

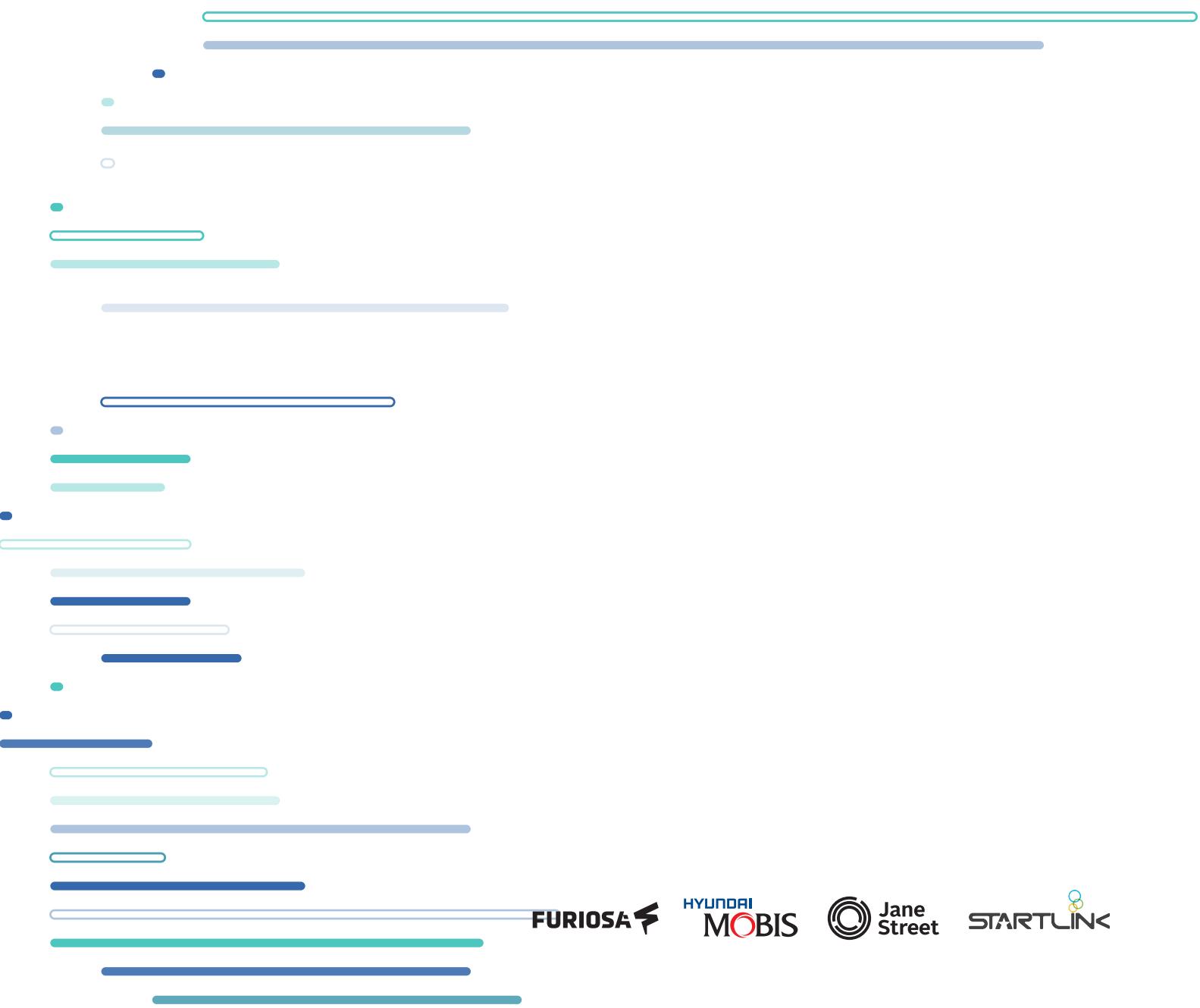
승실대학교 · SCCC

2024 **SCON**

승실대학교 프로그래밍 대회

문제지

Official Problemset



FURIOSA

HYUNDAI
MOBIS

Jane
Street

STARTLINK

언어 가이드

- 채점은 Intel Xeon E5-2666v3 프로세서를 사용하는 AWS EC2 c4.large 인스턴스에서 진행합니다.
- 채점 서버의 운영체제는 Ubuntu 16.04.7 LTS입니다.
- 아래 언어 중 원하는 언어를 선택해 사용할 수 있습니다.
 - C11: gcc 11.1.0
 - C++17: g++ 11.1.0
 - Java 15: OpenJDK version "16.0.1" 2021-04-20
 - Python 3: Python 3.11.5
 - PyPy3: Python 3.10.12, PyPy 7.3.12 with GCC 10.2.1 20210130
 - 컴파일과 실행 옵션은 <https://help.acmicpc.net/language/info>에서 확인할 수 있습니다.
- C11/C++17에서 `scanf_s`와 `Windows.h`등의 비표준 함수를 사용할 수 없습니다.
- Java를 사용하는 경우, `main` 메소드를 포함하는 클래스의 이름은 `Main`이어야 합니다.
- Python에서 `numpy`와 같은 외부 모듈을 사용할 수 없습니다.
- 채점 사이트에서 컴파일 에러를 받은 경우, ‘컴파일 에러’ 글씨를 누르면 오류가 발생한 위치를 볼 수 있습니다.
- 아래 코드는 표준 입력(standard input)을 통해 공백으로 구분된 두 정수를 입력으로 받아서 표준 출력(standard output)을 통해 합을 출력하는 코드입니다.

– C11

```

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

– C++17

```

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

– Java 15

```

1 import java.util.Scanner;
2
3 public class Main {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6         int a = sc.nextInt();
7         int b = sc.nextInt();
8         System.out.println(a + b);
9         sc.close();
10    }
11 }
```

- Python 3 / PyPy3

```

1 a, b = map(int, input().split())
2 print(a + b)
```

- 입출력 양이 많을 때는 위 코드를 사용한 입출력이 너무 오래 걸리기 때문에 다른 방식으로 입출력해야 합니다.
- C11/C++17에서 scanf와 printf를 사용하는 경우, 입출력 속도는 문제를 해결할 수 있을 정도로 충분히 빠릅니다.
- C++17에서 cin과 cout을 사용하는 경우, 입출력 전에 ios_base::sync_with_stdio(false);와 cin.tie(nullptr);를 사용하여야 합니다. 단, 이 이후에는 cin, cout 계열 함수와 scanf, printf 계열 함수를 섞어서 사용하면 안 됩니다. 또한, 개행문자로 std::endl 대신 "\n"을 사용해 주세요.
