

Concept of Programming and Programming Languages

Que: 1 Multiple Choice Questions

- Set of instructions is called \_\_\_\_\_.  
 a. Group                                      b. Software                                      c. Program                                      d. None of these
- Which language is directly understood by computer without any translation?  
 a. Procedure Oriented Language                                      b. Machine Language  
 c. Assembly Language                                      d. High Level Language
- Mnemonic codes & symbolic addresses are used in which programming language?  
 a. Object Oriented Language                                      b. Non-Procedural Language  
c. Assembly Language                                      d. Machine Language
- Which translator does not save object code after translation of source program written in high level language?  
 a. Translator                                      b. Compiler                                      c. Assembler                                      d. Interpreter
- Process of finding and correcting errors in a program is called \_\_\_\_\_.  
 a. Compilation                                      b. Coding                                      c. Debugging                                      d. Documentation

Que: 2 Fill in the Blanks:

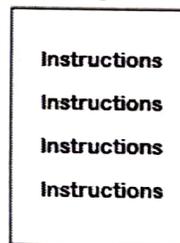
- A person who writes the program is called \_\_\_\_\_
- Low level internal details of hardware are required for programming in \_\_\_\_\_
- \_\_\_\_\_ is the pictorial representation of algorithm
- Process of translating source program written in high level language into object code is called \_\_\_\_
- Those errors which are not detected by the compilers are called \_\_\_\_\_ errors.

- Ans: 1. Programmer                                      2. Low Level                                      3. Flow Chart  
 4. Compilation                                      5. Logical Errors

Que: 3 Write the Full form of following:

- |            |                                     |
|------------|-------------------------------------|
| 1. Opcode  | Operation Code                      |
| 2. Operand | Operation Address                   |
| 3. 4GL     | 4 <sup>th</sup> Generation Language |
| 4. SQL     | Structured Query Language           |
| 5. OOP     | Object Oriented Programming         |

Program



Que: 4 Short Answer Type Questions.

Q1: What is Programming?

Ans: Programs are a set of instructions that a computer can understand to perform a task. The process of writing these instructions into a program is called programming. The person who writes the program is called the programmer.

Q2: What are Procedure Oriented Programming Languages?

Ans: Procedure Oriented Languages are considered as third generation programming languages (3GLs). To create programs in these languages, it is divided into small procedures or subroutines. Each procedure contains a series of instructions for carrying out a specific task. After creating the procedure, we can use them one or more times anywhere in the program. The sequence of program instructions in these languages is very important. FORTRAN, COBOL, Pascal, C, etc. are examples of some popular procedural languages.

Q3: Write the names of different symbols used in flowcharts.

Ans: Following are the different symbols used in the flow chart:

- Terminal (Oval symbol)
- Input / Output (parallelogram)
- Processing (rectangle)
- Diamond (rhombus)
- Flow lines (arrows)
- Connectors (circles)

**Q4: Write the steps used in Programming Process.**

**Ans: The steps used in the programming process are as follows:**

1. Define the problem to be solved
2. Develop a plan for solving problem
3. Coding the solution in high level language
4. Compile the program
5. Test and debug the program
6. Documenting the program

**Q5: What are Syntax Errors?**

**Ans: These are the errors that occur when we develop programs that do not follow the rules or syntax of programming language. These types of errors are automatically detected by compilers during the compilation process. A program cannot be successfully compiled until all syntax errors in the program have been corrected. Some examples of syntax errors in C language are: Missing Semicolon, Variable not declared, etc.**

**Que: 5 Long Answer Type Questions.**

**Q1: What are low level programming languages? Explain their advantages and disadvantages.**

**Ans: Machine and assembly languages are called low-level languages. These are explained below:**

- **Machine language: Machine language is also called binary language. It is the fundamental language of computer systems because it is understood directly by the computer system. The computer does not require any translation to understand this language. This language is made up of only two binary digits 0 and 1.**
- **Assembly Language: This language is also called Symbolic Language because it uses the symbolic names of the instructions instead of the binary code. The symbolic names of the assembly language instructions can be easily remembered.**

**Advantages of Low-Level Languages:**

- These languages can communicate / interact directly with computer hardware
- These languages work faster than computers.

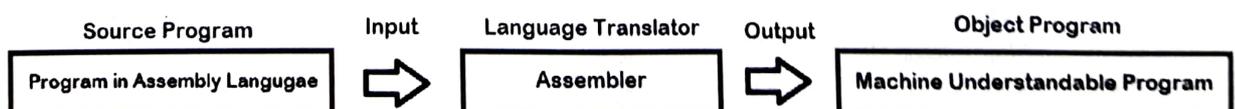
**Disadvantages of Machine Language:**

- For programming in low level languages, the programmer needs to know the internal structure of the hardware.
- Programs created in low level languages are machine dependent.

**Q2: What are Language Translators? Explain any one translator in detail.**

**Ans: Language translators are also called language processors. These are system-programs. The purpose of developing language translators is to accomplish two main tasks: first, to translate source programs into Object Code, and second, to detect syntax errors in the source program. Each language has its own translator program that can only translate programs written in that particular language. Examples of language translators are: assemblers, compilers and interpreter.**

**Assembler: This is a language translator that converts programs written in assembly language into machine language. A program written in assembly language is called a source program. This source program cannot be directly understood by the computer. Therefore, it is necessary to translate it into a machine-understandable format. It is the assembler that converts the source program of the assembly language into a machine-understandable program. The code generated after translation is called the object program, which is used to execute the program.**



**Fig: Working of Assembler**

**Q3: What is algorithm? Explain the different features that an algorithm should have.**

**Ans: Creating algorithms is a basic requirement in computer programming. This is a step-by-step description of how to solve a given problem. An algorithm has limited steps and it should always produce some (right or wrong) result. Before creating a program, a programmer first sets the algorithm. An algorithm should have the following features:**

- Every step must be accurate.
- Every step should be clear, which means it should not be ambiguous.
- Input and output should be carefully determined.
- Steps should not be repeated indefinitely.
- After implementing the steps, the required output should be obtained in any case.

**Q4: Explain different types of errors found in the computer programs.**

**Ans: There are two common types of errors in programs:**

- **Syntax Errors:** These are the errors that occur when we develop programs that do not follow the rules or syntax of programming language. These types of errors are automatically detected by compilers during the compilation process. A program cannot be successfully compiled until all syntax errors in the program have been corrected. Some examples of syntax errors in C language are: Missing Semicolon, Variable not declared, etc.
- **Logical Errors:** These errors occur when there are the errors in the logic of the program. If there are logical errors in our program, it will be successfully compiled but it will produce incorrect results / output. Such errors cannot be detected by the compiler. These can be easily found with the help of debugging tools.