

1. 예제

I . 예제_12.1

- 함수에 두 개의 정수를 전달한 후 전달한 두개의 정수를 교환하는
함수 swapVal함수와 swapAdr함수 비교 예제

```
#include <stdio.h>

void swapVal(int x, int y);
void swapAdr(int *p1, int *p2);

int main(void)
{
    int a=3, b=5;

    swapVal(a,b);    // Call-by-Value
    printf("swapVal함수 호출 후 %d %d\n", a, b); // 값 바뀌지 않음

    swapAdr(&a, &b); // Call-by-Reference
    printf("swapAdr함수 호출 후 %d %d\n", a, b); // 값 바뀜

    return 0;
}

void swapVal(int x, int y)
{
    int temp;

    temp = x;
    x = y;
    y = temp;
}

void swapAdr(int *p1, int *p2)
{
    int temp;
    temp = *p1;
    *p1 = *p2;
    *p2 = temp;
}
```

II . 예제_12.2

- 배열에 저장된 데이터를 출력하는 함수 예제

```
#include <stdio.h>

void output(int *p);

int main(void)
{
    int arr[5] = {1, 2, 3, 4, 5};

    output(arr);

    return 0;
}

void output(int *p)
{
    int i;

    for(i=0 ; i<5 ; i++)
    {
        //printf("%d ", p[i]);
        printf("%d ", *(p+i));
    }

    printf("\n");
}
```

2. 문제

I . 문제_12.1

- 정수를 하나 입력받아 변수 a에 저장 후 입력받은 값의 제곱을 계산하여 다시 a에 대입해주는 프로그램 작성
- 입력받은 값의 제곱을 계산하는 함수(Square)를 독립적으로 구현
- main함수는 사용자 정의 함수를 호출하여 결과 값을 받아 출력

① Call-by-Value 방식으로 구현

- square 함수 호출 시 변수 a를 전달
- 함수는 제곱 값을 계산하여 반환, 반환 값을 변수 a에 다시 저장
- 함수의 호출 형태 : a = Square(a);

```
정수 입력: 10
연산 결과 : 100
```

```
계속하려면 아무 키나 누르십시오 . . .
```

② Call-by-Reference 방식으로 구현

- square 함수 호출 시 변수 a의 주소 값을 전달
- 함수는 주소 값을 참조해서 변수 a의 값을 알아낸 다음 제곱을 계산하여 변수 a의 값을 변경
- 함수의 호출 형태 : Square(&a);

정수 입력: **10**

연산 결과 : **100**

계속하려면 아무 키나 누르십시오 . . . █