- Java 15에서는 BufferedReader와 BufferedWriter를 사용하여야 합니다.
- Python 3 및 PyPy3에서는 input() 대신 sys.stdin.readline().rstrip("\n")을 사용하여야 합니다. 코드의 가장 위 부분에 import sys 와
input = lambda: sys.stdin.readline().rstrip("\n") 을 사용하여야 합니다.
- 아래 코드는 표준 입력(stdandard input)을 통해 문제의 개수 T를 입력받은 다음 T줄에 걸쳐 공백으로 구분된 두 정수를 입력으로 받아 표준 출력(stdandard output)을 통해 두 정수의 합을 총 T줄에 걸쳐 출력하는 코드입니다.

- C++17

```

1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     ios_base::sync_with_stdio(false);
6     cin.tie(nullptr);
7     int T;
8     cin >> T;
9     for(int i=1; i<=T; i++){
10         int a, b;
```

```
11         cin >> a >> b;
12         cout << a + b << "\n"; // do not use endl
13     }
14     return 0;
15 }
```

– Java 15

```
1 import java.util.*;
2 import java.io.*;
3
4 public class Main{
5     public static void main(String[] args) throws IOException {
6         BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
7         BufferedWriter bw = new BufferedWriter(new OutputStreamWriter(System.out));
8
9         int T = Integer.parseInt(br.readLine());
10        for(int i=1; i<=T; i++){
11            String[] temp = br.readLine().split(" ");
12            int a = Integer.parseInt(temp[0]);
13            int b = Integer.parseInt(temp[1]);
14            bw.write(String.valueOf(a + b) + "\n");
15        }
16        br.close();
17        bw.close();
18    }
19 }
```

– Python 3 / PyPy3

```
1 import sys
2 input = lambda: sys.stdin.readline().rstrip("\n")
3
4 T = int(input())
5 for _ in range(T):
6     a, b = map(int, input().split())
7     print(a + b)
```

