

# C++ & Algorithm Seminar

## 3. 1차원 배열과 2차원 배열

SHIN Hyun (kyaryunha)

한양대학교 컴퓨터소프트웨어학부 19학번  
전기전자컴퓨터 학술 동아리 바라미 27기

April 15, 2022

# Contents

## 1 Explanation - 배열

### ■ 배열이란

## 2 Problem - 1차원 배열

### ■ BOJ 10818 최소, 최대

### ■ BOJ 11659 구간 합 구하기4

## 3 Explanation - 2차원 배열

## 4 Problem - 2차원 배열

### ■ BOJ 2167 2차원 배열의 합

## Explanation - 배열

# 배열 - 배열이란

## 배열이란

- 같은 타입의 변수들로 이루어진 유한 집합
- `int arr[100];` 와 같이 선언해, `arr[0]`, `arr[1]`, ... `arr[99]` 와 같이 접근할 수 있다.
- 대괄호 안에 적은 숫자를 N이라 하면,  $0 \sim N-1$ 까지 총 N개의 공간을 생성한다.

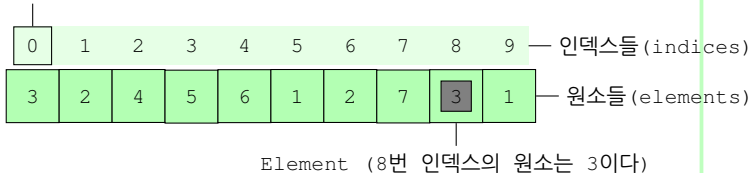
```
int arr[10]; // 크기 5의 배열 선언
arr[0] = 3; // 0번째 항목을 3로 선언
arr[1] = 2; // 1번째 항목을 2로 선언
arr[2] = 4; // 2째 항목을 4으로 선언
arr[10] = 3; // Error: arr[10]라 선언하면, arr[9]까지만 사용할 수 있다.
```

# 배열 - 값과 함께 선언하는 방법

값과 함께 선언도 가능하며, 이 배열을 시각화하면 다음과 같다.

```
int arr[10] = {3, 2, 4, 5, 6, 1, 2, 7, 3, 1};
```

Index (0번 인덱스)



# 배열이 왜 필요한지

배열을 쓰지 않으면, int a1, a2, a3 ... 변수를 한가득 선언 해야 하는데, 배열을 사용할 시 int arr[100] 과 같이 한 번에 할 수 있다. for문 등과 같이 사용해 반복 작업을 줄일 수 있다.

```
#include<iostream>

using namespace std;

int main(void) {
    int N;
    int arr[105]; // 105개의 int를 저장할 수 있다.
    cin>>N; // 몇 개를 입력 받을 것인가?
    for(int i=0;i<N;i++) {
        cin>>arr[i]; // 데이터를 N개 입력 받는다.
    }
    for(int i=0;i<N;i++) {
        cout<<arr[i]<<' '; // 입력 받은 데이터들을 출력한다.
    }
}
```

## Problem - 1차원 배열

## BOJ 10818 최소, 최대

최소, 최대

완료

☆

3 브론즈 III

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
1 초	256 MB	207450	89777	68685	42.865%

## 문제

N개의 정수가 주어진다. 이때, 최솟값과 최댓값을 구하는 프로그램을 작성하시오.

## 입력

첫째 줄에 정수의 개수 N ( $1 \leq N \leq 1,000,000$ )이 주어진다. 둘째 줄에는 N개의 정수를 공백으로 구분해서 주어진다. 모든 정수는 -1,000,000보다 크거나 같고, 1,000,000보다 작거나 같은 정수이다.

## 출력

첫째 줄에 주어진 정수 N개의 최솟값과 최댓값을 공백으로 구분해 출력한다.

