

Q1:

- 1) Machine languages
- 2) Assembler languages
- 3) High level
- 4) Forth generation languages
- 5) Object oriented languages
- 6) Visual Programming Languages

Q2: Answer one only:

1- Cost of use is a major element in the evaluation of any programming language; Describe the types of different cost.

Cost of use. Cost is certainly a major element in the evaluation of any programming language, but different cost measures are feasible:

(a) Cost of program execution. Research on the design of optimizing compilers, efficient register allocation, and the design of efficient run-time support mechanisms was important. Cost of program execution, although always of some importance in language design, is of primary importance for large production programs that will execute repeatedly. For many applications, speed of execution is of highest concern. With desktop machines running at several million instructions per second and sitting idle much of the time.

(b) Cost of program translation. When a language like C is used in teaching, the question of efficient translation (compilation) rather than efficient execution may be paramount. Typically, student programs are compiled many times while being debugged but are executed only a few times. In such a case it is important to have a fast and efficient compiler rather than a compiler that produces optimized executable code.

(c) Cost of program creation, testing, and use. Yet a third aspect of cost in a programming language is exemplified by the language Smalltalk. For a certain class of problems a solution may be designed, coded, tested, modified, and used with a minimum investment of programmer time and energy. Smalltalk is cost effective in that the overall time and effort expended in solving a problem on the computer is minimized. Concern with this sort of overall cost in use of a language has become as important in many cases as the more traditional concern with efficient program execution and compilation.

(d) Cost of program maintenance. Many studies have shown that the largest cost involved in any program that is used over a period of years is not the cost of initial design, coding, and testing of the program, but total life cycle costs including development costs and the cost of maintenance of the program while it is in production use. Maintenance includes the repair of errors discovered after the program is put into use, changes in the program required as the underlying hardware or operating system is updated, and extensions.

الحلول النموذجية لمادة اساسيات تقنيات البرمجة / المرحلة الاولى فرع البرمجيات الدور الثاني

Q1:

- 1) Machine languages
- 2) Assembler languages
- 3) High level
- 4) Forth generation languages
- 5) Object oriented languages
- 6) Visual Programming Languages

Q2: Answer one only:

1- Cost of use is a major element in the evaluation of any programming language; Describe the types of different cost.

Cost of use. Cost is certainly a major element in the evaluation of any programming language, but different cost measures are feasible:

(a) Cost of program execution. Research on the design of optimizing compilers, efficient register allocation, and the design of efficient run-time support mechanisms was important Cost of program execution. Although always of some importance in language design, is of primary importance for large production programs that will execute repeatedly. For many applications, speed of execution not of highest concern. With desktop machines running at several million instructions per second and sitting idle much of the time.

(b) Cost of program translation. When a language like C is used in teaching, the question of efficient translation (compilation) rather than efficient execution may be paramount. Typically, student programs are compiled many times while being debugged but are executed only a few times. In such a case it is important to have a fast and efficient compiler rather than a compiler that produces optimized executable code.

(c) Cost of program creation, testing, and use. Yet a third aspect of cost in a programming language is exemplified by the language Smalltalk. For a certain class of problems a solution may be designed, coded, tested, modified, and used with a minimum investment of programmer time and energy. Smalltalk is cost effective in that the overall time and effort expended in solving a problem on the computer is minimized. Concern with this sort of overall cost iii use of a language has become as important in many cases as the more traditional concern with efficient program execution and compilation.

(d) Cost of program maintenance. Many studies have shown that the largest cost involved in any program that is used over a period of years is not the cost of initial design, coding, and testing of the program, but total life cycle costs including development costs and the cost of maintenance of the program while it is in production use. Maintenance includes the repair of errors discovered after the program is put into use, changes in the program required as the underlying hardware or operating system is updated, and extensions.

Q1 2- If you have the infix form expression $(B^2 - 4 * A * C)^{(1/2)}$

Describe the postfix form expression for it.

$(B^2 - 4 * A * C)^{(1/2)}$

infix to postfix

$(B^2 - 4 * A * C)^{(1/2)}$

$((B^2) - ((4A*) * C)^{(1/2)})$

$((B^2) - ((4A*C*))^{(1/2)})$

$(B^2 4A*C*-)^{(1/2)}$

$(B^2 4A*C*- 1/2/)$

postfix notation

Q3: Define the following

1- **Data** A computer must provide various kinds of elementary data items and data structures to be manipulated.

2- **Orthogonality**. The term orthogonality refers to the attribute of being able to combine various features of a language in all possible combinations, with every combination being meaningful.

3- **Modular programming**

Modular programming allows the grouping of procedures into what is known as modules.

With modular programming, procedures of common functionality are grouped together into separate modules. A program therefore no longer consists of only one single part rather is now divided into several smaller parts which interact with each other through procedure calls.

4- **Coding** is the process of transforming an algorithm into a program in a specific computer language.

Q4: What does this expression evaluate to?

$$-4 * (10 / (1 + 2) - (9 - 2) \% 3 + 4) / 3$$

Q5: Define the translator and describe the types of Programs languages translator.

