

2022-1 HI-ARC 초급스터디

2주차. 완전탐색

이지은 (leeju1013)

목차

1. 반복문
2. 배열
3. 벡터
- 4. 완전탐색**

부록1. 테스트 케이스
부록2. 배열 크기 제한

별 찍기 - 1

완료



브론즈 III

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
1 초	128 MB	186742	115160	97369	62.633%

문제

첫째 줄에는 별 1개, 둘째 줄에는 별 2개, N번째 줄에는 별 N개를 찍는 문제

입력

첫째 줄에 $N(1 \leq N \leq 100)$ 이 주어진다.

출력

첫째 줄부터 N번째 줄까지 차례대로 별을 출력한다.

예제 입력 1 복사

5

예제 출력 1 복사

```
*
**
***
****
*****
```

```

1 #include <iostream>
2 using namespace std;
3 int main(){
4     int N; cin>>N;
5
6     if(N==1) cout<<"*";
7     else if(N==2) cout<<"*\n**";
8     else if(N==3) cout<<"*\n**\n***";
9     else if(N==4) cout<<"*\n**\n***\n****";
10    else if(N==5) cout<<"*\n**\n***\n****\n*****";
11    else if(N==6) cout<<"*\n**\n***\n****\n*****\n*****";
12    .
13    .
14    .
15    else if(N==100) cout<<"*\n**\n***\n****\n*****\n*****\n*****\n*****\n*****\n*****";
16    return 0;
17 }

```

stdin

5

stdout

*

**

만약 N이 최대 1000 이라면?
만약 N이 최대 10000 이라면?
만약 N이 최대 10^8 이라면?

1. 반복문

- What?

- 프로그램 내에서 똑같은 명령을 일정 횟수만큼 반복 수행하도록 제어하는 명령문

- How?

- **for**(①초기화식; ②조건식; ④증감식){
 ③실행문;
}

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     for(int i=0; i<10; i++) cout<< i <<" ";
6     return 0;
7 }
```

stdout

0 1 2 3 4 5 6 7 8 9

예제 입력 1 복사

5

예제 출력 1 복사

*
**


```
1 #include <iostream>
2 using namespace std;
3 int main(){
4     int N; cin >> N;
5
6     for (int i = 0; i<N; i++){
7         for (int j=0; j <=i; j++)
8             cout << "*";
9         cout << '\n';
10    }
11
12    return 0;
13 }
```

1. 반복문

- while문

- **while**(①조건식){
 ②실행문;
}

- do-while문

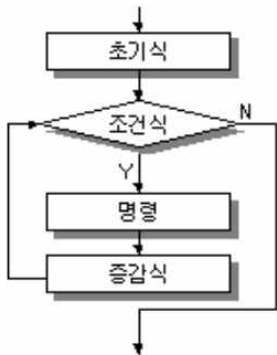
- **do**{
 ①실행문;
} **while**(②조건식)

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int i=0;
6     while(i<10){
7         cout<< i <<" ";
8         i++;
9     }
10    return 0;
11 }
```

⚙️ stdout

0 1 2 3 4 5 6 7 8 9

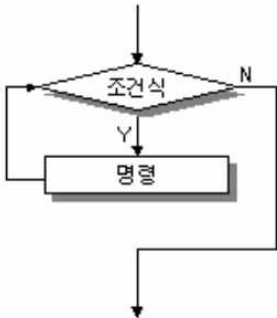
```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     for(int i=0; i<10; i++) cout<< i <<" ";
6     return 0;
7 }
```



for 문

```

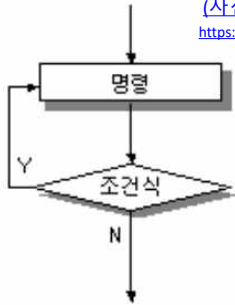
for(①초기화식; ②조건식; ④증감식){
    ③실행문;
}
  
```



while 문

```

while(①조건식){
    ②실행문;
}
  
```



do~while 문

```

do{
    ①실행문;
} while(②조건식)
  
```

(사진 출처)

