

나도 디버깅 할 줄 알아요

강사 : 표지원

Debugging이란?

~~벌레 잡는 일~~

오류 수정. 컴퓨터 프로그램의 잘못을 찾아내고 고치는 작업이다!

보통 우리가 C언어를 배우기 시작하면, 이론적인 부분을 배우지만.....

소입설이나 창소프나

2학년 되면 배우는 자료구조론 등등

막상 나도 왜 생겼는지 잘 모르겠는 오류가 발생한다면!

왜 안되는 거지(?!?!?!?!?) 하면서 시간을 많이 날려버리게 되는 현실.. ㅠㅠㅠㅠ

(예를 들어, 맞왜틀..)

Debugging의 중요성

XXX : 아... 이거 잘못 적어서 3시간 날렸어 ㅠㅠㅠㅠㅠ

처음 코딩을 접하시는 분들은 아직 잘 모르실 수도 있는데,
실제로 여러분들이 짠 코드에서 에러가 난다면 대부분의 경우는

80%

15%

5%

오타!

내용 헛갈림.

기타 이유.

인데,

이런 단순한 오타가 몇 십줄 되는 코드에서는 정말 정~~말 잘 안보이기 때문에,

과제나 문제풀이(주로 대회 혹은 연습) 시간을 굉장히 잡아먹게 됩니다.

(여러분의 멘탈도 탈탈 털리게 되구요.)

그래서!!!!

제가 여러분에게 조금 도움이라도 되었으면 하는 마음에
제 경험 기준 쌓아온 디버깅 노하우를 알려드리고자 합니다.

(잘 체크해서 읽어주셨음 주셨음 좋겟슴당 히히)

Debugging하는 방법

우선, **디버깅**에서 여러분이 사용할 단축키들과 도구들은 다음과 같아요.

디버깅 단축키 4개 : **F11, F9, F5, Shift + F5** (기능은 뒤에서 설명)

조사식과 **자동** 탭 (VS2019버전 기준. 사실 조사식만 사용해도 좋습니다.)

!조사식이 안보인다면, 디버깅 중에 상단 바 검색란에 조사식 입력하시면 나와요!

Debugging하는 방법

기능은 다음과 같습니다.

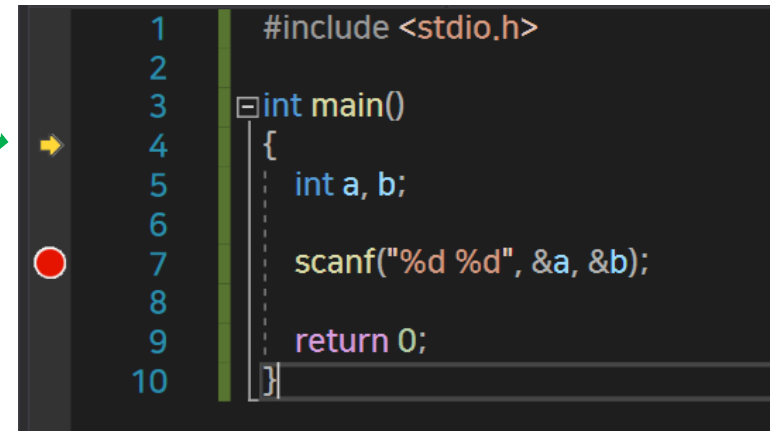
F11 : 한줄 한줄 넘어가면서 디버깅하기.

F9 : N번 줄에 중단점 놓기.

F5 : 현재 줄부터, 가장 가까운 중단점까지 실행.

(옆에 사진에선, 4~6번까지 모든 내용이 실행되고, 7번째 줄을 가리키고 있겠네요.)

Shift + F5 : 디버깅 종료.



```
1 #include <stdio.h>
2
3
4 int main()
5 {
6     int a, b;
7     scanf("%d %d", &a, &b);
8
9     return 0;
10 }
```

