

## Assignment - 1

1. Define the term Compiler:-
- A compiler is a type of translator program that scans the entire program written in a high-level language and translates it as a whole into machine code. That is, the source program into an object program.
- The compiler completes the task of translation in two steps:

In the first step, the compiler reads the source program & converts into the object programs. In this step, if there is a syntax error in the source program, the compiler gives the user an error message to correct it. This error message is called Compiled Time Diagnostic Error Message. Once the program is compiled, there is no need to compile it later.

2. Define the High level Language.

→ The high-level language is a programming language that allows a programmer to write the programs which are independent of a particular type of computer. The high-level languages are considered as high-level because they are closer to human languages than machine-level languages.

When writing a program in a high-level language, then the whole attention needs to be paid to the logic of that the problem. A compiler is required to translate a high-level language into a low-level language.

3. Write a short note on:

a). Machine language :

The machine language is a language that consists of a set of instructions that are in the binary form 0 or 1. As we know that computers can understand only machine instructions, which are in binary digits, i.e., 0 and 1, so the instructions given to the computer can be only in binary codes. Creating a program in a machine language is a very difficult task as it is not easy for the programmers to write the program in machine instructions. It is error-prone as it is not

easy to understand, and its maintenance is also very high. A machine language is not portable as each computer has its machine instructions, so if we write a program in one computer will no longer be valid in another computer.

The different processor architectures use different machine codes, for example, a PowerPC processor contains RISC architecture, which requires different code than Intel x86 processor, which has a CISC architecture.

### b. Assembly Language:

The assembly language contains some human-readable commands such as mov, add, sub, etc. The problems which we were facing in machine-level language are reduced to some extent by using an extended form of machine-level language known as assembly language. Since assembly language instructions are written in English words like mov, add, sub, so it is easier to write and understand.

As we know that computers can only understand the machine-level language instructions, so we require a translator that converts the assembly code into machine code. This translator is known as assembly assembler.

The assembly language code is not portable because the data is stored in computer registers, and the computer has to know the different sets of registers.

The assembly code is not faster than machine code because the assembly language comes above the machine language in the hierarchy, so it means that assembly language has some abstraction from the hardware while machine language has zero abstraction.

### C. Compiler:

A compiler is a type of translator program that scans the entire programs written in high-level language & translates it as a whole into machine code.. That is, the source program into an object program.

The compiler completes the task of translation in two steps -

In the first step, the compiler reads the source program and converts into the object programs. In this step, if there is a syntax error in the source program, the compiler gives the user an error message to correct it. This error message is called Compiled Time Diagnostic Error Message.

functions of the compiler:

- Link the corresponding sub-routine
- Convert source program statements into object programs or machine language
- If there is any syntax error in the program, make a list of errors during translation.
- Identify the range of main memory.

d). Interpreter :

An interpreter like a computer compiler, is a kind of translator program that reads one line of a program written in a high-level language and converts it into machine language and executes it, displaying instant results.

The difference with the compiler is that the compiler converts the entire source program into an object program and provides the latest results but the interpreter converts the source program into a line-by-line object program and displays instant results.

Interpreter translates line-by-line and reports the error code once it encountered during the translation process. It directly executes the operations specified in the source program when the input is given by the user. It gives better error diagnostics than a compiler.

4. Define a computer and write different characteristics of the computer.

→ A computer is an electronic device that receive data in a specific form, & it performs a series of operations according to the pre determined set of instructions & gives a result.

The characteristics of computer are:

a. Speed :

Computers are fast in doing calculations. The speed of the computer is measured in terms of million instructions per second (MIPS)

b. Storage Capacity :

Computers come with large amount of memory. They can hold lot of data. Computers can show a particular piece of information from large amount of data in a short time.

c. Diligence :

After doing work for sometime, human become tired but computer do not become tired. They work continuously. In fact, some computers which control telephone exchanges are never stopped. This is called diligence.

d. Accuracy :

The results that the computers produce are accurate provided data & programs are reliable.

e. Versatility :

We can use computer to perform completely different type of work at the same time.

f. Reliability :