<https://m.blog.naver.com>

1. 반복문 키워드

- break

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     for(int i=0; i<10; i++){
6         if(i==7) break;
7         cout<< i <<" ";
8     }
9     return 0;
10 }
```

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int i=0;
6     while(i<10){
7         if(i==7) break;
8         cout<< i <<" ";
9         i++;
10    }
11    return 0;
12 }
```

stdout

0 1 2 3 4 5 6

- continue

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     for(int i=0; i<10; i++){
6         if(i==7) continue;
7         cout<< i <<" ";
8     }
9     return 0;
10 }
```

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int i=0;
6     while(i<10){
7         if(i==7){
8             i++;
9             continue;
10        }
11        cout<< i <<" ";
12        i++;
13    }
14    return 0;
15 }
```

stdout

0 1 2 3 4 5 6 8 9

최소, 최대

성공



3 브론즈 III

시간 제한

메모리 제한

제출

정답

맞힌 사람

정답 비율

1 초

256 MB

195593

84441

64772

42.829%

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
```

```

1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int a, b, c, d, e;
6     cin>> a >> b >> c >> d >> e;
7
8     int max_num=0;
9     if(a > max_num) max_num = a;
10    if(b > max_num) max_num = b;
11    if(c > max_num) max_num = c;
12    if(d > max_num) max_num = d;
13    if(e > max_num) max_num = e;
14    cout<< max_num ;
15    return 0;
16 }

```

N=5라면? max_num을 왼쪽처럼 구할 수 있음.

근데 이 문제에선 N이 최대 100만.
100만개의 변수를 만들기.....?

심지어 N은 고정값X

2. 배열

- When?
 - 같은 타입의 데이터를 많이 다뤄야 하는 경우
- How?
 - 배열 선언
`int a, b, c, d, e; => int arr[5];`
 - 배열 접근
`arr[0], arr[1], arr[2], arr[3], arr[4]`

<https://nanite.tistory.com/56>

<https://nanite.tistory.com/56>

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int arr[5];
6     arr[0]=7, arr[3]=2;
7     cout<< arr[0]+arr[3];
8     return 0;
9 }
```

9

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     string str = "ABCDE";
6     cout<<str[0]<<" "<<str[3];
7     return 0;
8 }
```

A D

2. 배열

- 배열 초기화

`int arr[5];` //쓰레기값 들어있음

- 선언과 동시에 초기화

`int arr[5]={0,2,5,7,8};` //각각 초기값 지정함

`int arr[5]={0,2,5};` //생략한 것은 초기값 0

`int arr[5] ={};` //초기값 전부 0

- 전역 변수 배열 선언

`-fill(arr, arr + N , 0);`

시작 위치 끝 위치 초기화할 값

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int arr[5]={0,2,5};
6     for(int i=0; i<5; i++)
7         cout<<arr[i]<<" ";
8     return 0;
9 }
```

 stdout

0 2 5 0 0

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int a, b, c, d, e;
6     cin>> a >> b >> c >> d >> e;
7
8     int max_num=0;
9     if(a > max_num) max_num = a;
10    if(b > max_num) max_num = b;
11    if(c > max_num) max_num = c;
12    if(d > max_num) max_num = d;
13    if(e > max_num) max_num = e;
14    cout<< max_num ;
15    return 0;
16 }
```

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int arr[5];
6     for(int i=0; i<5; i++) cin>>arr[i];
7
8     int max_num=0;
9     for(int i=0; i<5; i++){
10         if(arr[i] > max_num) max_num = arr[i];
11     }
12     cout<< max_num ;
13     return 0;
14 }
```

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
```

```
1 #include <iostream>
2 using namespace std;
3
4 int arr[1000000];
5 int main(){
6     int N; cin >> N;
7
8     for (int i= 0; i<N; i++) cin >> arr[i];
9
10    int min_num = 1000000;
11    int max_num = -1000000;
12
13    for (int i=0; i<N; i++){
14        if(arr[i] < min_num) min_num = arr[i];
15        if(arr[i] > max_num) max_num = arr[i];
16    }
17
18    cout << min_num << " " << max_num;
19    return 0;
20 }
```

3. 벡터

- How?

- #include <vector> // vector 헤더파일을 추가

- 벡터 선언

vector<int> v(1000000); //초기값 항상 0

vector<int> v;

- 벡터 초기화

fill(arr.begin(), arr.end(), 초기화할 값);

