**Chapter-1**

**The programming process**

1. **Computer program**

A computer program consists of a group of instructions for a computer that cause it to perform a desired task. A computer program is a collection of instructions that performs a specific task when executed by a computer.

1. **Programming process**

The programming process is, therefore, a problem solving process, and it consists of the following activities:

1. Defining the problem
2. Preparing an algorithm
3. Preparing a program flowchart
4. Coding the program
5. Debugging and testing
6. Documenting
7. **Programmer**

The person who writes the program is called Programmer. If the person writes Java program then he is Java Programmer or if the person writes Python program then he is Python Programmer and so on.

1. **System Analyst**

Someone who plans the collection of equipment, programs, people, and procedures that make up a system is called System Analyst.

1. **Input & Input Device**

Data that is to be read and processed by a program; the operation of reading such data is called input. The devices which are used as input peripheral is called input device. Keyboard, Mouse, Scanner etc. are the example of input devices.

1. **Output & Output Device**

The results of processing by the computer by the computer; the process of producing such results is called output. The devices which are used as output peripheral is called output device. Monitor, Printer, Speaker etc. are the example of output devices.

1. **Documenting & Documentation**

Documenting is a name of programming process activities which preparing a written record of all activities associated with the programming process.

Documentation is a name of document where programming process or using process is written down. The documentation may provide on paper, online or any digital media. User guide or quick reference guide is the example of documentation.

1. **Documenting Type**

There are two types of documenting. They are,

1. Technical Documenting: The process of preparing documentation for developer.
2. End User Documenting: The process of preparing documentation for end user. End user documentation is also called user manual.
3. **Problem**

A statement of the desired output (result to be produced by the program), which usually be provided to the programmer.

1. **Loop & Looping**

A group of instructions that is executed repeatedly until a specific condition is encountered. Looping is a process to perform a set of operations repeatedly.

1. **Algorithm & Program Flowchart**

A sequence of steps that describe a method for solving a problem is called Algorithm. A symbolic representation of algorithm is called Program Flowchart.

1. **Compiler, Bug, Debugging, Execution-time-error, Syntax error & Logic error**

A compiler is a special program that processes statements written in a particular programming language and turns them into machine language or "code" that a computer's processor uses.

An error in a computer program is called bug.

Removing the errors from a program is called debugging.

An error, detected during the execution of a program is called execution-time-error. Suppose if we try to something divide by zero then the execution cannot be continued.

A violation of the rules of a programming language is called syntax error.

An error that occurs as a result of faulty reasoning; cannot be detected by a translation program, but will produce incorrect results is called logic error.

1. **Counter**

A counter is a device for keeping track of the number of times something occurs.

1. **Desk Checking**

Desk checking is a reviewing process in which a representative sample of data is manually processed through an algorithm, flowchart, pseudo code, or coded program to locate errors.

1. **Assembly Language**

An assembly language is a low-level programming language for microprocessors and other programmable devices. It is not just a single language, but rather a group of languages. An assembly language implements a symbolic representation of the machine code needed to program a given CPU architecture.

1. **High-Level Programming Language and Examples**

A programming language in which one source program instruction may be translated into one or more object program instructions and which are relatively easy to learn is called high-level programming language. Some examples are given below.

1. BASIC (Beginner’s All-purpose Symbolic Instruction Code)
   1. Easy to learn
   2. Easy to use
   3. To use in solving algebraic problems
   4. To handle business data processing
   5. Widely used on personal computers
2. COBOL (Common Business Oriented Language)
   1. Design for business data-processing
   2. Originally used on large computers
   3. Now available on some personal computers
3. FORTRAN (Formula Translator)
   1. Design for mathematical problem solving
   2. Available on most computers
4. Pascal
   1. General purpose programming language
   2. Available on most computers
5. RPG II (Report Program Generator)
   1. Powerful language for business application
   2. Widely used on smaller business computers
   3. Now available on large computers and some personal computers
6. **Binary**

Refers to a numbering system that represents all values with a combination of ones (1s) and zeros (0s) is called binary.

1. **Branch**

Transfer control to another part of a program is called branch.

1. **Character & Coding**

A letter of the alphabet, a digit, or a special character ($, %, +, etc.) is called character.

Instruction that transfers control to a subroutine is called coding.

1. **Detail line, EOF & Execute**

A printed line that contains information about a single entity is called detail line.

End-of-file condition when reading a file is called EOF.

Cause of a program or group of instructions to perform its intended task is called execute or run.

1. **Field, Record, File & Listing**

A collection of characters used to represent a unit of information about an entity; a subdivision of a record is called filed or item.

A component of a file containing information about an entity; a collection of field is called record.

A collection of related records is called file.

Output on a printer is called listing.

1. **Increment**

Increase the value of a counter is increment. Generallyadding one into certain variable or objects for increasing its weight when a specific condition becomes true is called increment.

1. **Memory**

Computer memory is any physical device capable of storing information temporarily or permanently. For example, Random Access Memory (RAM) is a volatile memory that stores information on an integrated circuit used by the operating system, software, and hardware.

1. **Source program**

Instructions for the computer written in a form that is relatively easy for the programmer to work with; must be converted to machine language by a translation program before it can be run on a computer.

1. **Testing & Total Line**

Running a program with sample data to identify logic errors is called testing.

A line that summarizes data obtained from one or more input records is called total line.

1. **Computer’s limited capabilities**

There are four capabilities of computer.

1. Can perform arithmetic operations.
2. Can compare between two things
3. Can move data about in its memory
4. Can input data and can output results

***TRAINER: MD. REJAUL ISLAM***

***IDB-BISEW ROUND-33***