# NUMBER SYSTEM AND REAL WORLD

## Introduction and Binary Number System

In computing, different number systems are used to represent and process information. The most fundamental number system used in computing is the binary number system. Binary is a base-2 number system that uses only two digits: 0 and 1. This number system is used in machine-level programming, where instructions and data are represented as binary codes.

For example, the binary code 10101010 can represent the decimal number 170. In machine-level programming, this binary code can be used to perform arithmetic operations, such as addition and subtraction.

## Binary Number System in Computing

The binary number system is the basis of computing and is used to represent information in computers. It is a number system that uses only two digits: 0 and 1. This is in contrast to other number systems, such as the decimal system, which uses 10 digits from 0 to 9.

***Key Points of Binary Number System in Computing***

Binary is the basis of computing: The binary number system is used to represent information in computers.

***Two digits 0 and 1*:** Binary uses only two digits, 0 and 1, to represent information.

***Simple and efficient*:** Binary is a simple system that is easy for computers to understand and process, and it can be stored and transmitted efficiently.

***Used in Programming, Data Storage, and Networking:*** Binary is used in a wide range of applications, including computer programming, data storage, and networking.

## Binary in Machine Level Language

Machine level language, also known as machine code, is the lowest-level language that a computer's processor can understand. It is a binary code that uses a series of 0s and 1s to represent instructions and data.

***Use of Binary in Machine Level Language:***

In machine level language, binary is used to represent:

***Instructions***: Binary codes are used to represent instructions that the processor can execute, such as arithmetic operations, jumps, and loads.

***Data:*** Binary is used to represent data, such as numbers, characters, and addresses.

***Addresses:*** Binary is used to represent memory addresses, which are used to access data and instructions.

## The Role of Hexadecimal in Memory Addressing

In computer systems, hexadecimal notation plays a crucial role in memory addressing. Hexadecimal is a base-16 number system that uses 16 distinct symbols, including 0-9 and A-F, to represent values. In memory addressing, hexadecimal is used to represent memory addresses in a more human-readable format than binary. Each byte of memory is assigned a unique address, which is typically represented as a hexadecimal value. For example, the memory address 0x1000 might represent the starting address of a program, while 0x2000 might represent the address of a specific data structure. Hexadecimal notation is used because it is more compact and easier to read than binary, making it easier for programmers to work with memory addresses. Additionally, hexadecimal notation can be easily converted to and from binary, making it a convenient intermediate representation for memory addresses. As a result, hexadecimal has become a standard notation for representing memory addresses in programming and computer systems.

## Situations Where Octal is Preferred

Octal notation is preferred in certain situations due to its unique properties. One such situation is when working with bitwise operations, where octal's base-8 system aligns perfectly with the 3-bit groups of binary data. This makes it easier to perform bitwise operations, such as masking and shifting, as the octal representation can be easily manipulated to achieve the desired result. Additionally, octal is preferred in situations where a compact and readable representation of binary data is required, such as in debugging and logging applications. In these cases, octal's concise notation and ease of conversion to and from binary make it a more convenient choice than hexadecimal or binary notation. Overall, octal's advantages in bitwise operations and compact representation make it a preferred choice in specific situations.

## Comparison and Contrast

In conclusion, binary, octal, and hexadecimal number systems are used in computing for different purposes. Binary is used in machine-level programming, where instructions and data are represented as binary codes. Octal is preferred in situations where a compact representation of binary data is needed, such as in Unix file permissions. Hexadecimal is widely used in memory addressing, where it is used to represent memory addresses.

***The advantages of binary number system are:***

* It is the most fundamental number system used in computing.
* It is used in machine-level programming, where instructions and data are represented as binary codes.

***The limitations of binary number system are:***

* It is difficult to read and write binary codes.
* It is not compact, as it requires a large number of digits to represent a decimal number.

***The advantages of octal number system are:***

* It is compact, as it requires fewer digits to represent a decimal number.
* It is preferred in situations where a compact representation of binary data is needed.

***The limitations of octal number system are:***

* It is not widely used in computing.
* It is not as fundamental as binary number system.

***The advantages of hexadecimal number system are:***

* It is widely used in memory addressing, where it is used to represent memory addresses.
* It is compact, as it requires fewer digits to represent a decimal number.

***The limitations of hexadecimal number system are:***

* It is not as fundamental as binary number system.
* It is not as compact as octal number system.

## Summary

In summary, each number system has its advantages and limitations, and is used in computing for different purposes.