



C/C++

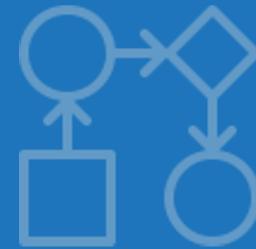
structure



pointer



function



array[]



switch/case



if else

프로그래밍 기초

Lab 8: 포인터와 배열

Lab 1. 1차원 배열의 합 계산

- 포인터 및 1차원 배열 활용

- 파일 이름: lab08_01.c
- `int sum_array(int *array, int size)` 구현
 - 전달받은 배열 array의 각 원소를 한 줄에 화면 출력
 - 1차원 배열의 모든 원소의 합을 구하고 합을 리턴하는 함수
- `main()` 함수
 - `int data[] = {10, 20, 30, 40, 50, 60, 70, 80, 90}`
 - 1차원 배열을 `sum_array()`에 전달하고 배열의 합을 화면에 출력
- 출력 결과

10 20 30 40 50 60 70 80 90

배열의 합계: 450

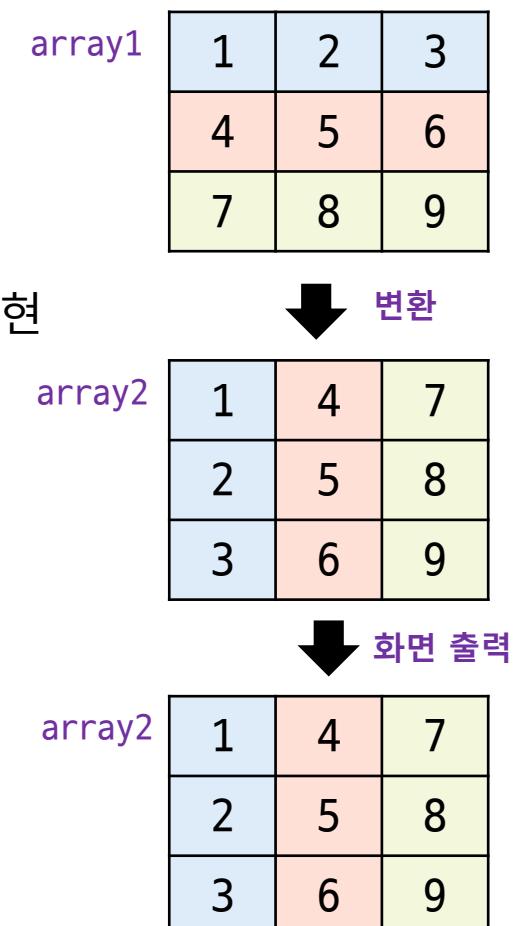
sum_array() 내부에서 출력

main() 함수에서 출력

Lab 2. 2차원 배열의 값 변환

■ 포인터 및 2차원 배열 활용

- 파일 이름: lab08_02.c
- `void input_value(int (*array)[3], int row, int col)` 구현
 - 3x3 크기의 배열에 1~9까지의 값을 순서대로 저장
- `void transform(int (*array1)[3], int (*array2)[3], int row, int col)` 구현
 - 기능: array1의 [행][열]을 array2의 [열][행]으로 변환
 - 반드시 반복문을 사용해서 변환
 - 함수 내부에서는 **포인터로 배열의 원소에 접근**
 - 하드코딩 하지 마세요.**
- `void print_array(int (*array)[3], int row, int col)` 구현
 - 배열의 원소를 배열 형태(3x3)로 화면 출력
 - 자리 수 맞춤
- `main()` 함수 기능
 - `int array1[3][3] = {0}; int array2[3][3] = {0};` 초기화
 - `input_value()` 호출 -> `transform()` 호출 -> `print_array()` 호출 후 결과 출력



Lab 2. 2차원 배열의 값 변환

```
*****  
* Lab 8.2 (11장. 포인터 및 배열)  
* - 2차원 배열을 함수로 전달 후 배열의 값 변경 및 화면 출력  
*****  
  
#include <stdio.h>  
  
void input_value(int (*array)[3], int row, int col)  
{  
  
}  
  
void transform(int (*array1)[3], int (*array2)[3], int row, int col)  
{  
  
}  
  
void print_array(int (*array)[3], int row, int col)  
{  
  
}
```

```
int main()  
{  
    int array1[3][3] = {0};  
    int array2[3][3] = {0};  
  
    printf("Array Initialization(1~9)\n");  
    input_value(array1, 3, 3);  
    print_array(array1, 3, 3);  
  
    printf("Swap rows and columns of array1\n");  
    transform(array1, array2, 3, 3);  
  
    printf("Print array2\n");  
    print_array(array2, 3, 3);  
  
    return 0;  
}
```

Array Initialization(1~9)

1	2	3
4	5	6
7	8	9

Swap rows and columns of array1

Print array2

1	4	7
2	5	8
3	6	9