```
1 #include <iostream>
2 #include <vector>
3 using namespace std;
4
5 int main() {
6     vector<int> v;
7
8     v.push_back(7);
9     cout<<v[0]<<" "<<v.size()<<'\n';
10
11     v.pop_back();
12     cout<<v.size();
13     return 0;
14 }
```

stdout
7 1
0

(vector 메소드)

<http://www.cplusplus.com>


```
1 #include <iostream>
2 #include <vector>
3 using namespace std;
4
5 vector<int> v(10000000);
6 int main(){
7     int N; cin >> N;
8
9     for (int i= 0; i<N; i++) cin >> v[i];
10
11     int min_num = 10000000;
12     int max_num = -10000000;
13
14     for (int i=0; i<N; i++){
15         if(v[i] < min_num) min_num = v[i];
16         if(v[i] > max_num) max_num = v[i];
17     }
18
19     cout << min_num << " " << max_num;
20     return 0;
21 }
```

```
1 #include <iostream>
2 #include <vector>
3 using namespace std;
4
5 vector<int> v;
6 int main(){
7     int N; cin >> N;
8
9     for (int i= 0; i<N; i++){
10         int x; cin>>x;
11         v.push_back(x);
12     }
13
14     int min_num = 10000000;
15     int max_num = -10000000;
16
17     for (int i=0; i<N; i++){
18         if(v[i] < min_num) min_num = v[i];
19         if(v[i] > max_num) max_num = v[i];
20     }
21
22     cout << min_num << " " << max_num;
23     return 0;
24 }
```

4. 완전탐색(Brute-force)

- What?
 - 가능한 모든 경우의 수를 탐색하는 알고리즘
 - 장점 : 예외 없이 100% 정답 출력
 - 단점 : 시간이 최대로 걸림

블랙잭

상문 다국어

☆ 한국어

보존즈 II

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
1 초	128 MB	93029	43839	34128	46.198%

문제

카지노에서 제일 인기 있는 게임 블랙잭의 규칙은 상당히 쉽다. 카드의 합이 21을 넘지 않는 한도 내에서, 카드의 합을 최대한 크게 만드는 게임이다. 블랙잭은 카지노마다 다양한 규정이 있다.

한국 최고의 블랙잭 고수 김정인은 새로운 블랙잭 규칙을 만들어 상근, 창영이와 게임하려고 한다.

김정인 버전의 블랙잭에서 각 카드에는 양의 정수가 쓰여 있다. 그 다음, 딜러는 N장의 카드를 모두 숫자가 보이도록 바닥에 놓는다. 그런 후에 딜러는 숫자 M을 크게 외친다.

이제 플레이어는 제한된 시간 안에 N장의 카드 중에서 3장의 카드를 골라야 한다. 블랙잭 변형 게임이기 때문에, 플레이어가 고른 카드의 합은 M을 넘지 않으면서 M과 최대한 가깝게 만들어야 한다.

N장의 카드에 써져 있는 숫자가 주어졌을 때, M을 넘지 않으면서 M에 최대한 가까운 카드 3장의 합을 구해 출력하시오.

입력

첫째 줄에 카드의 개수 N ($3 \leq N \leq 100$)과 M ($10 \leq M \leq 300,000$)이 주어진다. 둘째 줄에는 카드에 쓰여 있는 수가 주어지며, 이 값은 100,000을 넘지 않는 양의 정수이다.

합이 M을 넘지 않는 카드 3장을 찾을 수 있는 경우만 입력으로 주어진다.

출력

첫째 줄에 M을 넘지 않으면서 M에 최대한 가까운 카드 3장의 합을 출력한다.

예제 입력 1 복사

5 21
5 6 7 8 9

예제 출력 1 복사

21

- N, M 입력받음
- N개의 숫자들 입력받음
- 그 중 3개의 숫자를 골라야함.
- 숫자 합이 M을 넘지않으면서 M에 가장 가까운 걸로.

N=5, M=21

예제 입력 1 복사

5 21
5 6 7 8 9

예제 출력 1 복사

21

	0	1	2	3	4
arr	5	6	7	8	9

i j k

