**B1 – Structured English (Pseudocode):**

* **Pseudocode:** This is not a programming language but is used to plan algorithms. It allows a user to focus on the logics and steps of a program rather than language-specific syntax. It is intended to be very readable and easy to understand.
* It is an alternative method of describing an algorithm that uses text instead of a diagram.
* It should avoid commands or syntax that are only found in a certain programming language.

**Structure:**

* **Sequential Structure:** This is process of following a series of steps or an order of events. Each instruction must be executed in the order they are presented.
* **Hierarchy Structure:** This is an organizational structure in which items are ranked according to levels of importance.
* **Indentation:** Used to separate different statements in your program. Makes the code more understandable and easily readable.

**Pseudocode Operations:**

* BEGIN | Used to signal the start of the program.
* END | Used to signal the end of the program.
* INPUT | Used to enter a value into the program
* OUTPUT | Used to display something to the user from the program
* PRINT | Another way to display something to the user.
* READ | This is an input used when reading data from a data file
* WRITE | This is an output to a data file. Writes a value to a text file.

**Pseudocode Decision:** (Conditional statements to alter the direction of the program)

* IF | An programming conditional statement that if proved true, it will branch to another part of the program.
* THEN | In case the IF statement is proved true, THEN it will execute a specific line of code.
* ELSE | In case the IF statement is proved false, it will execute something else.
* ELSE IF (EL IF) |Lies between an IF & ELSE statement. In case the first IF statement is proved False, it will then follow to the ELSE IF statement to check another condition before then jumping to ELSE in case that is also proved to be false.

Example:

BEGIN

INPUT Number

IF Number = 69 THEN

 PRINT Nice

ELSE IF Number = 420 THEN

 PRINT Nice

ELSE

 PRINT Lame

END

* WHEN | Another type of conditional statement. This disrupts the normal flow of the program.

BEGIN

X = 0

REPEAT

 X = X + 1

 WHEN X = 69 THEN

 PRINT Nice

 Until X = 100

END

**Pseudocode Repetition:** (Repetition of instructions)

* FOR… STEP… NEXT… | This is known as the FOR Loop. Used when the number of iterations is known before entering the loop. It is a control flow statement for specifying iteration.
* FOR is the condition/expression. STEP is adding to a value. NEXT goes through the loop again.

Example:

FOR counter = 1 to 100 STEP 1

 OUTPUT counter

NEXT counter

This will set counter to be a number between 1 and 100. One will be added to counter, while counter will be outputted to the user. Then it will loop again until counter is 100.

* REPEAT… UNTIL… | This uses a check value to test the completion of the loop. The loop must executed at least once.

Example:

X = 1

Repeat

 OUTPUT X

 X = X + 1

UNTIL X = 10

This will output X, then add one to X and loop until X is equal to 10.

* WHILE… ENDWHILE | The condition of a while loop is checked before the loops start. This means that the loop may never occur.

Example:

X = 1

WHILE X <= 10 DO

 OUTPUT X

 X = X + 1

ENDWHILE

This will output X, then add one to X and loop until X is equal to 10.

**B2 – Flowcharts Using Standard Symbols:**

|  |  |  |
| --- | --- | --- |
| **Symbol:** | **Name:** | **Function:** |
|  | Start/End | An oval that represents the start or the end of the flowchart. |
|  | Input/Output | An parallelogram that represents an input or an output |
|  | Process | A rectangle that represents a process.  |
|  | Decision | A diamond that indicates a decision in a flowchart. Remember to put either ‘Yes’ or ‘No’ next to the arrows to indicate the True or False direction of the flowchart |
|  | Arrow/Connectors | An arrow to show the direction of the flowchart. If used appropriately, it can be used to represent loops  |