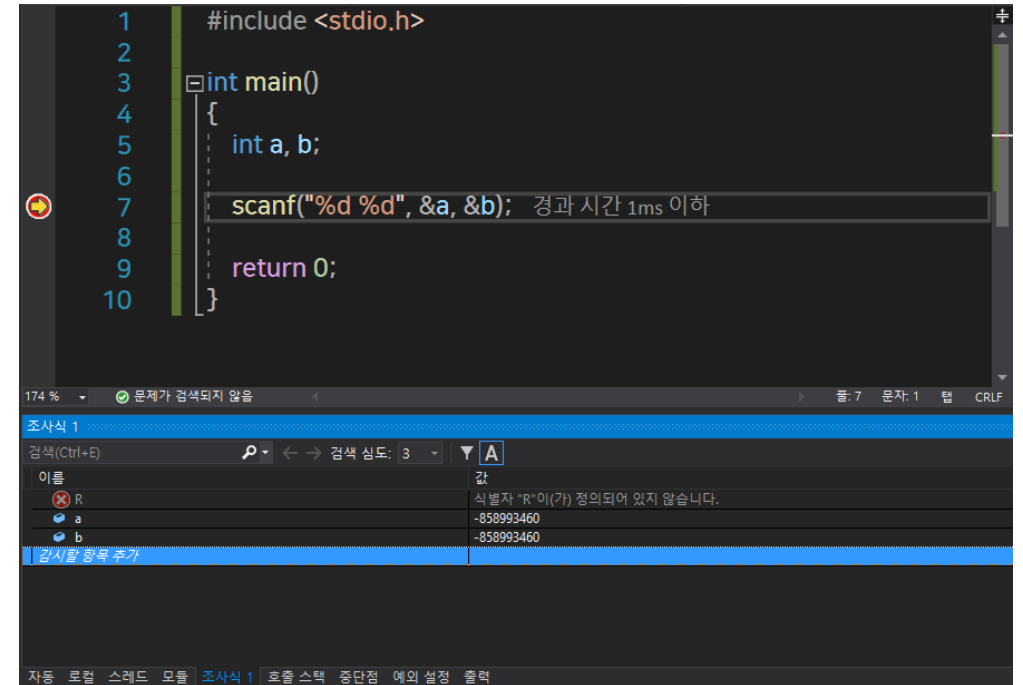
Debugging하는 방법

탭은 다음과 같습니다.

조사식 : 말 그대로, 조사를 할 수 있는 식

자동 : 자동적으로 조사할 내용을 추가해주는 조사식

(a와 b는 초기화를 하지 않아서, 일명 쓰레기 값이 들어가 있는 것을 확인 할 수 있네요.)



Debugging하는 방법

보통 디버깅을 한다, 즉 **한줄 한줄 보고싶다면**, **F11**을 여러 번 눌러서 보셔도 좋습니다.

이때, 본인이 선언한 변수 혹은 배열요소의 값이 **어떻게 변하는지 확인**해 보시려면
방금 전 보았던 **조사식**에서 보고싶은 변수나 배열요소의 이름을 적어서 확인해 보실 수도 있어요!