대회 중 유의 사항

- 2024 숭실대학교 프로그래밍 대회(2024 SCON) 유의 사항입니다.
- 대회 정보
 - 이 대회는 숭실대학교 IT대학이 주최, 컴퓨터학부 문제해결 소모임 SCCC가 주관합니다.
 - 이 대회는 현대모비스와 스타트링크의 후원을 받아 진행됩니다.
- 사진 촬영 안내
 - 대회 당일 현장 스태프가 대회 현장을 촬영하여 사진으로 기록 및 온라인 게시할 예정입니다.
 - 사진은 추후 숭실대학교, 숭실대학교 IT대학, IT대학 소속 학부, SCCC의 홍보와 SCCC가 주관하는 대회 홍보에 사용될 수 있습니다.
- 대회 진행 관련
 - 대회는 3시간 동안 10문제로 진행됩니다.
 - 6문제 이상 해결한 팀은 스코어보드 갱신이 중단됩니다.
 - 대회 종료 1시간 전부터 모든 팀의 스코어보드 갱신이 중단됩니다.
 - 대회 도중에는 대회 스태프와 팀원을 제외한 타인과 대화할 수 없습니다.
 - 대회 사이트 및 언어 공식 레퍼런스 사이트를 제외한 모든 인터넷의 사용은 금지됩니다.
 - * BOJ Help : <https://help.acmicpc.net/language/info>
 - * C/C++ : <https://en.cppreference.com/w/>
 - * Java : <https://docs.oracle.com/en/java/javase/15/docs/api/>
 - * Python : <https://docs.python.org/3/>
 - 화장실 이용 시 스태프와 동행해야 하며, 한 번에 한 명씩 이용 가능합니다.
 - 대회 도중 휴대전화 사용이 불가능합니다. 불가피한 경우 스태프의 감독하에 사용 가능합니다.
 - 문제와 관련된 질문은 대회 페이지의 ‘질문’ 탭을 이용해야 합니다. 현장 스태프는 문제에 대한 질문을 받지 않습니다.
 - 대회 공지는 대회 페이지의 ‘공지’ 탭을 이용해 전달합니다. 주기적으로 확인해 주시길 바랍니다.
- 문제 관련
 - 문제는 운영진들이 생각하는 난이도순으로 정렬되어 있지만, 모든 문제를 읽고 고민하는 것을 권장합니다.
 - 모든 문제는 C++, Java, PyPy3으로 해결할 수 있음이 보장됩니다. (Python3는 보장하지 않음)
 - 모든 문제의 메모리 제한은 1024MB로 동일합니다.
 - 제출한 프로그램은 문제에 명시된 제한 시간 내에 정답을 출력하고 정상적으로 종료되어야 합니다. 이는 return code가 0이어야 함을 의미하여, 이외의 exit code는 런타임 에러가 발생합니다.
 - 제출한 프로그램은 문제에 명시된 제한 메모리보다 많은 메모리를 사용할 수 없습니다.
 - 언어별 추가 시간과 추가 메모리가 주어지지 않습니다.
 - 제출한 프로그램은 표준 입력(standard input)을 통해 입력받아서 표준 출력(standard output)을 통해 정답을 출력해야 합니다.
 - 표준 입출력을 제외한 파일 입출력, 네트워킹, 멀티 스레딩 등의 시스템 콜은 사용할 수 없습니다.

- 팀의 등수는 다음과 같은 방법을 이용해 계산합니다.
 - 문제의 패널티 = (대회가 시작한 시점으로부터 처음으로 맞았습니다!!를 받기까지 걸린 분 단위 시간) + (제출 횟수 - 1) * 20분
 - 팀의 패널티 = 맞았습니다!!를 받은 모든 문제의 패널티의 합
 - 팀의 등수 = (더 많은 문제를 푼 팀의 수) + (푼 문제의 개수가 동일하면서 패널티가 더 작은 팀의 수) + 1
 - 컴파일 에러는 패널티에 포함되지 않습니다.
- 문제 관련 질문에 대한 답변은 다음 중 하나로 주어집니다.
 - 문제를 잘 읽어주시길 바랍니다 / **Read the problem statement**
질문에 대한 답변이 문제 지문에 있다는 의미입니다.
 - 답변할 수 없습니다 / **No comments**
질문이 잘못되었거나 답변으로 인해 대회 공정성이 해쳐지는 등 답변하기 힘들다는 의미입니다.
 - 전체 공지사항을 참고하시길 바랍니다 / **Please refer to the announcement**
답변 대신 전체 공지사항으로 공지한다는 의미입니다.
 - 네 / **Yes**
 - 아니오 / **No**
 - 기타 답변

