**What is an algorithm?**

A program is a set of instructions that a computer executes to achieve a certain desired effect - perform a calculation, render a picture, or produce music. A program is written in a specific programming language.

A computer cannot solve a problem by itself. The programmer must write down in minute details each step the computer has to take to solve the problem. An algorithm is the set of steps taken to solve a given problem.

An algorithm is a step-by-step solution to a given problem. An algorithm has the following properties:

* finiteness - the process terminates, the number of steps are finite
* definiteness - each step is precisely stated
* effective computability - each step can be carried out by a computer

We will write our algorithms in pseudocode. Pseudocode is independent of any language. It is a structured way of expressing an algorithm in English. We should always write our pseudocode so that it reflects the language used in the problem. After we have written our pseudocode and made sure that it is logically consistent and solves the problem; we can easily translate the pseudocode into any programming language that we choose.

There is no rigorous standard for pseudocode. Each author has his own personal style. But there is a structured part to pseudocode. There are three basic constructs in an algorithm:

* Linear Sequence: is progression of tasks or statements that follow one after the other.
* Conditional: IF-THEN-ELSE is decision that is made between two course of actions.
* Loop: WHILE and FOR are sequences of statements that are repeated a number of times.

**SEQUENCE:** In a sequence, there is a linear progression of statements. The result of one statement is used in subsequent statements. Here are some key actions that are performed and the common words used in pseudocode:

* Input: READ, OBTAIN, GET
* Output: PRINT, WRITE
* Compute: COMPUTE, CALCULATE, DETERMINE
* Initialize: SET, INITIALIZE
* Add one: INCREMENT
* Subtract one: DECREMENT

**CONDITIONAL:** In a conditional statement you make a test. The result of the test is a Boolean - either True or False. If the result of the test is True you take a certain course of action and if the result of the test is False you take another course of action.

IF condition THEN

sequence 1

ELSE

sequence 2

ENDIF

If the condition is True *sequence 1* is executed, otherwise *sequence 2* is executed. The ELSE sequence is optional.

**LOOPS:** A loop is a sequence that gets executed several times. A complete execution of a sequence is called an iteration of the loop. There are two main loop constructs - WHILE and FOR.

WHILE condition

sequence

ENDWHILE

The sequence is executed as long as the condition is True. The loop terminates when the condition is False.

FOR range of iteration

sequence

ENDFOR

The phrase *range of iteration* specifies the beginning and the end of the iteration. Here are some examples:

FOR each element in list

FOR each month of the year

FOR i = 1 to 10

You can use a WHILE loop and a FOR loop interchangeably. But normally you use a FOR loop when you know the exact number of iterations of the loop. You use a WHILE loop when the number of iterations is not known a priori. You can also nest one loop inside another or a conditional inside a loop or vice versa. It is best to indent your pseudocode to delineate your blocks of code.

**Method or Function:** A method or function is a body of code that has a specific functionality. A method may or may not have input values to work with. A method may or may not return the result. We use the keyword CALL to invoke a method.

CALL Sort on StudentList

CALL Move with row, column