
 Devising a way to storing **signed** integer numbers inside a computer

- A "straight forward" encoding for signed numbers is the **sign-magnitude** encoding method:

```

  0 0 0 0 0 1 1 1
  ^
  |
  sign      Remaining bits is the magnitude (absolute value)
  bit
  
```

Examples:

```

  00000111    represents 7
  10000111    represents -7
  
```

- Problems with sign-magnitude encoding:
 - 2 representation for ZERO: 00000000 (+0) and 10000000 (-0)
 - Meaning of operation depends on value, e.g., $7 + (-3)$ is evaluated using a subtraction
- Computers **now** use the **two's complement encoding** method to represent **signed** integer numbers.
 - The reason that two's complement encoding is used is for efficiency.
 - Only 1 representation for ZERO
 - Computer does not need to make any decision to perform **add** and **subtract**
- The best way to understand the two's complement encoding is to consider the decimal analogy: **ten's complement encoding**
- **10's complement encoding:**
 - Lecture notes on 10s complement encoding: [click here](#)

-
- Back to representing signed numbers in computers - **2s complement** encoding:
 - Lecture notes on 2s complement encoding: [click here](#)
 - Another binary number encoding for **signed numbers** is **excess 2^{n-1}** encoding: [click here](#)
-

- **How to tell which representation method is used**

- **Question:**

- How can we tell if **11111111** represents:
 - 255 (using the **unsigned binary number representation**)
 - -1 (using the **signed binary number representation**)

- **Answer: context**
-

- Example (I have to do it in C++ because Java does not have *unsigned* numbers)

```

short x = -1;           // signed numeric variable in C++
                        // It stores 11111111 11111111 in x

cout << x;             // Prints -1

unsigned short y;     // unsigned numeric variable in C++

y = x;                // Stores 11111111 11111111 in y

cout << y;            // Prints 65535 (2^16 - 1)

```

The **variable type information** gives the **context** to the program on *how to interpret* the **representation**

- **Example Program:** (Demo above code)

Example

- Prog file: [click here](#)

- **An English language example of context**

- **Question:**

```

■ Make a correct English sentence that start with:

    You is ....

```

Answer:

```

    You is an English word.

```

- If you have a hard time coming up with such a sentence, it is because you have **always taught** that **"you"** is a **personal pronoun**

```

■ The context in which you used the word "you" has always been as a personal pronoun

```

- You was **not aware** that the word **you** can be used in a **different context**.
-
-

○ **Joke:**

```
TEACHER: Millie, give me a sentence starting with 'I. '  
MILLIE: I is..  
TEACHER: No, Millie..... Always say, 'I am.'  
MILLIE: All right... 'I am the ninth letter of the alphabet.'
```