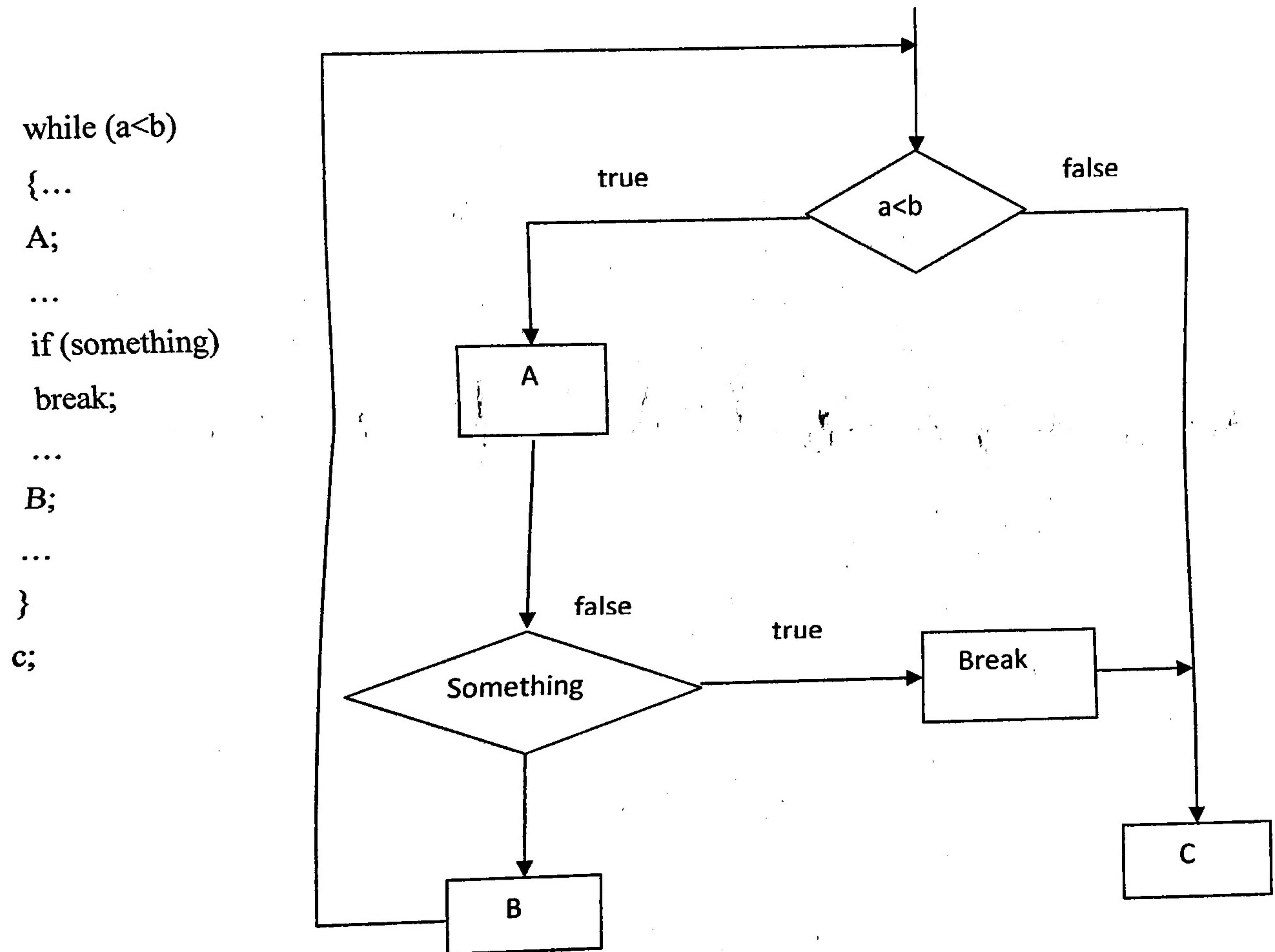
Programming languages translator (PLT)

PLT are programs that translate other programs into machine language instruction codes that computers can execute and can be divided into the following:-

- 1) An assembler translates the symbolic instructions codes of program written in an assembler language into machine language instructions.
- 2) A compiler program that translate a high level programming language into machine language program.
- 3) An interpreter program that translates and executes each source language statement before translating and executing the next one.

Q6: Define the break statement; explain the structured of it with example.

- **Break statement.** Some languages, such as C, include a **break** statement as a form of structured explicit control. Usually the **break** causes control to move forward in the program to an explicit point at the end of a given control structure. Thus **break** in C causes control to exit the immediately enclosing while, for, or switch statement. This still gives a one-in, one-out control structure that permits formal properties of a program to be developed.



Q7: Define iteration statements, then compare between repetition while condition and repetition while incrementing a counter, give example for each type.

Iteration Statements

The basic structure of an iteration statement consists of a head and a body. The head controls the number of times that the body will be executed, while the body is usually a (compound statement that provides the action of the statement. Although the bodies of iteration statements are fairly unrestricted, only a few variants of head structure are usually used. Let us look at some typical ones.

- **Repetition while condition holds.** A somewhat more complex iteration may constructed using a repeat while head. A typical form is:

while test do body

In this form of iteration statement the test expression is reevaluated each times after the body has been executed.

- **Repetition while incrementing a counter.** The third alternative form of iteration statement is the statement whose head specifies a variable that serves as a counter or index during the iteration. An initial value, final value, and increment are specified in the head, and the body is executed

- repeatedly using first the initial value as the value of the index variable, then the initial value plus the increment, then initial value plus twice the increment, and so on, until the final value is reached.

Either **For** $i=1$ to final value **do** *body(increment)*

Or **For** $i=$ final value down 1 **do** *body(decrement)*

Q8: Write an algorithm to find of 10 given numbers and convert it into flowchart.

Algorithm & Flowchart

Develop an Algorithm to find the sum of 10 given numbers and convert it into Flowchart.

Step 1. Initialize $SUM = 0$
 Step 2. For 10 numbers
 Step 3. Read number
 Step 4. Add the number to SUM
 Step 5. Any more number?
 Step 6. If YES Goto Step 2
 No Write SUM
 Stop