예제 입력 1 복사

```
5
20 10 35 30 7
```

예제 출력 1 복사

```
7 35
```

Figure: BOJ 10818 최소, 최대 (<https://www.acmicpc.net/problem/10818>)



# BOJ 10818 최소, 최대 - 코드

Full ver: <https://www.acmicpc.net/source/share/25a55f30bc084cb5bfd97271b13ed39b>

```
int N;
cin>>N;
for(int i=0;i<N;i++){
    cin>>arr[i]; // N개의 정수를 입력 받는다.
}
int mini=1000000; // 문제에서 입력 제한이 100만보다 작다했으므로
int maxi=-1000000;
for(int i=0;i<N;i++) {
    // min(a,b) 함수는 a, b 중 작은 값을 리턴한다.
    mini = min(mini, arr[i]);
    // max(a,b) 함수는 a, b 중 큰 값을 리턴한다.
    maxi = max(maxi, arr[i]);
}
cout<<mini<<' '<<maxi;
```

# BOJ 11659 구간 합 구하기4

## 구간 합 구하기 4 중급



3 실버 III

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
1 초	256 MB	28544	13596	10597	46.533%

### 문제

수  $N$ 개가 주어졌을 때,  $i$ 번째 수부터  $j$ 번째 수까지 합을 구하는 프로그램을 작성하시오.

### 입력

첫째 줄에 수의 개수  $N$ 과 합을 구해야 하는 횟수  $M$ 이 주어진다. 둘째 줄에는  $N$ 개의 수가 주어진다. 수는 1,000보다 작거나 같은 자연수이다. 셋째 줄부터  $M$ 개의 줄에는 합을 구해야 하는 구간  $i$ 와  $j$ 가 주어진다.

### 출력

총  $M$ 개의 줄에 입력으로 주어진  $i$ 번째 수부터  $j$ 번째 수까지 합을 출력한다.

### 제한

- $1 \leq N \leq 100,000$
- $1 \leq M \leq 100,000$
- $1 \leq i \leq j \leq N$

#### 예제 입력 1 복사

```
5 3
5 4 3 2 1
1 3
2 4
5 5
```

#### 예제 출력 1 복사

```
12
9
1
```

Figure: BOJ 11659 구간 합 구하기4 (<https://www.acmicpc.net/problem/11659>)

## BOJ 11659 구간 합 구하기4 - 아이디어

- 1 만약  $i = 1, j = 10$ 만까지인 쿼리가 10만번 주어진다면,  
100억번은 연산이 너무 많아, 시간초과가 난다.
- 2 따라서, 구간 합을 더 빠르게 구해야 한다.
- 3 1부터  $i$ 번째까지의 합을 저장하는 `sum` 배열을 선언한다.
- 4  $i \leq j$ , 일 때,  $i$ 부터  $j$ 까지의 합은,  $\text{sum}[j] - \text{sum}[i-1]$ 과 동일하다.
- 5 예를 들어 `arr[3]`부터 `arr[6]`의 합은 노란색의 합과 같고,  
이는 빨간색 상자 2개의 차( $\text{sum}[6] - \text{sum}[2]$ )와 같다. ( $5+6+1+2 = 23-9 = 14$ )

0	1	2	3	4	5	6	7	8	9	배열 인덱스
3	2	4	5	6	1	2	7	3	1	arr 배열
3	5	9	14	20	21	23	30	33	34	sum 배열

## BOJ 11659 구간 합 구하기4 - 코드

Full ver: <https://www.acmicpc.net/source/share/2d2731e4f2fe46b3ad4c2293f191a806>

// 자세한 주석(설명)은 위에 링크에서 봐주세요!

```
cin>>N>>M;
for(int i=0;i<N;i++){
    cin>>arr[i];
    if(i==0) sum[0] = arr[i];
    else sum[i] = sum[i-1] + arr[i];
}
for(int i=0;i<M;i++) {
    int a, b;
    cin>>a>>b;
    a--;
    b--;
    cout<<sum[b]-sum[a-1]<<'\\n';
}
```

## Explanation - 2차원 배열

## 2차원 배열이란

### 2차원 배열이란

- 배열은 1차원이 아닌, 여러 차원으로 선언할 수 있다.
- `int arr[3][4];` 와 같이 선언해, `arr[0][0]`, `arr[0][1]`, ... `arr[2][3]` 와 같이 접근할 수 있다.
- 대괄호 안에 적은 숫자를 각각N, M이라 하면, 총  $N \times M$ 개의 공간을 생성한다.

```
int arr[3][4];  
arr[0][0] = 0;  
arr[0][1] = 1;  
arr[0][2] = 2;  
arr[2][3] = 11;
```

## 2차원 배열 - 값과 함께 선언하는 방법1

값과 함께 선언도 가능하다

```
int main(void) {  
    int arr[3][4] = {  
        {0, 1, 2, 3},  
        {4, 5, 6, 7},  
        {8, 9, 10, 11},  
    };  
    for(int i=0; i<3; i++) {  
        for(int j=0; j<4; j++) {  
            cout<<arr[i][j]<<' ';  
        }  
        cout<<'\n';  
    }  
}
```

## 2차원 배열 - 값과 함께 선언하는 방법2

값과 함께 선언 할 때는, 데이터를 1줄로 넣어도 된다. (순서대로 쌓인다. 아래 코드는 앞페이지의 코드와 동일한 결과를 출력한다.)

