

함수와 배열 연습 문제

연습문제 1

- 배열에 1부터 10까지의 숫자를 임의의 순서로 선언
- 배열을 넘겨 모든 요소를 출력하는 함수 printAllElements()를 구현하라
- 인자로 넘긴 배열에서 최대값을 찾아 반환하는 함수 findMax()를 구현하라
- 인자로 넘긴 배열에서 최소값을 찾아 반환하는 함수 findMin()를 구현하라
- 인자로 넘긴 배열을 오름차순으로 정렬하는 함수 sortByAscending()를 구현하라
- 인자로 넘긴 배열을 내림차순으로 정렬하는 함수 sortByDescending()를 구현하라

```
// 예시 코드)
#include <stdio.h>

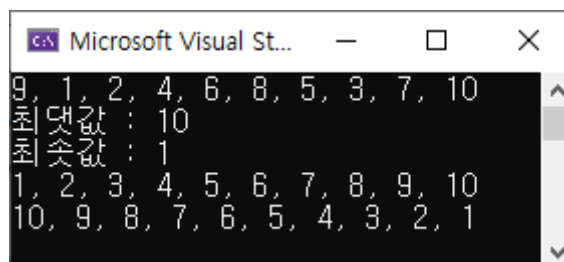
void printAllElements(int []);
int findMax(int[]);
int findMin(int[]);
void sortByAscending(int[]);
void sortByDescending(int[]);

void main() {
    int arr[10] = { 9,1,2,4,6,8,5,3,7,10 };

    ...

}
```

실행 결과



```
Microsoft Visual St...
9, 1, 2, 4, 6, 8, 5, 3, 7, 10
최대값 : 10
최소값 : 1
1, 2, 3, 4, 5, 6, 7, 8, 9, 10
10, 9, 8, 7, 6, 5, 4, 3, 2, 1
```

연습문제 2

- 일차원 배열에 1부터 10까지의 임의의 숫자들을 저장
- 사용자로부터 임의의 숫자를 입력받기
- 인자로 배열과 입력받은 숫자를 넘겨 해당 숫자가 몇 번 나왔는지 반환하는 `getFrequency()` 함수를 구현하라
- 인자로 넘긴 배열에서 각 숫자가 몇 번씩 나왔는지를 출력하는 `printEachFrequency()` 함수를 구현하라
- 인자로 넘긴 배열에서 가장 많은 빈도가 나온 숫자를 **모두** 출력하는 `printMostFrequent()` 함수를 구현하라

```
// 예시 코드)
#include <stdio.h>

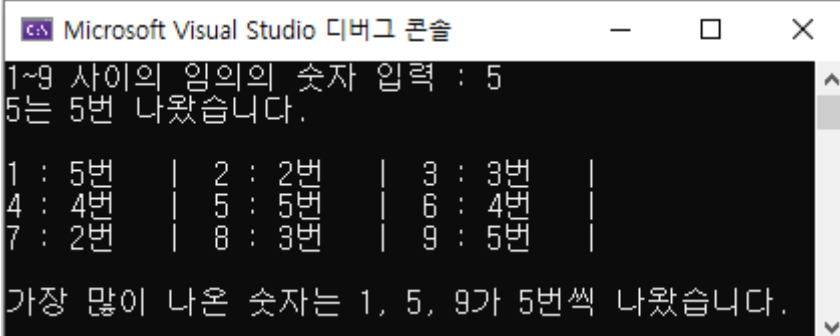
int getFrequency(int, int[]);
void printEachFrequency(int[]);
void printMostFrequent(int[]);

void main() {
    int ary[] = {
1,3,4,5,6,7,8,9,2,8,2,5,9,1,5,9,1,9,6,3,4,1,5,4,6,7,9,8,5,1,3,4,6 };

    ...

}
```

실행 결과



```
Microsoft Visual Studio 디버그 콘솔
1~9 사이의 임의의 숫자 입력 : 5
5는 5번 나왔습니다.

1 : 5번   | 2 : 2번   | 3 : 3번   |
4 : 4번   | 5 : 5번   | 6 : 4번   |
7 : 2번   | 8 : 3번   | 9 : 5번   |

가장 많이 나온 숫자는 1, 5, 9가 5번씩 나왔습니다.
```

연습문제 3

- 인자로 넘긴 두 일차원 배열을 비교해 같다, 다르다를 출력하는 함수 compareArray() 를 구현하라
- 인자로 넘긴 두 이차원 배열을 비교해 같다, 다르다를 출력하는 함수 compareArray2() 를 구현하라

```
// 예시 코드)
#include <stdio.h>
#define SIZE 4

____ compareArray(____);
____ compareArray2(____);

void main() {
    int aryA1[] = { 1,3,5,7,9 };
    int aryA2[] = { 1,3,5,7,9,10 };

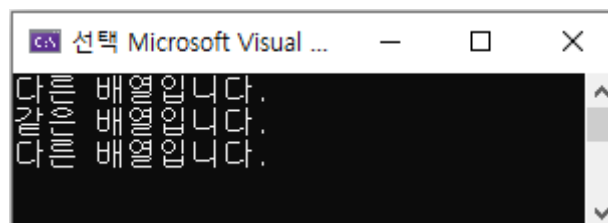
    int aryB1[] = { 10,20,30,40 };
    int aryB2[] = { 10,20,30,40 };

    int aryC1[][SIZE] = { {10,30,20,40}, {1,3,2,4} };
    int aryC2[][SIZE] = { {10,30,20,40}, {1,3,2,4},{0} };

    ...

}
```

