Module 1

Overview of System Software

Syllabus

- Introduction to System Software with examples
- Software Hierarchy
- Differentiate between system software and application software
- Introduction to Language Processors: Compiler, Assembler, Interpreter.

Basic Terms

SOFTWARE

HARDWARE

FIRMWARE

Basic Definitions

What is System?

System is the collection of various components

Ex:- College is a system.

College is a system because it consists of various components like various departments, classrooms, faculties and students.

What is Programming? – Art of designing and implementing the programs.

Basic Definitions

In college system, what is program?

- A LECTURE can be a program. Because it has input and output.

Input-> The information that teacher is delivering.

Output-> The knowledge student has been received.

System programming is an art of designing and implementing system Programs.

What is Software ?

- **Software** is collection of many programs

- Software is generally divided into:
  - System software
  - Application software

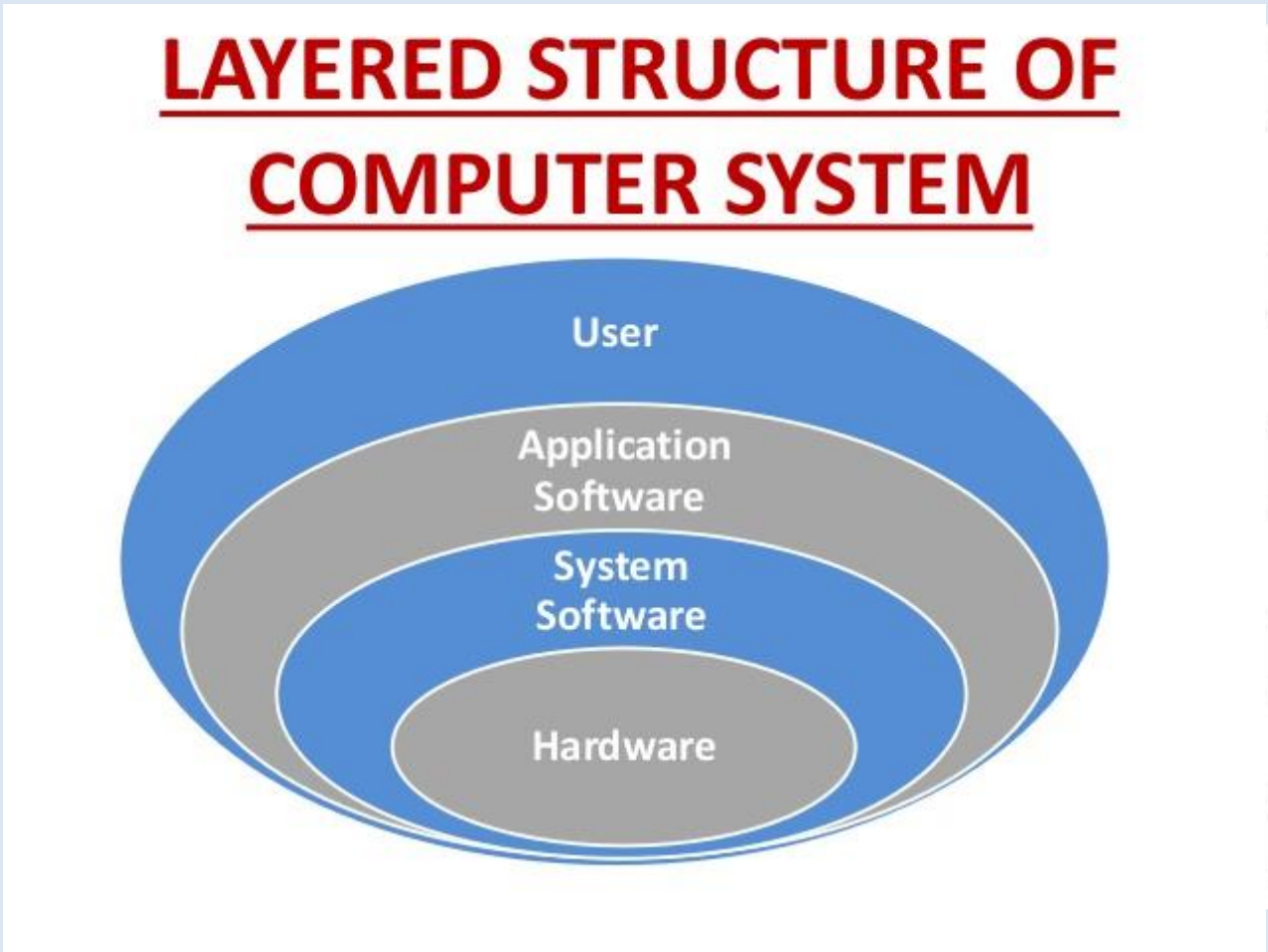
a) System software are programs which help in running the computer system

