

C프로그래밍 과제 #1 (변수, 연산자, 데이터 저장 방식)

학번 : 20180018 이름 : 강승아

1. C 언어에서 정수 값은 2의 보수 표기법으로 표현된다. 다음 각 10진수를 8비트 변수(실제로는 char 타입)로 저장한다면 비트열의 값이 어떻게 되는지 기술해 보라. 단, 2)번 문제의 경우 어떻게 구했는지 그 과정을 함께 기술하라.

1) 12 : 00001100

2) -12 : 11110100

=> 정수를 음수 형으로 표현하기 위해서는 1의 보수를 취한 후 1을 더해준다. 이와 같이 하였을 경우 음수 표현이 가능하다.

EX)

2 : 00000010

-2 : 11111110 (2의 표현 값에 1의 보수를 취하고 1을 더한 값)

이때 두 숫자를 합할 경우 목표하던 0의 값인 00000000이라는 결과 값이 나온다.

3) 1 : 00000001

4) -1 : 11111111

5) 0 : 00000000

6) 127 : 01111111

7) -128 : 10000000

2. 다음은 8비트로 정수 값을 표현하기 위한 2의 보수 표현이다. 각각 10진수 값이 무엇인지 기술해 보라. 단, 2)번 문제의 경우 어떻게 구했는지 그 과정을 기술하라.

1) 00001111 : 15

2) 11110001 : -15

=> 8비트의 맨 앞 숫자는 해당 값의 부호를 결정한다. 0일 경우 양수, 1일 경우 음수를 의미한다. 이때 음수일 경우 1을 빼준 후 2의 보수를 취하여 해당 값의 절댓값을 구한 후 음수 부호를 붙여준다.

풀이과정)

1. 11110001에서 00000001을 뺀다.

2. 11110000에서 2의 보수를 취한다.

3. 00001111의 값을 구한다. -> 15

4. -부호를 붙여 최종 값을 도출한다. => '-15'

3) 01000101 : 69

4) 10111010 : -57

5) 00000000 : 0

6) 10000000 : -127

7) 11111111 : -1

3. C 언어에서 실수값은 부호, 지수, 가수 부분으로 저장되는데, float(32비트) 타입의 경우 부호(1비트), 지수(8비트), 가수(23비트)로 저장된다. 다음 10진수 실수값을 float 타입의 변수로 저장한다면 비트열의 값이 어떻게 되는지 기술해 보라. 단, 2)번 문제의 경우 그 과정을 자세히 설명해 보라. 본인이 생각한 답이 맞는지는 수업 자료 “(C-04) Ch02_1_데이터 표현방식의 이해.pptx”의 16페이지 프로그램을 통해 확인할 수 있다.

1) -13.25 : $-1101.01 \Rightarrow -1.10101 \times 2^3 \Rightarrow 1\ 10000010\ 1010100000000000000000$

2) 85.8125 : 1010101.1101 (2진수로 변환)

$\Rightarrow 1.0101011101 \times 2^6 \Rightarrow 00000110$ 에 127 더하기 $\Rightarrow 10000101$ 의 지수

$\Rightarrow 0\ 10000101\ 010101\ 110100000000000000$

3) -85.8125 : $-1010101.1101 \Rightarrow -1.0101011101 \times 2^6$

$\Rightarrow 1\ 10000101\ 010101110100000000000000$

4. (프로그래밍) 사용자로부터 다섯 자리 자연수를 입력받아 각 자리 숫자를 차례로 출력해 보라. 예를 들어, 사용자가 56034라고 입력한다면 5, 6, 0, 3, 4를 각각 출력할 수 있어야 한다. 연산자를 적절히 사용하면 된다.

<프로그램 코드>

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    int num[5];
```

```
    int input = 0;
```

```
    while (!(10000 <= input && input <= 99999)) // 다섯자리 숫자가 아닐 경우 재입력
```

```
    {
```

```
        printf("다섯자리 자연수 입력 : ");
```

```
        scanf("%d", &input);
```

```
    }
```

```
    num[0] = input / 10000;
```

```
    input -= 10000 * num[0];
```

```
    num[1] = input / 1000;
```

```
    input -= 1000 * num[1];
```

```
    num[2] = input / 100;
```

```
    input -= 100 * num[2];
```

```
    num[3] = input / 10;
```

```
    input -= 10 * num[3];
```

```
    num[4] = input;
```

```
    for (int i = 0; i < 5; i++)
```

```
        printf("%d \n", num[i]);

    return 0;

}
```

<실행 화면>