```
for(int i=0; i<N-2; i++){  
    for(int j=i+1; j<N-1; j++){  
        for(int k=j+1; k<N; k++){  
            if(ans <= arr[i]+arr[j]+arr[k] && arr[i]+arr[j]+arr[k] <=M)  
                ans = arr[i]+arr[j]+arr[k];  
        }  
    }  
}  
cout<< ans;
```

i j k

5 6 7

5 6 8

5 6 9

5 7 8

5 7 9

5 8 9

6 7 8

6 7 9

6 8 9

7 8 9

```

1 #include <iostream>
2
3 using namespace std;
4 typedef long long ll;
5
6 int main(){
7     ios_base::sync_with_stdio(false); cin.tie(NULL); cout.tie(NULL);
8
9     int N,M, ans=0; cin>>N>>M;
10    int arr[101];
11
12    for(int i=0; i<N; i++) cin>>arr[i];
13
14    for(int i=0; i<N-2; i++){
15        for(int j=i+1; j<N-1; j++){
16            for(int k=j+1; k<N; k++){
17                if(ans <= arr[i]+arr[j]+arr[k] && arr[i]+arr[j]+arr[k] <=M)
18                    ans = arr[i]+arr[j]+arr[k];
19            }
20        }
21    }
22    cout<< ans;
23    return 0;
24 }

```

```

1 #include <iostream>
2 #include <vector>
3
4 using namespace std;
5 typedef long long ll;
6
7 int main(){
8     ios_base::sync_with_stdio(false); cin.tie(NULL); cout.tie(NULL);
9
10    int N,M, ans=0; cin>>N>>M;
11    vector<int> v(101);
12
13    for(int i=0; i<N; i++) cin>>v[i];
14
15    for(int i=0; i<N-2; i++){
16        for(int j=i+1; j<N-1; j++){
17            for(int k=j+1; k<N; k++){
18                if(ans <= v[i]+v[j]+v[k] && v[i]+v[j]+v[k] <=M)
19                    ans = v[i]+v[j]+v[k];
20            }
21        }
22    }
23    cout<< ans;
24    return 0;
25 }

```

부록1. 테스트 케이스

입력

첫째 줄에 A와 B가 주어진다. ($0 < A, B < 10$)

출력

첫째 줄에 A+B를 출력한다.

예제 입력 1 복사

1 2

예제 출력 1 복사

3

5 1000번

A+B

```
1 #include <iostream>
2 using namespace std;
3 int main(){
4     int a, b;
5     cin >> a >> b;
6     cout << a+b;
7     return 0;
8 }
```

입력

첫째 줄에 테스트 케이스의 개수 T가 주어진다.

각 테스트 케이스는 한 줄로 이루어져 있으며, 각 줄에 A와 B가 주어진다. ($0 < A, B < 10$)

출력

각 테스트 케이스마다 A+B를 출력한다.

예제 입력 1 복사

```
5
1 1
2 3
3 4
9 8
5 2
```

예제 출력 1 복사

```
2
5
7
17
7
1 #include <iostream>
2 using namespace std;
3 int main(){
4
5     int t, a, b;
6     cin >> t;
7
8     for(int i=0; i<t; i++){
9         cin >> a >> b;
10        cout << a+b << '\n';
11    }
12    return 0;
13 }
```

TC수를 알려주는 경우

3 10950번

A+B - 3

