

structure
C/C++



pointer



function



array[]



switch/case

for, while

프로그래밍 기초



malloc/free



if else

Lab 7: 포인터 기초

Lab 1. 두 개의 정수의 합과 차를 동시에 반환하는 함수 구현

■ 포인터 활용

- 파일 이름: lab07_01.c
- C 언어 함수는 하나의 값만 리턴 가능
- 여러 개의 값을 반환해야 되는 경우 포인터 변수를 사용함
 - 변수의 주소를 활용
 - 함수 내부에서 연산한 결과를 main() 함수에서 사용 가능
- main() 함수 기능
 - 두 정수를 화면에서 입력 받고 get_sum_diff()에 전달
 - get_sum_diff() 함수에서 수행한 합(add)과 차(subtract)의 연산 결과를 받아옴
- `void get_sum_diff(int x, int y, int *p_sum, int *p_diff)`를 구현
 - 두 정수(x, y)의 합을 p_sum에 저장
 - 두 정수(x, y)의 차를 p_diff에 저장

Lab 1. 두 개의 정수의 합과 차를 동시에 반환하는 함수 구현

```
#include <stdio.h>

void get_sum_diff(int x, int y, int *p_sum, int *p_diff);

int main()
{
    int sum = 0, diff = 0;
    int x = 0, y = 0;

    // 화면 입력 코드 추가
    . . .

    get_sum_diff( ); // 파라미터 구성

    printf("sum = %d\n", sum);
    printf("diff= %d\n", diff);

    return 0;
}
```

```
void get_sum_diff(int x, int y, int *p_sum, int *p_diff)
{
    // 함수 내부 구현
}

```

실행 결과

```
Input two integers: 100 200
x: 100, y: 200
sum = 300
diff= -100
```

Lab 2. 정수형 변수의 값을 char*를 활용해서 1 바이트씩 출력

■ 정수형 변수의 값을 1바이트씩 주소와 값을 출력하시오.

- 파일 이름: lab07_02.c
- `int value = 0x41424344; // ABCD의 ASCII 코드 hexa 값`
- `char *ch = (char *)&value;`

Hex	Decimal	Char
40	64	@
41	65	A
42	66	B
43	67	C
44	68	D

실행 결과

```
value: 0x41424344
ch+0: 0x16f772ef8, D
ch+1: 0x16f772ef9, C
ch+2: 0x16f772efa, B
ch+3: 0x16f772efb, A
```

0x41424344

Little endian

0xf8	0x44	0100 0100	'D'
0xf9	0x43	0100 0011	'C'
0xfa	0x42	0100 0010	'B'
0xfb	0x41	0100 0001	'A'
...		...	

주소 증가