간식 안내

- 제공되는 간식에 알레르기 유발 물질이 포함되어 있으니 주의해서 섭취하시길 바랍니다.
- 영양성분표 및 원재료명 관련 문의는 대회 페이지의 질문 기능을 통해 질문하면 답변드리겠습니다.
- 오예스
 - 우유, 밀, 계란, 대두 함유
 - 땅콩을 사용한 제품과 같은 시설에서 제조
- 쿠크다스 케이크
 - 밀, 우유, 대두, 계란, 돼지고기 함유
- 꿀이구마 호박 찹쌀 약과
 - 밀 함유
 - 대두, 난류, 메밀, 땅콩, 토마토, 우유, 복숭아를 사용한 제품과 같은 시설에서 제조
- 오레오
 - 밀, 대두, 우유 함유
 - 달걀, 메밀, 땅콩, 고등어, 게, 새우, 돼지고기, 복숭아, 토마토, 아황산류, 호두, 닭고기, 쇠고기, 오징어, 조개류(굴, 전복, 홍합 포함), 잣을 사용한 제품과 같은 시설에서 제조
- 마이구미
 - 우유, 돼지고기 함유
 - 복숭아 혼입 가능
- 신라명과 마드레느
 - 밀, 계란, 대두, 우유, 잣 함유

2024 Soongsil Programming Contest

Problem List

#	Problem Name	Time limit	Memory limit	Page
A	과민성 대장 증후군	2 seconds	1024MiB	8 – 8
B	팀명 정하기 2	1 second	1024MiB	9 – 10
C	온데간데없을뿐더러	1 second	1024MiB	11 – 11
D	미로 탈출	2 seconds	1024MiB	12 – 12
E	수식 고치기	2 seconds	1024MiB	13 – 14
F	피보나치 기념품	1 second	1024MiB	15 – 16
G	시간표 만들기	1 second	1024MiB	17 – 18
H	아이템 2	2 seconds	1024MiB	19 – 19
I	불꽃놀이의 아름다움	3 seconds	1024MiB	20 – 21
J	Traveling SCCC President 2	3 seconds	1024MiB	22 – 23

문제지에 있는 문제가 총 10문제가 맞는지 확인하시길 바랍니다.

문제는 출제진이 생각하는 난이도순으로 정렬되어 있지만, 모든 문제를 읽고 고민하는 것을 권장합니다.

모든 문제는 C++17, Java 15, PyPy3으로 풀 수 있음을 보장합니다. (단, Python 3는 보장하지 않음)

I번 문제와 J번 문제는 입출력 양이 매우 많습니다. 2~4페이지에 있는 언어 가이드를 참고하시길 바랍니다.

A. 과민성 대장 증후군

상원이는 과민성 대장 증후군을 앓고 있다. 과민성 대장 증후군의 원인은 스트레스!

상원이의 N 일 동안의 스트레스 변화량 A_1, \dots, A_N 이 주어진다. $A_i \geq 0$ 이면 i 번째 날에 A_i 만큼 스트레스가 쌓이고, $A_i < 0$ 이면 i 번째 날에 $-A_i$ 만큼 스트레스가 해소된다. 단, 변화를 관찰하기 시작한 시점의 스트레스 양은 0이며, 누적된 스트레스 양보다 해소하는 스트레스 양이 더 많을 경우 스트레스는 0이 될 때까지만 감소한다.

상원이는 스트레스가 M 이상 쌓인 날에 복통을 겪게 될 때, 상원이가 며칠 동안 복통에 시달리게 되는지 알아보자.

입력 형식

첫째 줄에 스트레스 변화를 관찰한 일수 N 과 복통을 겪게 되는 스트레스의 양 M 이 주어진다.

둘째 줄에 스트레스 변화량 A_1, A_2, \dots, A_N 이 공백으로 구분되어 주어진다.

출력 형식

상원이가 복통을 겪게 되는 일수를 출력한다.

제한

- $1 \leq N \leq 10^5$
- $1 \leq M \leq 10^9$
- $-10^4 \leq A_i \leq 10^4 (1 \leq i \leq N)$
- 입력으로 주어지는 수는 모두 정수이다.

