with the installation chosen so that he can determine for himself the special procedures that must be taught the students in addition to the text material.

4. What method will be used for key punching cards?

There are four possibilities: (a) have students key punch their own cards; (b) have students use mark sensing equipment to produce their own cards; (c) have key punching done commercially; (d) have students who are receiving special training as typists or key punch operators do the key punching. A student who does not know how to type can waste hours trying to key punch his programs. Mark sensing equipment does not appear to be perfected to the stage where it can process the cards for a class of students (or a school) without breaking down just when it is needed most. Comercial key punching adds an extra cost. The best approach would be to work out a cooperative effort between students programming and students learning to type or key punch so that most of the key punching would be done by those students who will actually need this skill. The working out of the problems involved in such a team effort would be an excellent experience for the students.

The curriculum of the secondary schools is already affected by the influence of computers. We as teachers of mathematics should be aware of the effect this may have on future courses and how we can best use the computer to further our two main goals of teaching students and teaching mathematics.