**1) Explain generation of computer hardware?**

Classification of generations of computers

The evolution of computer technology is often divided into five generations.

**Five Generations of Computers**

**Generations of computers Generations timeline Evolving hardware**

First generation 1940s-1950s Vacuum tube based

Second generation 1950s-1960s Transistor based

Third generation 1960s-1970s Integrated circuit based

Fourth generation 1970s-present Microprocessor based

Fifth generation the present and the future Artificial intelligence based

**First Generation of Computers**

**The main characteristics of first generation of computers (1940s-1950s)**

* Main electronic component – vacuum tube
* Main memory – magnetic drums and magnetic tapes
* Programming language – machine language
* Power – consume a lot of electricity and generate a lot of heat.
* Speed and size – very slow and very large in size (often taking up entire room).
* Input/output devices – punched cards and paper tape.
* Examples – ENIAC, UNIVAC1, IBM 650, IBM 701, etc.
* Quantity – there were about 100 different vacuum tube computers produced between 1942 and1963.

**Second Generation of Computers**

**The main characteristics of second generation of computers (1950s-1960s)**

* Main electronic component – transistor
* Memory – magnetic core and magnetic tape / disk
* Programming language – assembly language
* Power and size – low power consumption, generated less heat, and smaller in size (in comparison with the first generation computers).
* Speed – improvement of speed and reliability (in comparison with the first generation computers).
* Input/output devices – punched cards and magnetic tape.
* Examples **–** IBM 1401, IBM 7090 and 7094, UNIVAC 1107, etc.

**Third Generation of Computers**

**The main characteristics of third generation of computers (1960s-1970s)**

* Main electronic component – integrated circuits (ICs)
* Memory – large magnetic core, magnetic tape / disk
* Programming language – high level language (FORTRAN, BASIC, Pascal, COBOL, C, etc.)
* Size – smaller, cheaper, and more efficient than second generation computers (they were called minicomputers).
* Speed – improvement of speed and reliability (in comparison with the second generation computers).
* Input / output devices – magnetic tape, keyboard, monitor, printer, etc.
* Examples **–** IBM 360, IBM 370, PDP-11, UNIVAC 1108, etc.

**Fourth Generation of Computers**

**The main characteristics of fourth generation of computers (1970s-present)**

* Main electronic component – very large-scale integration (VLSI) and microprocessor.
* VLSI– thousands of transistors on a single microchip.
* Memory – semiconductor memory (such as RAM, ROM, etc.)
* RAM (random-access memory) – a type of data storage (memory element) used in computers that temporary stores of programs and data (volatile: its contents are lost when the computer is turned off).
* ROM (read-only memory) – a type of data storage used in computers that permanently stores data and programs (non-volatile: its contents are retained even when the computer is turned off).
* Programming language – high level language (Python, C#, Java, JavaScript, Rust, Kotlin, etc.).
* A mix of both third- and fourth-generation languages
* Size – smaller, cheaper and more efficient than third generation computers.
* Speed – improvement of speed, accuracy, and reliability (in comparison with the third generation computers).
* Input / output devices – keyboard, pointing devices, optical scanning, monitor printer, etc.
* Network – a group of two or more computer systems linked together.
* Examples **–**IBM PC, STAR 1000, APPLE II, Apple Macintosh, etc.

**Fifth Generation of Computers**

**The main characteristics of fifth generation of computers (the present and the future)**

* Main electronic component: based on artificial intelligence, uses the Ultra Large-Scale Integration (ULSI) technology and parallel processing method.
* **ULSI** – millions of transistors on a single microchip
* **Parallel processing method** – use two or more microprocessors to run tasks simultaneously.
* Language – understand natural language (human language).
* Power – consume less power and generate less heat.
* Speed – remarkable improvement of speed, accuracy and reliability (in comparison with the fourth generation computers).
* Size – portable and small in size, and have a huge storage capacity.
* Input / output device – keyboard, monitor, mouse, trackpad (or touchpad), touchscreen, pen, speech input (recognise voice / speech), light scanner, printer, etc.
* Example **–** desktops, laptops, tablets, Smartphone’s, etc.

**2) Conversion**

(225)10

(N) 2

(N) 16

(N) 8

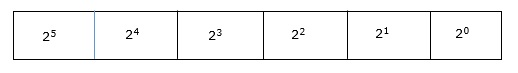
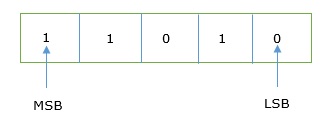
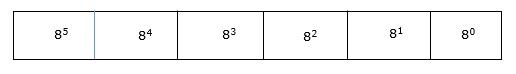
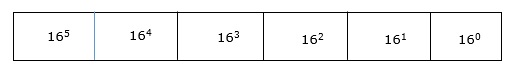
(1101011)2