```
1 #include <iostream>
2 using namespace std;
3 int main(){
4
5     int t, a, b;
6     cin >> t;
7
8     while(t--){
9         cin >> a >> b;
10        cout << a+b << '\n';
11    }
12    return 0;
13 }
```

입력

입력은 여러 개의 테스트 케이스로 이루어져 있다.

각 테스트 케이스는 한 줄로 이루어져 있으며, 각 줄에 A와 B가 주어진다. ($0 < A, B < 10$)

입력의 마지막에는 0 두 개가 들어온다.

출력

각 테스트 케이스마다 A+B를 출력한다.

예제 입력 1 복사

```
1 1
2 3
3 4
9 8
5 2
0 0
```

예제 출력 1 복사

```
2
5
7
17
7
1 #include <iostream>
2 using namespace std;
3 int main(){
4
5     int a,b;
6
7     for( ; ; ){
8         cin >> a >> b;
9         if (a==0 && b==0)
10             break;
11         cout << a+b << '\n';
12     }
13     return 0;
14 }
```

TC수를 알려주지 않고
TC의 끝을 알려주는 경우

3 10952번

A+B - 5

```
1 #include <iostream>
2 using namespace std;
3 int main(){
4
5     int a,b;
6
7     while(1){
8         cin >> a >> b;
9         if (!a && !b)
10             break;
11         cout << a+b << '\n';
12     }
13     return 0;
14 }
```


입력

입력은 여러 개의 테스트 케이스로 이루어져 있다.

각 테스트 케이스는 한 줄로 이루어져 있으며, 각 줄에 A와 B가 주어진다. ($0 < A, B < 10$)

출력

각 테스트 케이스마다 A+B를 출력한다.

예제 입력 1 복사

```
1 1
2 3
3 4
9 8
5 2
```

예제 출력 1 복사

```
2
5
7
17
7

1 #include <iostream>
2 using namespace std;
3 int main(){
4
5     int a,b;
6
7     while (1){
8         cin>> a >> b;
9         if(cin.eof()) break;
10        cout << a+b << '\n';
11    }
12    return 0;
13 }
```

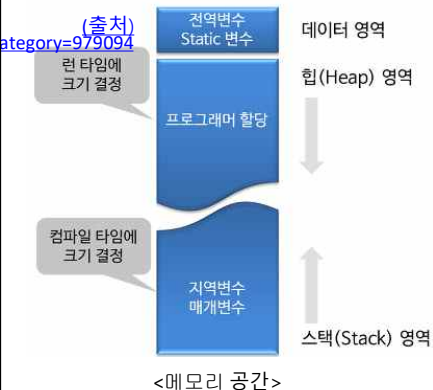
TC수를 알려주지 않고
TC의 끝을 알려주지 않는 경우

3 10951번

A+B - 4

```
1 #include <iostream>
2 using namespace std;
3 int main(){
4
5     int a,b;
6
7     while (cin >> a >> b){
8         cout << a+b << '\n';
9     }
10    return 0;
11 }
```

부록2. 배열 크기 제한



- 지역 배열 선언 -> stack에 할당
-> stack overflow

sol1) 전역 배열 선언

sol2) 벡터(vector) 사용

감사합니다

- 필수문제

3 2438번

별 찍기 - 1

3 10818번

최소, 최대

2 2798번

블랙잭

- 연습문제

5 1000번

A+B

3 10950번

A+B - 3

3 10952번

A+B - 5

3 10951번

A+B - 4

3 2739번

구구단

2 2231번

분해합

2 10808번

알파벳 개수

- 4월 5일(화요일) 저녁 8시

BAE<J>OON>
ONLINE JUDGE

문제 ▾

문제

> 전체 문제

> 문제 출처

> 단계별로 풀어보기

> 알고리즘 분류

> 추가된 문제

> 문제 순위

> 북마크

문제

> 내가 실패한 문제

> 내가 못 풀 문제

> 나만 풀 문제

> 풀 사람이 한 사람

> 아무도 못 풀 문제

> 최근 제출된 문제

> 최근 풀린 문제

> 안 풀 문제 랜덤

> 전체 문제 랜덤

단계	제목	총 문제
3	반복문	14
10	브루트 포스	5