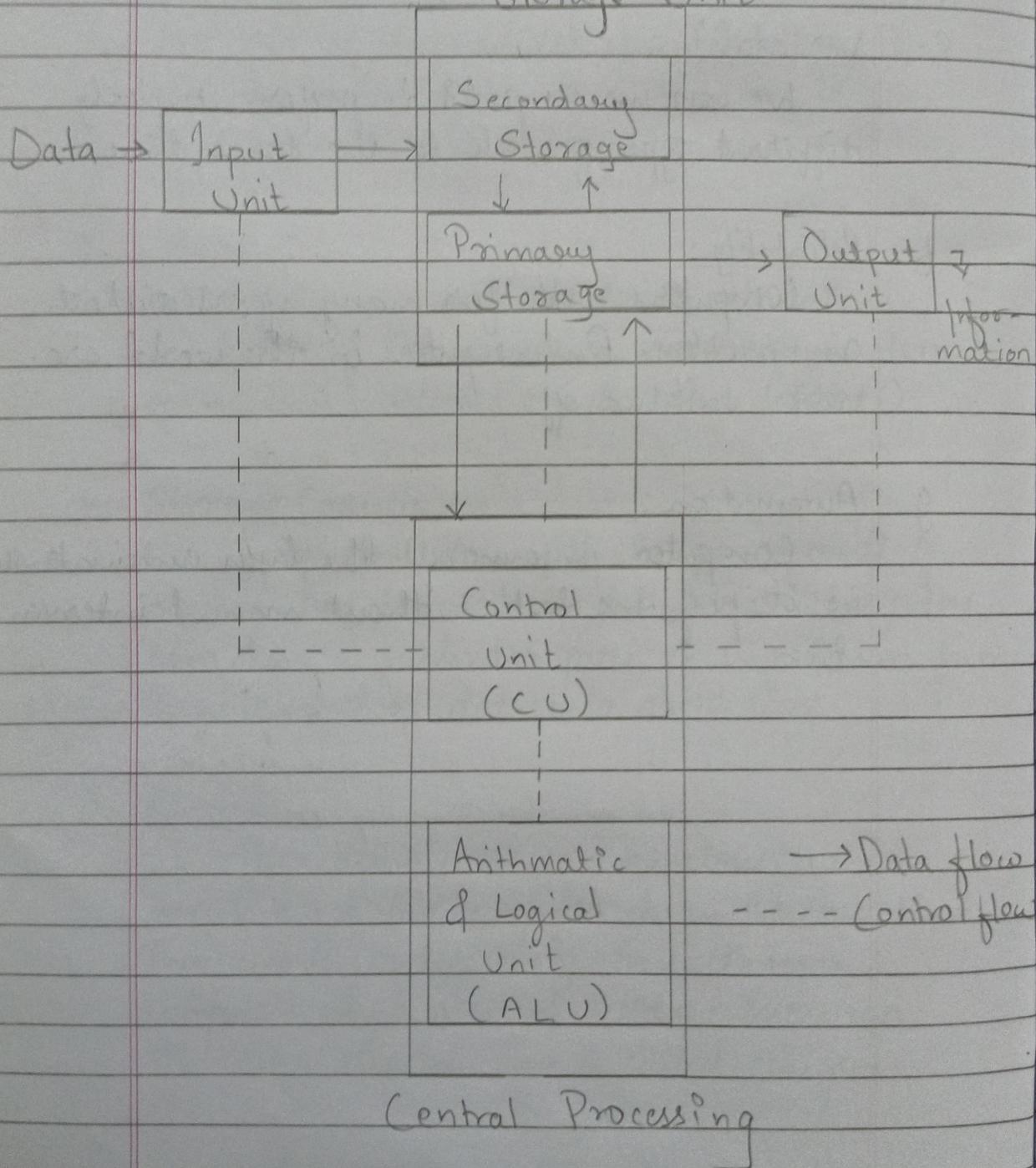
Computers work for many years without any problem. Few computers in the world are never switched off.

g. Automation :

Computer performs all the tasks automatically i.e it performs tasks without manual intervention.

5. Draw a block diagram of the computer system and discuss the functionality of each in detail.

Block diagram of Computer Storage Unit



a). Input:

This is the process of entering data and programs into the computer system. You should know the computer is an electronic machine like any other machine which takes as inputs raw data & performs some processing giving out processed data. Therefore, the input unit takes data from us to the computer in an organized manner of processing.

b). Storage:

The process of storing data & instructions permanently is known as storage. Data has to be fed into the system before the actual processing starts. It is because the processing speed of Central Processing Unit (CPU) is so fast that the data has to be provided to CPU with the same speed. Therefore the data is first stored in the storage unit for faster access & processing. This storage unit or the primary storage of the computer system is designed to do the above functionality. It provides space for storing data and instructions.

c) Processing:

The task of performing operations like arithmetic & logical operations is called processing. The Central Processing Unit (CPU) takes data and instructions from the storage unit & makes all sorts of calculations based on the instructions given and the type of data provided. It is then sent back to the storage unit.

d). Output:

This is the process of producing results from the data for getting useful information. Similarly the output produced by the computer after processing must also be kept somewhere inside the computer before being given to you in human readable form. Again the output is also stored inside the computer for further processing.

e). Control:

The manner how instructions are executed and the above operations are performed. Controlling of all operations like input, processing & output are performed by control unit. It takes care of step by step processing of all operations inside the computer.

### f. Central Processing Unit (CPU):

The ALU and the CU of a computer system are jointly known as the central processing unit. You may call CPU as the brain of any computer system. It is just like brain that takes all major decisions, makes all sorts of calculations and directs different parts of the computer functions by activating and controlling the operations.

### 6. Compare microcomputers, minicomputers and mainframe computers:

#### a). Microcomputers:

A microcomputer is a computer whose CPU is a microprocessor. A microprocessor is a processor whose all components are on a single integrated-circuit chip. Those are normally single-microprocessor, single-user systems designed for performing basic operation like educational, training, small business applications, playing games, etc.