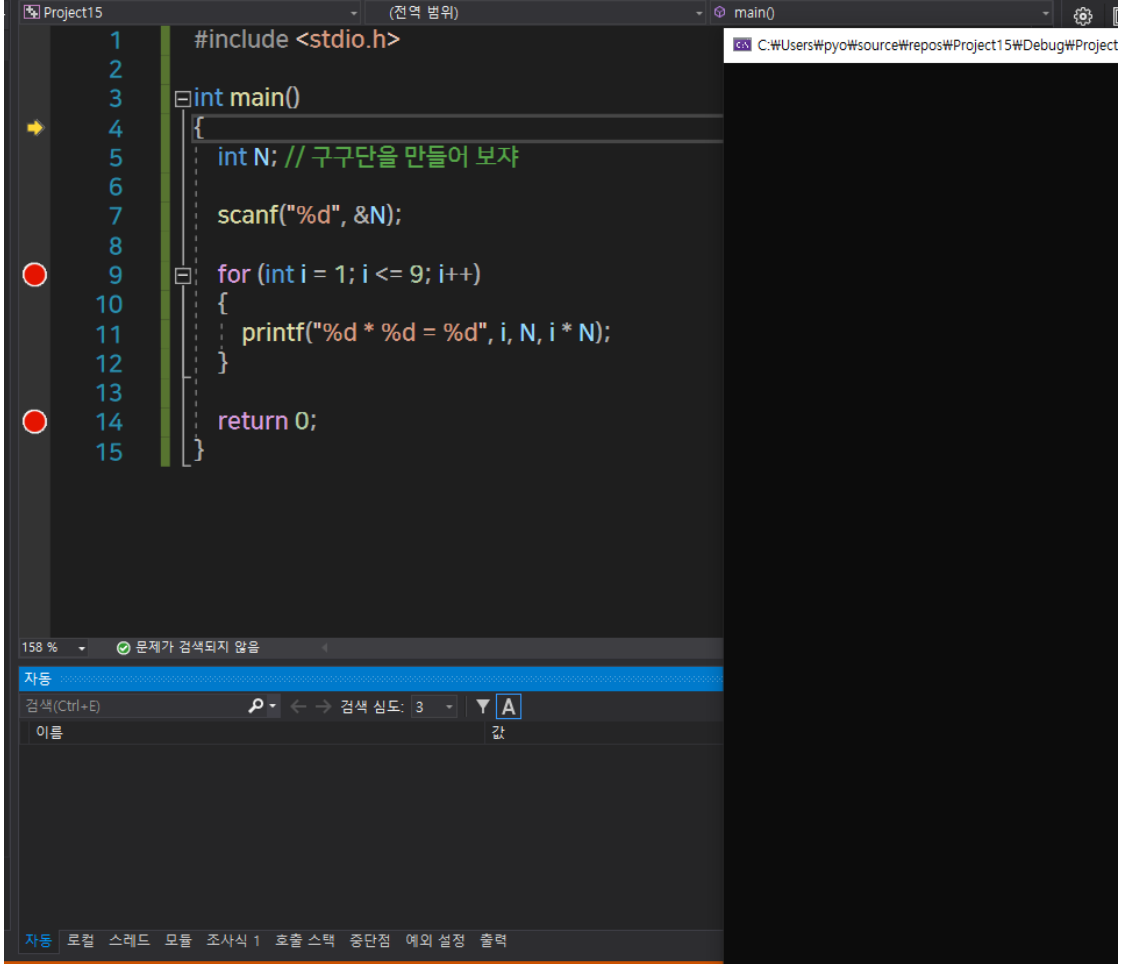
VS2019버전에서는, **자동** 탭이 알아서 지금 보고있는 범위 내에
살아있는 변수나 함수의 매개변수 값들을 보여줘서 상당히 편리하답니다.

(VS 다른버전에서도 자동 탭 있을 수 있음.)

Debugging하는 방법

예를 들어, 이렇게 하는 거예요!

우선 F11을 눌러서 디버깅을 시작했습니다.
우리는 구구단을 세로로 짚 출력해볼거예요.



```
1 #include <stdio.h>
2
3 int main()
4 {
5     int N; // 구구단을 만들어 보자
6
7     scanf("%d", &N);
8
9     for (int i = 1; i <= 9; i++)
10     {
11         printf("%d * %d = %d", i, N, i * N);
12     }
13
14     return 0;
15 }
```

Debugging하는 방법

예시로 7단을 볼 거예요.

F11으로 한 줄씩 넘어가서 입력을 마쳤습니다!

밑에 자동 탭에, N에 7값이 잘 들어갔음을 알 수 있네요.

또, for문에 있는 초기식이 실행되어서 i가 생겼네요.

