

Afe Babalola University, Ado-Ekiti, Nigeria

Course Title: Computer Programming 1 (CSC 201)

Instructor: Oguntimilehin A, ogunsabiodun@abuad.edu.ng

CHAPTER ONE

CONCEPTS OF COMPUTER PROGRAMMING

Computer program: A computer program is a set of instructions written in a programming language in a sequential order, so as to control the activities of a computer system.

Computer programming: This is the process of writing a set of instructions in sequential manner using programming language to control the activity of a computer system.

Programming language: This is an artificial language which is used in writing a set of instructions to control the activities of a computer system e.g FORTRAN, COBOL, BASIC. ADA e.t.c.

COMPUTER PROGRAMMING LANGUAGES

There are two main types of computer programming languages; they are:

1. (i) Low level language (ii) High level language.

(a) Low level language. This type of language is closer to the machine compared with the human natural language. The two major examples are the Machine language and the Assembly language.

- (i) **Machine Language:** This is the only language computer understands. The computer directly executes a program written in machine language. Such programs are coded using strings of 0's and 1's. It doesn't need a translator.

Disadvantages of Machine Language

Some of the disadvantages of Machine Language include:

- i. Very bulky.
- ii. They require much time for writing and reading.
- iii. They are prone to error which is difficult to be detected and corrected.
- iv. Very difficult learn.
- v. Can only run on the computer it is designed for (machine dependent)

(ii) **Assembly Language:**

Assembly Language uses MNEMONICS (symbols) to represent data and instructions. Such program eliminates problems associated with machine language. Computer cannot execute directly a program written in assembly language, it requires a translator called assembler. Assembler is a special program designed to translate a program content in assembly language to a machine language equivalent.

Disadvantages of Assembly Language

These include:

- i. It is machine dependent; programmer has to be knowledgeable in both subject area and the operations of the machine.
- ii. It is cumbersome though less cumbersome than that of machine language.
- iii. Very expensive to develop and
- iv. It consumes time

(b) High Level Language

A high level language is a problem orientated programming language, whereas a low level language is machine oriented. The source programs are written in human readable languages like English instead of mere symbols. In other words, a high level language is a convenient and simple means of describing the information structures and sequences of actions required to perform a particular task.

Advantages of High Level Language.

These include:

- i. The person writing the program does not need to know anything about the computer in which the program will be run (Machine Independent).
- ii. The programs are portable.
- iii. Very easy to learn and write

Features of High Level Language

- i. Machine independent (ii) Problem oriented
- iii Ability to reflect clearly the structure of program written in it.
- iv. Readability (v) Programs are portable.

Examples of High level Languages are FORTRAN, COBOL, QBASIC, VISUAL BASIC, JAVA , ADA, and PASCAL e.t.c.

Characteristics of a Good Program

- i. Transferability- Must be able to work on any computer machine.
- ii. Reliability- It can be relied upon to do what it is expected to do.
- iii. Efficiency / cost saving- It must not cost more than its benefits and enables problem to be solved appropriately, quickly and efficiently.
- iv. Simplicity- It should be as simple as possible to understand.

- v. Understandability / Readability- It must be readable and understandable by other programmers and end users.
- vi. Flexibility / Adaptability / Maintainability- A good program must be flexible adaptable and maintainable in order to suit user's need. Modification must be possible and very easy.

PHASES OF PROGRAM DEVELOPMENT (PROGRAMMING)

The process of producing a computer program (software) may be divided into Eight phases or stages:

- (1) Problem definition / Analysis
- (2) Selection or development of an algorithm
- (3) Designing the program
- (4) Coding the programming statements
- (5) Compiling / Compilation stage
- (6) Testing / Running and Debugging the program
- (7) Documentation.
- (8) Maintenance

1. Problem Definition / Analysis Stage: There is need to understand the problem that requires a solution. The need to determine the data to be processed, from of the data, volume of the data, what to be done to the data to produce the expected / required output.
2. Selection or development of an algorithm: An algorithm is the set of steps required to solve a problem written down in English language.
3. Designing the program: In order to minimize the amount of time to be spent in developing the software, the programmer makes use of flowchart. Flowchart is the pictorial representation of the algorithm developed in step 2 above. Pseudocode IPO chart (input processing output) and Hipo chart

(Hierarchical- input-processing and output) may be used in place of flowchart or to supplement flowchart.

4. Coding the statement: This involves writing the program statements. The programmer uses the program flow chart as a guide for coding the steps the computer will follow.
5. Compiling: There is need to translate the program from the source code to the machine or object code if it is not written in machine language. A computer program is fed into the computer first, then as the source program is entered, a translated equivalent (object program) is created and stored in the memory.
6. Running, Testing And Debugging : When the computer is activated to run a program, it may find it difficult to run it because many syntax errors might have been committed. Manuals are used to debug the errors. A program that is error free is tested using some test data. If the program works as intended, real required data are then loaded.
7. Documentation :This is the last stage in software development. This involves keeping written records that describe the program, explain its purposes, define the amount, types and sources of input data required to run it. List the Departments and people who use its output and trace the logic the program follows.

NEXT: PROGRAM TRANSLATORS