ALGORITHMS and Their Representations

ALGORITHMS are plans (blueprints) for performing some actions (or operations) on some objects (data, etc.). As an example, we will consider the computation of the weekly (gross) pay of an individual. The objects are numbers; the actions are arithmetic operations of addition and multiplication. This algorithm is not difficult, but it will illustrate many concepts.

VERBAL representations of an algorithm can be given as statements in any natural language. A common pay algorithm is shown in figure 1. The pay rate (of ten dollars an hour) was chosen simply to make multiplication easy for us. Usually, more important reasons determine the pay rate!

For example, if the hours worked are 25, then the pay is simply determined by multiplying the rate by the hours (10*25 = 250). But if the hours are 50, then the pay is the sum of two parts, a regular part and an overtime part. The pay for the first 40 hours is the regular rate multiplied by 40 which is (10*40 = 400). For the part over 40, the 10 hours over 40 are multiplied by the higher rate of 15 to get (50 - 40)*15 = 150. The total pay is the sum of these regular and overtime amounts (400 + 150 = 550). This total pay of $550 is finaly output.

PSEUDO-CODE representations of an algorithm are short symbolic descriptions similar to mathematics, logic, and natural language. For example, hours could be symbolized by H, and the pay symbolized by P. Similarly the mathematical operation of multiplication is shown as an asterisk (*), and the symbol for "less than or equal to" is "<=" which is a combination of the symbol for less than "<" and the symbol for equal "=". The pay algorithm, using these symbols, is given in figure 2. This pseudo-code is very similar to many programming languages.

FLOWCHART representations of algorithms consist of various boxes joined by arrows as in figure 3. Square boxes represent actions, and diamond-shaped boxes represent conditions which determine the arrows to be followed out of the boxes. This form often makes it easy for humans to follow the flow of control.

GRAPHS (or plots) are diagrammatic representations that are convenient for humans to understand. The graph of figure 4 shows how the pay rate R depends on the number of hours H worked. The total pay for any number of hours is actually the shaded area under the curve. For example, at H = 50 the total pay corresponds to the rotated L-shaped area, consisting of three smaller rectangles labelled a, b, and c. The pay is:

```
P = 10*40 + 10*10 + 5*10 = 550.
```

The pay could also be determined by adding area "a" to the rectangle formed by b and c:

```
P = 10*40 + 15*(50-40) = 550.
```

Another way is to take the large rectangle (a+b+c+d) and subtract away the smaller rectangle d.

```
P = 15*50 - 5*40 = 750 - 200 = 550.
```

DATA FLOW DIAGRAMS represent algorithms as machines, or black boxes, with data input and output as in figure 5. In this case, data "flows" in at one end (50 hours), and the resulting data (pay of 550 dollars) flows out at the other end. These diagrams hide the details of an algorithm, but they will be useful later to describe interaction among algorithms. Data flow diagrams indicate WHAT is being done, whereas flowcharts indicate HOW it is done.

More representations of algorithms are possible (flow blocks, tables, trees) and they will be considered later.