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int N; // 구구단을 만들어 보자
6
7     scanf("%d", &N);
8
9     for (int i = 1; i <= 9; i++)
10     {
11         printf("%d * %d = %d", i, N, i * N);
12     }
13
14     return 0;
15 }
```

경과 시간 6,717ms 이하

158 % 문제가 검색되지 않음

자동

검색 (Ctrl+E) < > 검색 심도: 3

이름	값
&N	0x008ff890 (7)
i	-858993460

자동 로컬 스레드 모듈 조사식 1 호출 스택 중단점 예외 설정 출력

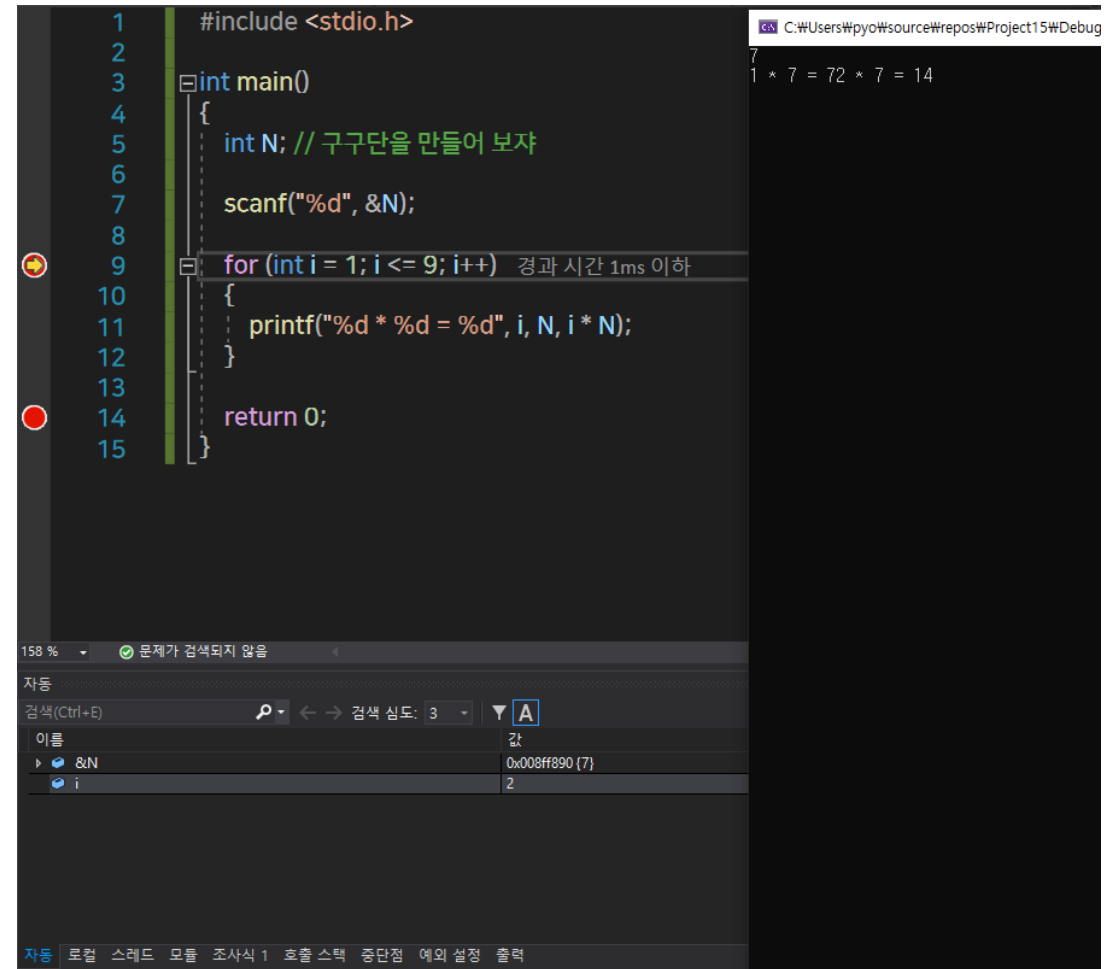
Debugging하는 방법

F11을 6번 정도 눌러서 for문 내 변화를 본 상황입니다
i가 증감식을 통해 2로 변했네요.

근데 문제가 발생했습니다!
개행을 했어야 하는데 안 했어요.

이렇게 문제가 바로 발견되었다면, Shift + F5로 종료 후 수정을 하셔도 되구요.

바로 코드에서 수정하고 왼쪽 화살표를 그 코드 실행 이전으로 돌릴 수도 있습니다.



Debugging하는 방법

급하게 개행 문자를 추가해서 아까 과정까지 바로 왔습니다.

또한, 입력이 잘 되는 것을 확인했으니, 9번 줄에 중단점을 두어서

F5키 한번에 for문으로 넘어올 수도 있었어요.

for문이 3번째로 실행되고 있는 과정입니다!

i가 3이 되었음을 자동 탭에서 볼 수 있고

출력도 잘 되고있는것을 확인할 수 있네요.

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int N; // 구구단을 만들어 보자
6
7     scanf("%d", &N);
8
9     for (int i = 1; i <= 9; i++)
10    {
11        printf("%d * %d = %d\n", i, N, i * N);
12    }
13
14    return 0;
15 }
```

Console Output:

```
7
1 * 7 = 7
2 * 7 = 14
```

Debugger Variables:

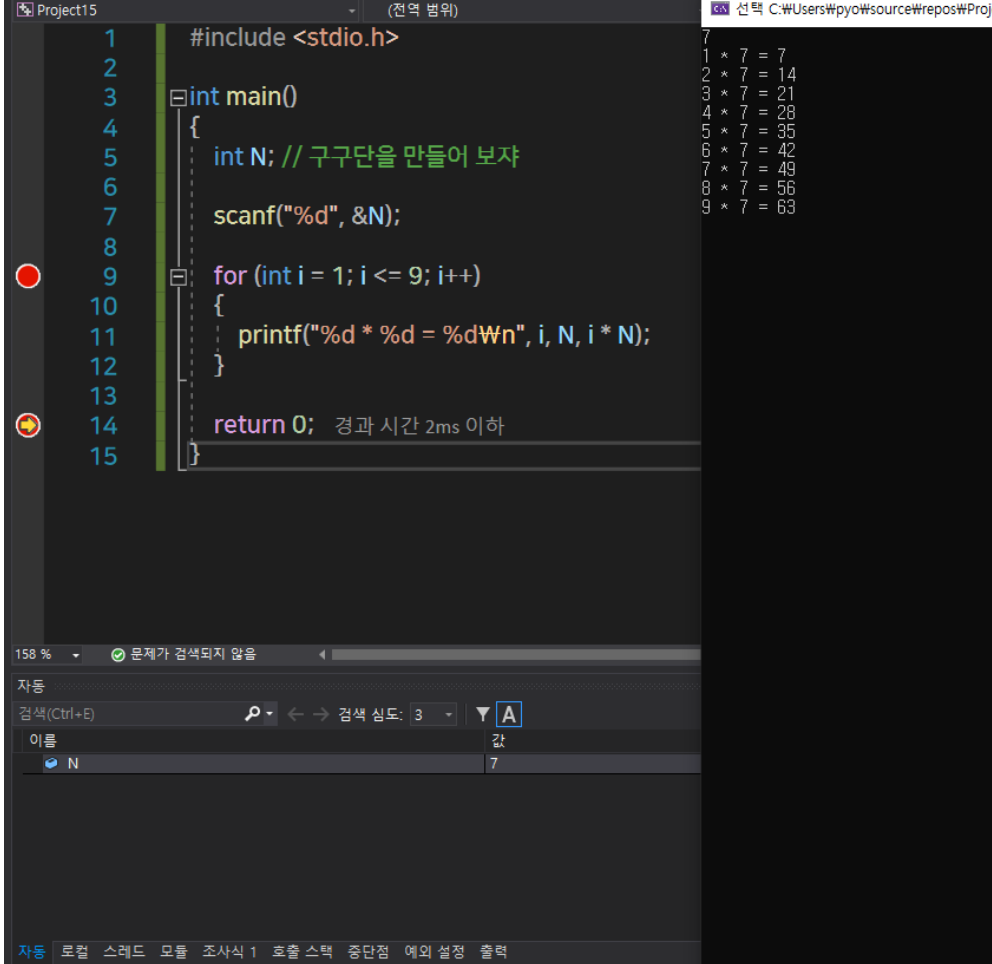
이름	값
N	7
i	3

Debugging하는 방법

마지막에 return 문장이 있는 14번째 줄에 중단점을 두어
남은 모든 for문을 실행시켰어요.
i가 사라진 것 역시 확인해 볼 수가 있네요.

여기까지 코드에 오류가 없었으니 디버깅을 종료해도 좋겠네요!

Shift + F5로 빠져나오도록 합시다.



```
1 #include <stdio.h>
2
3 int main()
4 {
5     int N; // 구구단을 만들어 보자
6
7     scanf("%d", &N);
8
9     for (int i = 1; i <= 9; i++)
10    {
11        printf("%d * %d = %d\n", i, N, i * N);
12    }
13
14    return 0; 경과 시간 2ms 이하
15 }
```

선택 C:\Users\pyo\source\repos\Proje

1 * 7 = 7
2 * 7 = 14
3 * 7 = 21
4 * 7 = 28
5 * 7 = 35
6 * 7 = 42
7 * 7 = 49
8 * 7 = 56
9 * 7 = 63

158 % 문제해 검색되지 않음

자동
검색(Ctrl+E) 검색 심도: 3

이름	값
N	7

자동 로컬 스레드 모듈 조사식 1 호출 스택 중단점 예외 설정 출력

Debugging하는 방법

이렇게 단순한 예시로 **디버깅**을 매우 간단하게 맛보았어요.