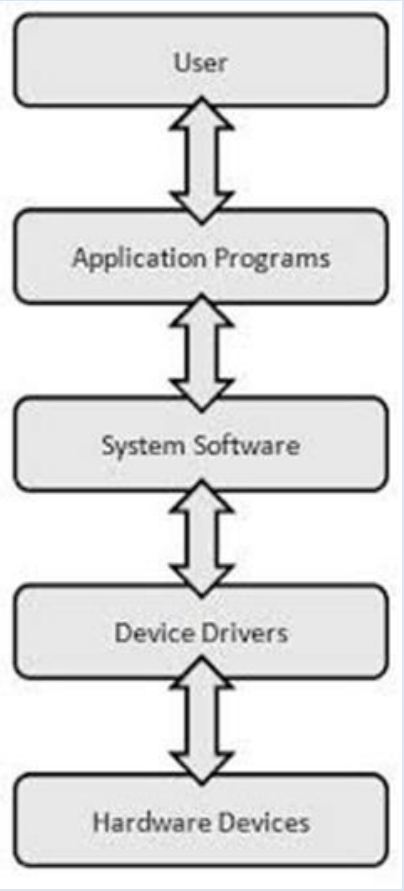
- E.g. Operating System, Compiler etc.

b) Application software are programs which perform specific tasks for the user.

- E.g. Word processing software, Graphics package, Theatre booking software.

Position of System Software



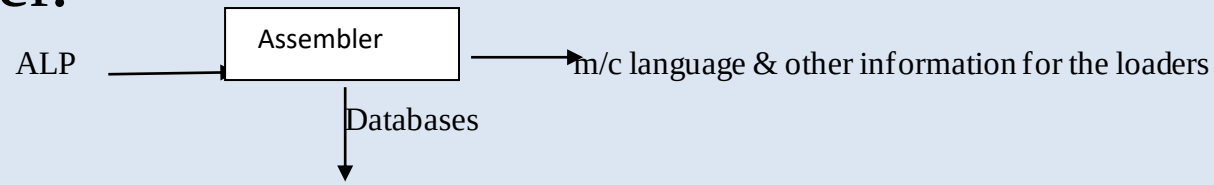


Differentiate Between System Software and Application Software

Basis	System Software	Application Software
Definition	System software consists of variety of programs that support the operation of computer.	Application software focuses on an application or problem to be solved.
Installation	System software are installed on the computer when operating system is installed.	Application software are installed according to user's requirements.
Interaction with	In general, the user does not interact with system software because it works in the background.	In general, the user interacts with application sofware.
Written In	System Software is written in a low-level language, i.e. assembly language.	Application Software is written in a high-level language like Java, C++, .net, VB, etc.
Manages	System software manages resources and helps to run hardware and application software.	Application software perform specific task according to their type.
Dependency	System software can run independently. It provides platform for running application software.	Application software can't run independently. They can't run without the presence of system software
Execution	System software will start running once you turn your computer on. It will continue running until the time that you will turn off your computer.	Application software will run only when required to do so. You will start a word processor when you need to open or prepare some documents.
Examples	Examples of system software are: Operating System, Compiler, Assembler, Macro Processor, Loader or Linker, Debugger, Text Editor, DBMS.	Some examples of application software are word processor, web browser, media player, etc.