실행 결과



연습문제 4

- 2차원 배열 A, B를 선언
- 인자로 넘긴 두 이차원 배열에 대해 행렬의 덧셈 연산을 수행하는 addArray() 함수를 구현하라
- 인자로 넘긴 두 이차원 배열에 대해 행렬의 뺄셈 연산을 수행하는 subArray() 함수를 구현하라
- 인자로 넘긴 이차원 배열을 행에 따라 개행하여 출력하는 printArray() 함수를 구현하라

```
// 예시 코드)
#include <stdio.h>
#define SIZE 4

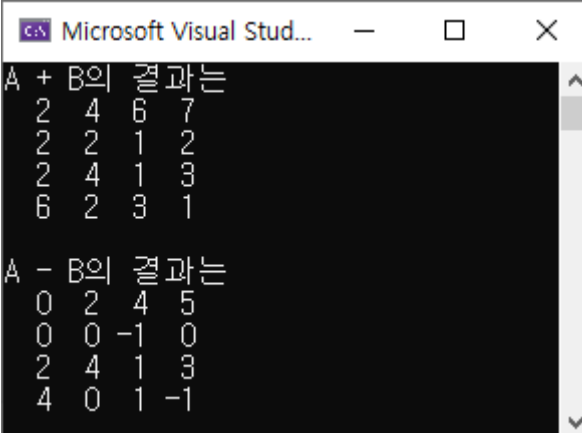
void addArray(int A[][SIZE], int B[][SIZE], int result[][SIZE]);
void subArray(int A[][SIZE], int B[][SIZE], int result[][SIZE]);
void printArray(int result[][SIZE]);

int main() {
    int A[][SIZE] = {
        {1,3,5,6},
        {1,1,0,1},
        {2,4,1,3},
        {5,1,2,0}
    };
    int B[][SIZE] = {
        {1,1,1,1},
        {1,1,1,1},
        {0,0,0,0},
        {1,1,1,1}
    };
    int result[][SIZE] = {0};

    ...

}
```

실행 결과



```
Microsoft Visual Stud...
A + B의 결과는
2 4 6 7
2 2 1 2
2 4 1 3
6 2 3 1

A - B의 결과는
0 2 4 5
0 0 -1 0
2 4 1 3
4 0 1 -1
```

연습문제 5

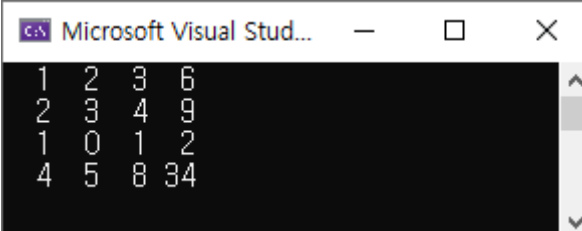
- [4][4] 크기의 배열에 [3][3]의 숫자들을 저장
- 이 배열을 인자로 넘겨 가로 합, 세로 합을 각각 계산하여 넣는 함수 calArray()를 구현하라
 - 각 줄에 해당하는 값들을 모두 더해 그 줄의 마지막 열에 저장하고, 각 열에 해당하는 값들을 모두 더해 그 열의 마지막 행에 저장한다.
- 인자로 최종 결과 배열을 넘겨 개행하여 출력하는 printArray()를 구현하라

```
1 2 3 _  
2 3 4 _  
1 0 1 _  
_ _ _ _   이렇게 4x4 행렬에 3x3 값이 주어진 경우
```

가로 덧셈 1+2+3의 값을 _에 넣고, 2+3+4의 값을 _에 넣고, 1+0+1의 값을 _에 넣는다.
세로 덧셈 1+2+1의 값을 _에 넣고, 2+3+0의 값을 _에 넣고, 3+4+1의 값을 _에 넣는다.
마지막으로 각 더한 _의 값들을 모두 더해 우측 하단의 _에 저장한다.

```
// 예시 코드)  
#include <stdio.h>  
#define SIZE 4  
  
void calArray(int[][SIZE], int);  
void printArray(int[][SIZE], int);  
  
int main() {  
    int arr[SIZE][SIZE] = {  
        {1,2,3},  
        {2,3,4},  
        {1,0,1}  
    };  
  
    ...  
  
}
```

실행 결과



```
Microsoft Visual Stud...  
1 2 3 6  
2 3 4 9  
1 0 1 2  
4 5 8 34
```