보통은 **디버깅**은 저렇게 코딩한 후 확인하는 식이 아니라
오류가 발생했을 때에 여러분이 활용하시면 되는데요,

우리는 알고리즘 문제풀이, 즉 **PS** 매일 해야 하잖아요~?

(뭐 과제일 수도 있구요.)

따라서 이제부터는 **PS를 할 때 여러분이 겪을 디버깅 상황과 함께 대처 방안**까지 알려드릴 게요!

(근데, 꼭 외울 필요는 없고 단순히 “가이드” 정도로만 여겨서 참고만 해주세요.)

(이후의 내용은 C언어반 분들이라면 3주차 내용을 모두 자습하시고 봐주세요.)

Debugging이 필요한 여러가지 상황

PS에서 디버깅을 해야 하는 경우는 대부분 다음과 같이 나뉩니다.

1. 예제는 **맞**는데 **왜 틀**리는 거임?

1-1. 아니 **반례**도 **맞**는데요?

2. 예제도 틀리는데 **왜 이게 오류**인지 모르겠어요... $\pi\pi$

3. “틀렸습니다”가 아닌, **런타임 에러** 혹은 **컴파일 에러**?

4. **그 외의 오류들**이라면?

Debugging 상황 - 1. 예제는 맞는데 왜 틀리는 거임? (맞왜틀..)

문제가 제시한 예제 입력에 따른 출력은 다 맞는데 틀린다면?

→ **반례 찾기** 그리고 **반례 디버깅하기**.

Debugging - 반례 찾기

(맞왜틀..)

이때, **반례**는 무조건

예제 입력 → **경계값** → **본인이 만든 임의의 값**

순서로 체크하는게 좋습니다.

!!입력 값에 음수, 양수 부분 나뉘는 것도 역시 체크해야 합니다!!

이거 중요해요 둘 다

!!나머지 연산도 제대로 되었는 가도 생각해주세요!!

과도한 반례는 코드에 이롭습니다.

Debugging - 반례 찾기



문제를 푸려고 하는건 아니에요!!
반례 구해보는 과정에만 집중해주세요.

예를 들면 이렇습니다. **10816번 숫자 카드 2** 문제에서,
입력과 출력에만 집중해주세요.

입력

첫째 줄에 상근이가 가지고 있는 숫자 카드의 개수 N ($1 \leq N \leq 500,000$)이 주어진다. 둘째 줄에는 숫자 카드에 적혀있는 정수가 주어진다. 숫자 카드에 적혀있는 수는 $-10,000,000$ 보다 크거나 같고, $10,000,000$ 보다 작거나 같다.

셋째 줄에는 M ($1 \leq M \leq 500,000$)이 주어진다. 넷째 줄에는 상근이가 몇 개 가지고 있는 숫자 카드인지 구해야 할 M 개의 정수가 주어지며, 이 수는 공백으로 구분되어져 있다. 이 수도 $-10,000,000$ 보다 크거나 같고, $10,000,000$ 보다 작거나 같다.

출력

첫째 줄에 입력으로 주어진 M 개의 수에 대해서, 각 수가 적힌 숫자 카드를 상근이가 몇 개 가지고 있는지를 공백으로 구분해 출력한다.

예제 입력 1 복사

```
10
6 3 2 10 10 10 -10 -10 7 3
8
10 9 -5 2 3 4 5 -10
```

예제 출력 1 복사

```
3 0 0 1 2 0 0 2
```

Debugging - 반례 찾기

입력

첫째 줄에 상근이가 가지고 있는 숫자 카드의 개수 $N(1 \leq N \leq 500,000)$ 이 주어진다. 둘째 줄에는 숫자 카드에 적혀있는 정수가 주어진다. 숫자 카드에 적혀있는 수는 $-10,000,000$ 보다 크거나 같고, $10,000,000$ 보다 작거나 같다.

셋째 줄에는 $M(1 \leq M \leq 500,000)$ 이 주어진다. 넷째 줄에는 상근이가 몇 개 가지고 있는 숫자 카드인지 구해야 할 M 개의 정수가 주어지며, 이 수는 공백으로 구분되어져 있다. 이 수도 $-10,000,000$ 보다 크거나 같고, $10,000,000$ 보다 작거나 같다.