#### b). Minicomputers:

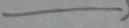
Minicomputers, they are also called mid-range servers, are more powerful computers than micro-computers in terms of processing power & capabilities. Minicomputers are mainly multiuser systems where many users simultaneously

work on the systems. Mini computers possess greater storage capacity and larger memories as compared to microcomputers. These are even capable of handling more input-output devices.

c). Mainframe Computers:

Mainframe computers are designed to handle huge volumes of data and information. These can support more than hundred users at same time. These very large & expensive computers have generate processing speeds & very large storage capacity and memory as compared to minicomputers. These computers even possess and work with more than one processor at the same time. Thus one can say these are multiuser, multiprocessor systems. For mainframe computers very sophisticated operating systems are needed to control and supervise their operation.

7. Compare between high level and low level languages.



## High level language

1). It is programmer friendly language

2). High level language is less memory efficient

3). It is easy to understand

4). It is simple to debug

5). It is simple to maintain

6). It is portable

7). It can run on any platform

8). It needs compiler or interpreter for translation

9). It is used widely for programming

## Low level language

It is machine friendly language

Low level language is high memory efficient

It is tough to understand

It is complex to debug comparatively.

It is complex to maintain comparatively

It is non-portable

It is machine-dependent

It needs assembler for translation

It is not commonly used nowadays in programming.

8. Perform Addition, multiplication, Subtraction & Division.

1).  $11100011 + 1001111$

$$\begin{array}{r} 1110011 \\ + 1001111 \\ \hline 100110010 \end{array}$$

2).  $1111 - 0011$

$$\begin{array}{r} 1111 \\ - 0011 \\ \hline 1100 \end{array}$$

3).  $1111100 \times 11100$

$$\begin{array}{r} 1111100 \\ \times 11100 \\ \hline 0000000 \\ 0,000000X \\ 1111100XX \\ 1111100XXX \\ \hline 1100110010000 \end{array}$$

4.  $\begin{array}{r} 111100 \\ \times 111 \\ \hline 100 \\ 111 \\ \hline 0001 \\ - 0 \\ \hline 10 \\ - 0 \\ \hline 100 \end{array}$

9. Convert following binary numbers to octal, decimal & hexadecimal.

i). 11100011

→ Binary to Octal

11100011

011 100 011  
4+2+1 4+2+1 4+2+1  
3 4 3

= 343

Binary to decimal

11100011

$$= 2^9 + 2^8 + 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0$$

$$= 128 + 64 + 32 + 0 + 0 + 0 + 2 + 1$$

$$= 227$$

Binary to hexadecimal

11100011

$$\begin{array}{r} 1110 \quad 0011 \\ 8421 \quad 8421 \\ = 14 \quad = 3 \end{array}$$

$$= E3$$

2). 1111

→. Binary to octal

001111

$$\begin{array}{r} 001 \quad 111 \\ 4+2+1 \quad 4+2+1 \\ = 1 \quad = 7 \\ = 17 \end{array}$$

Binary to Decimal

1111

$$2^3 + 2^2 + 2^1 + 2^0$$

$$= 8 + 4 + 2 + 1$$

$$= 15$$

Binary to Hexa decimal

1111

$$8 + 4 + 2 + 1$$

$$= 15 = F$$

3). 111100

→ Binary to Octal

111 100

$$4+2+1 \quad 4+2+1$$

$$= 7 \quad = 4$$

$$= 74$$

Binary to Decimal

111100

$$2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0$$

$$= 32 + 16 + 8 + 4 + 2 + 1 \\ = 63$$

Binary to Hexadecimal

111100

$$\begin{array}{r} 0011 \ 1100 \\ 8+4+2+1 \quad 8+4+2+1 \\ = 3 \quad = 12 \\ = 3C \end{array}$$

4. 111100

→. Binary to Octal

$$\begin{array}{r} 001 \ 111 \ 100 \\ 4+2+1 \quad 4+2+1 \quad 4+2+1 \\ = 1 \quad = 7 \quad = 4 \\ = 174 \end{array}$$

Binary to Decimal

$$\begin{array}{r} 111100 \\ = 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0 \\ = 64 + 32 + 16 + 8 + 4 + 2 \\ = 124 \end{array}$$

10. Convert the following decimal numbers into  
to the binary.

1.  $(56)_{10}$

2	56	0	
2	28	0	$(0111000)_2$
2	14	0	
2	7	1 ↑	111001
2	3	1	$= 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0$
2	1	1	$= 32 + 16 + 8 + 0 + 0 + 0$
	0		$= 56$

2. 89

2	89	1	
2	44	0	$(01011001)$
2	22	0	$= 2^7 + 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0$
2	11	1 ↑	$= 64 + 16 + 8 + 1$
2	5	1	$= 89$
2	2	0	
2	1	1	
	0		

3. 57

2	57	1	
2	28	0	(0111001)
2	14	0	
2	7	1 ↑	$= 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0$
2	3	1	$= 32 + 16 + 8 + 4 + 2 + 1$
2	1	1	$= 57$
		0	

4. 87

2	87	1	
2	43	1	(01010111)
2	21	1	
2	10	0	$= 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0$
2	5	1	$= 64 + 0 + 16 + 0 + 4 + 2 + 1$
2	2	0	$= 87.$
2	1	1	
		0	