(N) 10

(N) 8

(N) 16

**3) What is number system? Explain different type of number system?**

* <https://www.tutorialspoint.com/basics_of_computers/basics_of_computers_number_system.htm>
* The technique to represent and work with numbers is called number system. Decimal number system is the most common number system. Other popular number systems include binary number system, octal number system, hexadecimal number system, etc.
* Decimal Number System
* Decimal number system is a base 10 number system having 10 digits from 0 to 9. This means that any numerical quantity can be represented using these 10 digits. Decimal number system is also a positional value system. This means that the value of digits will depend on its position. Let us take an example to understand this.
* Say we have three numbers – 734, 971 and 207. The value of 7 in all three numbers is different−
* In 734, value of 7 is 7 hundreds or 700 or 7 × 100 or 7 × 102
* In 971, value of 7 is 7 tens or 70 or 7 × 10 or 7 × 101
* In 207, value 0f 7 is 7 units or 7 or 7 × 1 or 7 × 100
* The weightage of each position can be represented as follows −
* 
* In digital systems, instructions are given through electric signals; variation is done by varying the voltage of the signal. Having 10 different voltages to implement decimal number system in digital equipment is difficult. So, many number systems that are easier to implement digitally have been developed. Let’s look at them in detail.
* Binary Number System
* The easiest way to vary instructions through electric signals is two-state system – on and off. On is represented as 1 and off as 0, though 0 is not actually no signal but signal at a lower voltage. The number system having just these two digits – 0 and 1 – is called binary number system.
* Each binary digit is also called a bit. Binary number system is also positional value system, where each digit has a value expressed in powers of 2, as displayed here.
* 
* In any binary number, the rightmost digit is called least significant bit (LSB) and leftmost digit is called most significant bit (MSB).
* 
* And decimal equivalent of this number is sum of product of each digit with its positional value.
* 110102 = 1×24 + 1×23 + 0×22 + 1×21 + 0×20
* = 16 + 8 + 0 + 2 + 0
* = 2610
* Computer memory is measured in terms of how many bits it can store. Here is a chart for memory capacity conversion.
* 1 byte (B) = 8 bits
* 1 Kilobytes (KB) = 1024 bytes
* 1 Megabyte (MB) = 1024 KB
* 1 Gigabyte (GB) = 1024 MB
* 1 Terabyte (TB) = 1024 GB
* 1 Exabyte (EB) = 1024 PB
* 1 Zettabyte = 1024 EB
* 1 Yottabyte (YB) = 1024 ZB
* Octal Number System
* Octal number system has eight digits – 0, 1, 2, 3, 4, 5, 6 and 7. Octal number system is also a positional value system with where each digit has its value expressed in powers of 8, as shown here −
* 
* Decimal equivalent of any octal number is sum of product of each digit with its positional value.
* 7268 = 7×82 + 2×81 + 6×80
* = 448 + 16 + 6
* = 47010
* Hexadecimal Number System
* Octal number system has 16 symbols – 0 to 9 and A to F where A is equal to 10, B is equal to 11 and so on till F. Hexadecimal number system is also a positional value system with where each digit has its value expressed in powers of 16, as shown here −
* 
* Decimal equivalent of any hexadecimal number is sum of product of each digit with its positional value.
* 27FB16 = 2×163 + 7×162 + 15×161 + 10×160
* = 8192 + 1792 + 240 +10
* = 1023410
* Number System Relationship
* The following table depicts the relationship between decimal, binary, octal and hexadecimal number systems.

HEXADECIMAL DECIMAL OCTAL BINARY

0 0 0 0000

1 1 1 0001

2 2 2 0010

3 3 3 0011

4 4 4 0100

5 5 5 0101

6 6 6 0110

7 7 7 0111

8 8 10 1000

9 9 11 1001

A 10 12 1010

B 11 13 1011

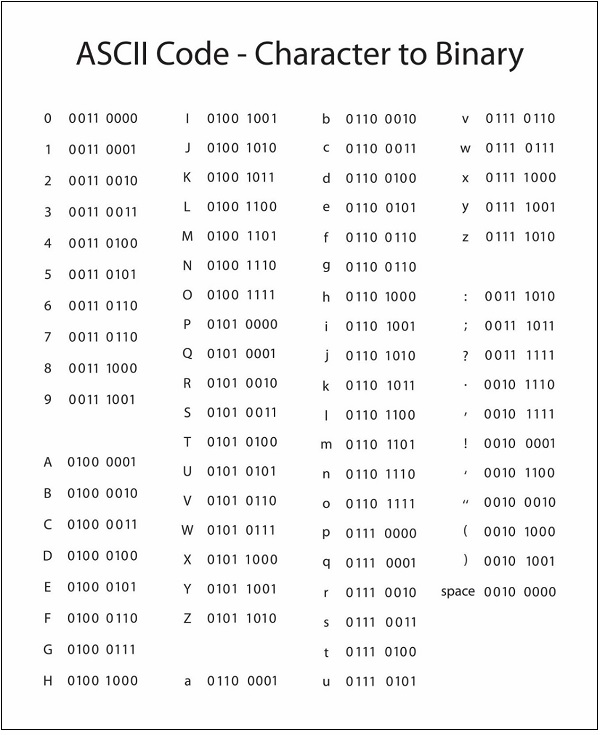
C 12 14 1100

D 13 15 1101

E 14 16 1110

F 15 17 1111

**ASCII**

* Besides numerical data, computer must be able to handle alphabets, punctuation marks, mathematical operators, special symbols, etc. that form the complete character set of English language. The complete set of characters or symbols are called alphanumeric codes. The complete alphanumeric code typically includes −
* 26 upper case letters
* 26 lower case letters
* 10 digits
* 7 punctuation marks
* 20 to 40 special characters
* Now a computer understands only numeric values, whatever the number system used. So all characters must have a numeric equivalent called the alphanumeric code. The most widely used alphanumeric code is American Standard Code for Information Interchange (ASCII). ASCII is a 7-bit code that has 128 (27) possible codes.
* 
* ISCII
* ISCII stands for Indian Script Code for Information Interchange. IISCII was developed to support Indian languages on computer. Language supported by IISCI include Devanagari, Tamil, Bangla, Gujarati, Gurmukhi, Tamil, Telugu, etc. IISCI is mostly used by government departments and before it could catch on, a new universal encoding standard called Unicode was introduced.
* Unicode
* Unicode is an international coding system designed to be used with different language scripts. Each character or symbol is assigned a unique numeric value, largely within the framework of ASCII. Earlier, each script had its own encoding system, which could conflict with each other.
* In contrast, this is what Unicode officially aims to do − Unicode provides a unique number for every character, no matter what the platform, no matter what the program, no matter what the language.