출력

첫째 줄에 입력으로 주어진 M 개의 수에 대해서, 각 수가 적힌 숫자 카드를 상근이가 몇 개 가지고 있는지를 공백으로 구분해 출력한다.

예제 입력 1 복사

```
10
6 3 2 10 10 10 -10 -10 7 3
8
10 9 -5 2 3 4 5 -10
```

예제 출력 1 복사

```
3 0 0 1 2 0 0 2
```

입력의 형식은 **예제 입력**을 참고해주세요.

입력은 **N 개의 숫자카드**가 들어왔고,

그 다음에 **M 개의 질문**이 들어왔습니다.

Debugging - 반례 찾기

(맞왜틀..)

그러면 단순히 우리는 생각할 수 있는 반례는

N이 최대이고, 숫자는 랜덤인 경우

N은 크거나 작거나 상관없는데,
숫자가 -10,000,000이랑 10,000,000이 모두 들어가는 경우

N과 M이 최대인 경우입니다.

즉, 끝에 걸리는 **엣지 케이스**들을 생각해 낼 수 있고 적용해 볼 수도 있겠죠.

Debugging - 반례 찾기

(맞왜틀..)

문제마다 다르겠지만

문제 상황을 이해하고 반례가 될 수 있는 입력을 넣는 것도 하나의 방법입니다.

10816번 제출 맞은 사람 숏코딩 재채점/수정 채점 현황 내 소스 강의 질문 검색

숫자 카드 2 성공 분류

Silver IV

이분 탐색 해시를 사용한 집합과 맵

beenpow → 난이도 의견을 입력하지 않았습니다 해시를 사용한 집합과 맵

hibye1217 → Binary Search / Sorting 이분 탐색 정렬

tkdgs685 → 난이도 의견을 입력하지 않았습니다

시간 제한	메모리 제한	제출	정답	맞은 사람	정답 비율
1 초	256 MB	16886	5528	3919	34.894%

그 경우에 해당하는 반례들을

직접 생각해내도 좋지만

보통 이 문제에 접근한 다양한 사람들이
이미 질문으로 반례를 구해 둔 경우가 많아요.

문제 상단에 있는 질문검색 탭을 이용하면 좋습니다.

Debugging - 반례 찾기

(맞왜틀..)

이렇게요.

대부분의 질문에
댓글로 반례를 달아 주시는 경우가 많아요.

맞왜틀 이라면
여기에 있는 모든 반례를 찾는 것도 좋겠네요.

반례 부탁드립니다	4 10816번 질문	skesswswkk	0	0	2달 전	2달 전
67퍼센트에서 시간초과가 납니다ㅠㅠ 도와주세요 해결	4 10816번 질문	csp213	1	0	3달 전	3달 전
HashMap을 이용한 풀이 해결	4 10816번 자유	updown2011	2	0	3달 전	3달 전
자연스럽지 않은 문장이 있습니다. 해결	4 10816번 오타/오역/요청	cko301	1	0	3달 전	1달 전
lower_bound, upper_bound 질문 드립니다. 해결	4 10816번 질문	adh30	2	0	4달 전	4달 전
어느 부분으로 방향성을 잡아야할지 모르겠습니다. 해결	4 10816번 질문	kangsy763	1	0	4달 전	4달 전
c++ 런타임이 뜨네요 해결	4 10816번 질문	supernova1997	5	0	6달 전	6달 전
[java] 이분탐색 뭐가 문제지 잘 모르겠습니다..ㅠㅠ 해결	4 10816번 질문	jennylee4517	1	0	6달 전	6달 전
런타임 에러가 걸립니다. 'C' 해결	4 10816번 질문	heroswift15	1	0	7달 전	7달 전
문제점이 있을까요??	4 10816번 질문	neyo0311	0	0	8달 전	8달 전
오타 있습니다 해결	4 10816번 오타/오역/요청	jaehoo1	1	0	8달 전	1달 전
파이썬 시간초과하는데 원인을 모르겠습니다 ㅠ	4 10816번 질문	rlawhdals94	0	0	8달 전	8달 전
10816 시간초과 도와주세요 ㅜㅜ 해결	4 10816번 질문	quf9484	1	0	9달 전	9달 전
82퍼에서 계속 시간초과가 뜹니다. 도움주세요 해결	4 10816번 질문	joseph415	1	0	9달 전	9달 전
시간초과 질문입니다.. 부탁드립니다	4 10816번 질문	lwh1992	0	0	9달 전	9달 전
오타 있습니다 해결	4 10816번 오타/오역/요청	mwvy3055	1	0	10달 전	1달 전
시간 초과 질문이며 해결	4 10816번 질문	choee9712	1	0	1년 전	1년 전
[c++] map 사용했는데 시간초과 났습니다ㅠㅠ	4 10816번 질문	diana0501	2	1	1년 전	1달 전
틀왜맞	4 10816번 오타/오역/요청	inhoj123	2	0	1년 전	1년 전

Debugging - 반례도 맞는데?

