**컴퓨터에서의 수 표현 개요**

* **컴퓨터의 수 표현 방식**: 컴퓨터는 **10진수**가 아닌 **2진수**로 수를 표현한다.
* **문서의 목적**: 이 문서는 일반적인 컴퓨터가 정수와 실수를 어떻게 표현하는지를 정리한 것이다.

**정수 표현 방법**

* **정수형 종류**: 일반적으로 컴퓨터에서 사용되는 정수형의 종류는 다음과 같다.
  1. **8비트**:
     + **부호 여부**: 없음
     + **범위**: [0, 2^8 - 1]
     + **어셈블리 자료형**: unsigned char
  2. **16비트**:
     + **부호 여부**: 있음
     + **범위**: [-2^15, 2^15 - 1]
     + **어셈블리 자료형**: short
  3. **32비트**:
     + **부호 여부**: 있음
     + **범위**: [-2^31, 2^31 - 1]
     + **어셈블리 자료형**: int
  4. **64비트**:
     + **부호 여부**: 있음
     + **범위**: [-2^63, 2^63 - 1]
     + **어셈블리 자료형**: long long
* **정수형의 크기**: 정수형의 크기는 운영체제, CPU 아키텍처, 프로그래밍 언어에 따라 다를 수 있다.

**메모리에 저장하는 방법**

* **메모리 사용**: 메모리에서는 정수 데이터를 저장하기 위해 **4칸**을 사용한다.
* **바이트 단위**: 바이트는 컴퓨터가 정보를 저장하는 가장 작은 단위이며, 메모리 상에서 주소가 배정될 수 있는 가장 작은 단위이다.
* **64비트 수 체계**: 64비트 수 체계는 8칸을 차지한다.

**정수의 음수 표현 방식**

* **부호 비트**: 총 32개의 비트 중 첫 번째 비트는 부호 표현을 위해 따로 배정된다.
  + **부호 비트의 역할**: 부호 비트가 **0**이면 양수를, **1**이면 음수를 나타낸다.
* **음수 표현 방법**: 음수를 표현하는 방법은 여러 가지가 있다.
  + **부호화 절대치 (Sign Magnitude)**: 부호 비트를 제외한 수를 양수로 읽고,

마이너스를 붙이는 방법이다.

* + **1의 보수 (1's Complement)**: 양수의 비트를 반전시켜 음수를 표현하는 방법이다.
  + **2의 보수 (2's Complement)**: 1의 보수 방법에 1을 더하는 방법이다.

**부동 소수점 표현 방식**

* **부동 소수점의 정의**: 부동 소수점 방식은 넓은 범위의 수를 표현할 수 있으며, 높은 정밀성을 보장한다.
* **부동 소수점 연산**: 과거에는 부동 소수 연산을 위해 별도의 프로세서가 필요했으나, 현재 대부분의 컴퓨터가 기본적으로 부동 소수점 방식을 사용한다.

**IEEE 754 표준**

* **표준의 정의**: IEEE Floating Point Standard (IEEE 754)는 1985년 IEEE에서 공표한 부동 소수점 방식의 표준안이다.
* **표준의 중요성**: 이 표준은 오랫동안 거의 유일하게 널리 사용된 방식이며, 현재에도 많은 컴퓨터 시스템에서 실수값을 표현하는 데 사용된다.
* **표준의 구성**: IEEE 754는 다음과 같은 내용을 포함한다.
  + **산술 형식**: 일반적인 실수값과 양의 무한대, 음의 무한대, NaN 등의 표현형식.
  + **교환 형식**: 부동 소수점 데이터를 교환할 때 사용할 수 있는 효율적이고 간편한 인코딩 방식.
  + **반올림 규칙**: 산술적인 계산이나 변환 과정에서 반올림할 때 지켜야 할 성질.

**복소수 표현 방법**

* **복소수 구현**: 복소수를 구현하는 표준 방법은 정의되지 않았다.
* **구조체 사용**: 복소수를 구현하려면 실수와 허수를 각각 저장할 수 있는 구조체를 만들 수 있다.
* **연산 방법**: 복소수와의 연산은 실수를 bbb가 000인 복소수로 취급하고 수행한다.

참조 : https://namu.wiki/w/%EC%BB%B4%ED%93%A8%ED%84%B0%EC%97%90%EC%84%9C%EC%9D%98%20%EC%88%98%20%ED%91%9C%ED%98%84