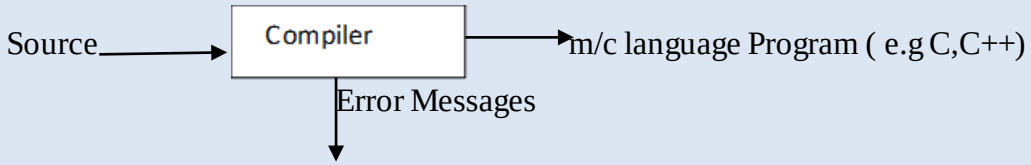
List of System Software's

1. Assembler: An assembler is a program that accepts as input an assembly language program & produces its machine language equivalent along with information for the loader.



2. Preprocessor : The C preprocessor is a macro processor that is used automatically by the C compiler to transform your program before actual compilation. It is called a macro processor because it allows you to define macros, which are brief abbreviations for longer constructs.

3. Compiler: A compiler is a program that reads an input in HLL & translates it into an equivalent program in machine language.



Phases of Compiler: Compiler operates in different phases.\

1. Lexical Analyzer
2. Syntax Analyzer
3. Semantic Analyzer
4. Intermediate Code Generator
5. Code Optimizer
6. Code Generator

Other- Symbol table & Error Handler

4. Interpreter: An interpreter is a translator that reads the program written in HLL & translate line by line into an intermediate form, which it then executes.

5. Loader: Loader is a system program which is responsible for preparing the object program for execution & initiates the execution.

OR

- A program routine that copies a program into memory for execution
- Function of Loader:

a) Allocation

b) Linking

c) Relocation

d) Loading

6. Linker: Linker is a program that combines object modules to form an executable program. Also called Link editor & binder.

7. Operating System: Operating system is system software, consisting of program and data that runs on computer and manage the computer hardware. It is an integrated set of programs that controls the resources of a computer system and provides its users with an interface that is easier to use.

- Objective of OS:

  -Make a computer system easier to use

  -Manage the resources of a computer system

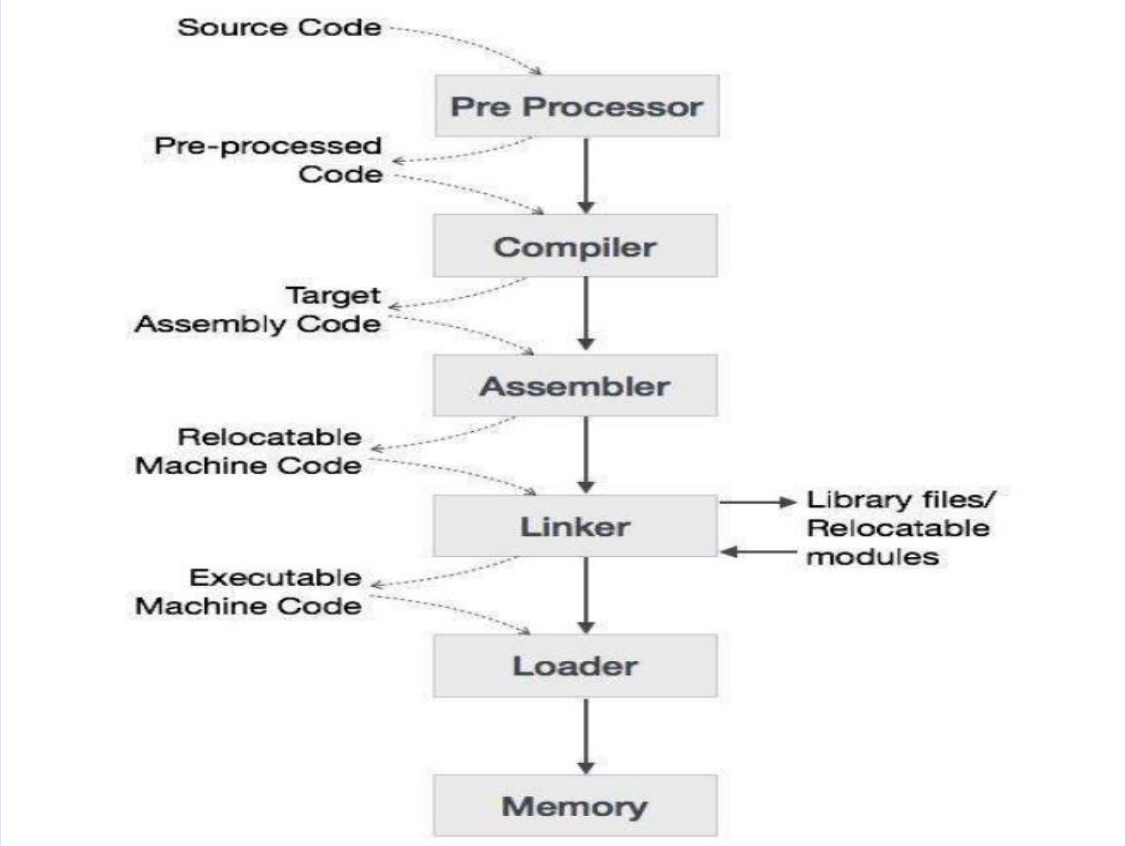
8. Device Drivers:

- A device driver is a program that controls a particular type of device that is attached to your computer.
- There are device drivers for printers, displays, CD-ROM readers, diskette drives, and so on.

Language Processing System

- Computer system is made of hardware and software.
- The hardware understands a language, which humans cannot understand. Humans write programs in high-level language, which is easier to understand and remember.
- These programs are then fed into a series of tools and OS components to get the desired code that can be used by the machine.
- This is known as Language Processing System.
- The tools/programs which performs this task are known as Language Processor

Language Processing System Contd…



Language Processing System Contd…

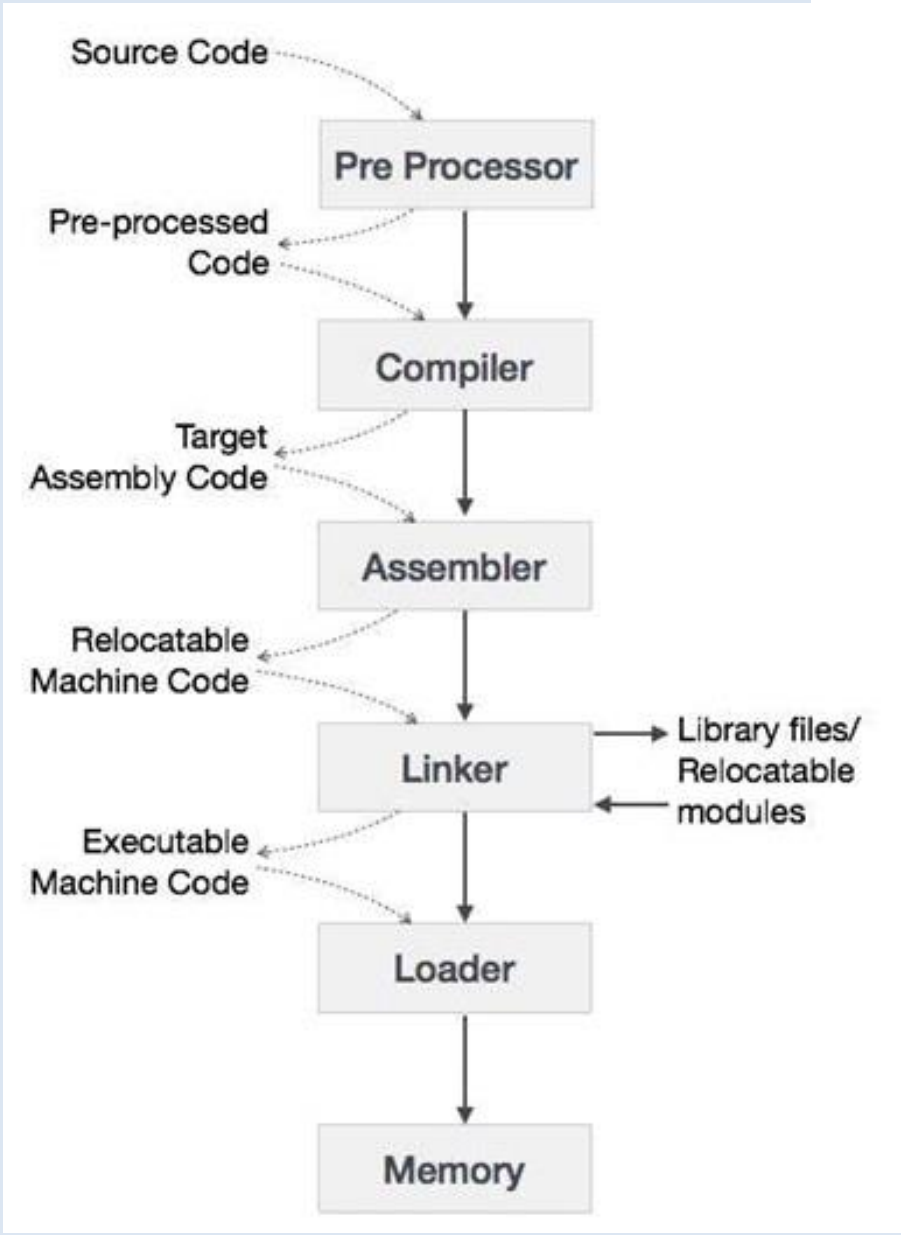
- Let us first understand how a program, using C compiler, is executed on a host machine.
- User writes a program in C language (high-level language).
- The C compiler, compiles the program and translates it to assembly program (low-level language).
- An assembler then translates the assembly program into machine code (object).
- A linker tool is used to link all the parts of the program together for execution (executable machine code).
- A loader loads all of them into memory and then the program is executed.

Questions

- Differentiate between application program and system program.

Indicate the order in which following system program are used, from developing program upto its execution. Assembler, loaders, Linkers, macroprocessor, compiler, editor.