다만 arr[11] 이런식으로 접근하는 것은 안된다.

```
int main(void) {  
    int arr[3][4] = { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, };  
    for(int i=0;i<3;i++) {  
        for(int j=0;j<4;j++) {  
            cout<<arr[i][j]<<' '  
        }  
        cout<<'\n';  
    }  
}
```



## 2차원 배열 - 시각화

앞에서 적은 `arr[3][4]`의 구조를 시각화해보면 인덱스들은 다음과 같다.  
`arr[행][열]` 라고 생각하면 편하다.

<code>arr[0][0]</code>	<code>arr[0][1]</code>	<code>arr[0][2]</code>	<code>arr[0][3]</code>	0행
<code>arr[1][0]</code>	<code>arr[1][1]</code>	<code>arr[1][2]</code>	<code>arr[1][3]</code>	1행
<code>arr[2][0]</code>	<code>arr[2][1]</code>	<code>arr[2][2]</code>	<code>arr[2][3]</code>	2행
0열	1열	2열	3열	

## 2차원 배열 - 시각화2

2차원 배열의 앞에서부터 0부터 N까지 차례대로 넣으면, 이런 형태 ..?

0	1	2	3
4	5	6	7
8	9	10	11

## Problem - 2차원 배열

## BOJ 2167 2차원 배열의 합

## 2차원 배열의 합

실용

☆

1 브론즈 1

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
2 초	128 MB	20559	11618	9188	58.366%

## 문제

2차원 배열에 주어졌을 때  $(i, j)$  위치부터  $(x, y)$  위치까지에 저장되어 있는 수들의 합을 구하는 프로그램을 작성하시오. 배열의  $(i, j)$  위치는  $i$ 행  $j$ 열을 나타낸다.

## 입력

첫째 줄에 배열의 크기  $N, M$  ( $1 \leq N, M \leq 300$ )이 주어진다. 다음  $N$ 개의 줄에는  $M$ 개의 정수로 배열이 주어진다. 배열에 포함되어 있는 수는 절댓값이 10,000보다 작거나 같은 정수이다. 그 다음 줄에는 합을 구할 부분의 개수  $K$  ( $1 \leq K \leq 10,000$ )가 주어진다. 다음  $K$ 개의 줄에는 네 개의 정수로  $i, j, x, y$ 가 주어진다 ( $1 \leq i \leq N, 1 \leq j \leq M$ ).

## 출력

$K$ 개의 줄에 순서대로 배열의 합을 출력한다. 배열의 합은  $2^{31}-1$ 보다 작거나 같다.

## 예제 입력 1 복사

```
2 3
1 2 4
8 16 32
3
1 1 2 3
1 2 1 2
1 3 2 3
```

## 예제 출력 1 복사

```
63
2
36
```

Figure: BOJ 2167 2차원 배열의 합 (<https://www.acmicpc.net/problem/2167>)

## BOJ 2167 2차원 배열의 합 - 코드 (1/2)

Full ver: <https://www.acmicpc.net/source/share/9fc7a043d3be41fb84937dac570f57e4>

2차원 배열 입력 코드

```
int N, M;
cin>>N>>M;
for(int i=0;i<N;i++) {
    for(int j=0;j<M;j++) {
        cin>>arr[i][j];
    }
}
```

## BOJ 2167 2차원 배열의 합 - 코드 (2/2)

구간 합 쿼리 코드

```

int K;
cin>>K;
for(int i=0;i<K;i++){
    int a, b, c, d, sum=0;
    cin>>a>>b>>c>>d;
    a--; b--; c--; d--;
    // (a,c) 에서 (b,d) 의 합을 구해야 한다.
    for(int j=a;j<=c;j++){
        for(int k=b;k<=d;k++){
            sum += arr[j][k];
        }
    }
    cout<<sum<<'\n';
}

```