(맞왜틀..)

그러면 대부분 이 경우예요!

개행을 했어야 하는데 하지 않았다! (아까 구구단처럼)

→ 'Wn' 추가해 주세요

입력되는 수가 변수의 저장 범위를 넘어서서 Overflow가 발생했다.

→ 주로 그게 int에서 발생하는데, long long으로 선언해주세요!

이 마저도 아니라면, 다른 문제가 있는지 보고
본인의 알고리즘이 맞는지 다시 점검해보세요!

Debugging 상황 - 2. 예제도 틀리는데 왜 오류인가요?

(왜틀..)

다양한 이유가 있겠지만, 이번 경우는 조금 당연합니다.

알고리즘을 잘못 짰다. (10%)

혹은, **참조를 잘못**했다. (90%)

주로 **참조를 잘못**하는 경우는

배열을 10칸 선언했는데 20번째 칸에 접근하려고 하는 경우 (런타임 에러 중 하나)

포인터가 가리키는 부분이 NULL인데, 그 값을 수정하려고 하는 경우 (nullptr)

Debugging 상황 - 2. 예제도 틀리는데 왜 오류인가요?

(왜틀..)

다양한 이유가 있겠지만, 이번 경우는 조금 당연합니다.

알고리즘을 잘못 짰다. (10%)

혹은, **참조를 잘못**했다. (90%)

주로 **참조를 잘못**하는 경우는

배열을 10칸 선언했는데 20번째 칸에 접근하려고 하는 경우 (런타임 에러 중 하나)

포인터가 가리키는 부분이 NULL인데, 그 값을 수정하려고 하는 경우 (nullptr)

이럴 때는 우선

F11을 통해 디버깅을 먼저 해 보시고

원치 잘 모르겠는 에러가 난다면,

구글링을 하시거나 **멘토에게 질문**해주시면 해결가능!

Debugging 상황 - 3. 런타임 에러??

(왜틀..)

런타임 에러란?

: 그냥 코드 문법에는 이상 없는데, **프로그램 실행 중에 일어나는 에러.**

전체 경우는 다음과 같아요.

배열에 할당된 크기를 넘어서 접근했을 때

전역 배열의 크기가 **문제에 주어진 메모리 제한을 초과할 때**

지역 배열의 크기가 스택 크기 제한을 넘어갈 때 (**너무 큰 배열일 선언된 경우**)

0으로 나눌 때 (**나눗셈 연산 확인 필요**)

라이브러리에서 예외를 발생시켰을 때 (**함수에서 예외발생**)

재귀 호출이 너무 깊어질 때 (**무한루프 혹은 너무 많은 함수호출**)

이미 **해제된 메모리를 또 참조할 때** (나중에 배우게 됩니다.)

**경우에 해당하는 부분을
디버깅을 통해 점검하면 해결!**

Debugging 상황 - 4. 이상한 에러문??

그 외에 해당하는 오류들

&

아무것도 아닌데 안되는 상황이 유지된다면?

구.글.링

Debugging은 언제나 옳아요.

여러분의 시간은 소중하니까, 1년 동안 PS하면서 만났던 오류의 대표유형들만 모아보았는데요

결론적으로, 디버깅도 경험이고 **그게 곧 실력**이에요.

대회에서나, 혹은 연습에서 문제 푸는 시간이 1이라면 디버깅이 4 이상을 차지할 정도로 시간이 소요되기 때문에, 평소에 틈틈이 디버깅 습관을 들여놓으시면 매우 좋습니다.

맞왜틀의 상황이 여러분에게 찾아온다면, 침착하게 오늘 한 내용들을 적용해보세요.

그 내용들만 **잘 써 먹을 수만 있다면 이미 당신은 디버깅 마스터!**

끝

(유용하게 쓰시길 바라겠습니다.)