Order in which system program are used, from developing program up to its execution



Language Translator

- Compiler
- Assembler
- Interpreter

Difference between Compiler, Interpreter & Assembler:

Sr. No	Assembler	Compiler	Interpreter
1.	It converts ALP into m/c language.	It converts HLL program into m/c language.	It converts HLL program line by line in intermediate form, which it then executes.
2.	It assembles whole source program.	It compiles whole source program	It translates line by line.
3.	Speed of execution is fast.	Speed of execution is fast.	Speed of execution is slow.
4.	Program need to assemble only once & can be executed repeatedly.	Program need to compile only once & can be executed repeatedly	For every run of program, program need to be translated.
5.	It creates an object file.	It creates an object file.	It does not create an object file.
6.	It requires large memory space to store (But less than compiler).	It requires large memory space to store.	It requires less memory space to store.
7.	e.g. Microsoft assembler (MASM)	MS-DOC C compiler.	Basic Interpreter.
8.	Source language- Assembly	Source language- C, C++	Source language- BASIC, LISP

FAQ?

Q.1 Differentiate between application program and system program. Indicate the order in which following system program are used, from developing program up to its execution. Assembler, loaders, Linkers, macro-processor, compiler, editor. (5 Marks)

Q.2 Define software and Differentiate between system software & application software.(5 Mrks)

Q.3 Differentiate between various language translators. (5 Mrks)

Thank You