예제

표준 입력(stdin)	표준 출력(stdout)
10 10 2 -1 4 7 4 -8 3 6 4 -7	6
5 1 1 -1 -1 -1 1	2

첫 번째 예시에서 10일 동안의 스트레스 양은 $[2, 1, 5, 12, 16, 8, 11, 17, 21, 14]$ 이며, 스트레스의 양이 10 이상인 날은 6일 있다.

두 번째 예시에서 5일 동안의 스트레스 양은 $[1, 0, 0, 0, 1]$ 이며, 스트레스의 양이 1 이상인 날은 2일 있다.

B. 팀명 정하기 2

숭실대학교 컴퓨터학부 문제해결 소모임 SCCC 부원들은 매년 SCON, ICPC와 같은 팀 대회에서 사용할 팀명을 정하기 위해 많은 고민을 한다. 특히 여러 개의 후보 중에서 실제로 사용할 팀명 하나를 정하는 것이 어렵기 때문에, 출업을 두 학기 남겨둔 정휘는 더 이상 부원들이 고통을 받지 않도록 가이드라인을 만들었다.

정휘의 가이드라인에 따르면 아래 조건 중 만족하는 조건의 개수가 많을수록 좋은 팀명이다.

- 알파벳 대문자보다는 소문자가 귀엽다. 따라서 대문자가 소문자보다 더 많이 등장하면 안 된다.
- 팀명이 너무 복잡하면 사회자가 제대로 읽지 못할 수 있다. 따라서 팀명은 10글자 이하로 지어야 한다.
- 숫자로만 구성된 팀명은 월드 파이널에 진출하지 못하는 징크스가 있다. 따라서 팀명에는 숫자가 아닌 글자가 하나 이상 포함되어 있어야 한다.

N 개의 팀명 후보가 주어진다. 가이드라인에서 제시한 세 가지 조건을 모두 만족하는 팀명을 출력하는 프로그램을 작성해 보자.

입력 형식

첫째 줄에 팀명 후보의 개수 N 이 주어진다.

둘째 줄부터 N 개의 줄에 걸쳐 팀명의 후보 S_1, S_2, \dots, S_N 이 한 줄에 하나씩 주어진다.

출력 형식

세 가지 조건을 모두 만족하는 팀명을 출력한다.

제한

- $1 \leq N \leq 100$
- $1 \leq |S_i| \leq 20 (1 \leq i \leq N)$. 즉, 팀명의 후보는 항상 길이가 20 이하인 문자열이다.
- S_i 는 알파벳 소문자, 알파벳 대문자, 숫자, 하이픈(‘-’)으로만 구성되어 있다.
- 세 가지 조건을 모두 만족하는 팀명은 정확히 하나만 주어진다.

예제

표준 입력(stdin)	표준 출력(stdout)
4 181920 LongestPathToWF NLP PS-akgwi	PS-akgwi

PS-akgwi는 알파벳 대문자보다 소문자가 더 많이 등장하고 8글자로 구성되어 있으며, 숫자가 아닌 문자를 포함하므로 세 가지 조건을 모두 만족한다.

NLP는 알파벳 대문자가 3번, 알파벳 소문자가 0번 등장하므로 첫 번째 조건을, LongestPathToWF는 15글자로 구성되어 있으므로 두 번째 조건을, 181920는 숫자로만 구성되어 있으므로 세 번째 조건을 만족하지 않는다.

참고

알파벳 대문자, 알파벳 소문자, 숫자, 하이픈의 아스키코드는 각각 65–90, 97–122, 48–57, 45이다.

181920은 2020 ICPC Asia Seoul Regional Contest 은상 수상팀이다.

LongestPathToWF는 2021 ICPC Asia Seoul Regional Contest 동상 수상팀이자 2022 ICPC World Finals 진출팀이다.

NLP는 2022 ICPC Asia Seoul Regional Contest 은상 수상팀이자 2023 ICPC World Finals 진출팀이다.

PS akgwi는 2023 ICPC Asia Seoul Regional Contest 은상, 2024 ICPC Asia Pacific Championship 은메달 수상팀이자 2024 ICPC World Finals 진출팀이다.

C. 온데간데없을뿐더러

‘온데간데없을뿐더러’는 어떻게 띄어 써야 할까? 형용사 ‘온데간데없다’와 어미 ‘-을뿐더러’가 결합하였기 때문에 띄어쓰기 없이 ‘온데간데없을뿐더러’라고 쓰는 것이 올바른 표현이다.

각각 N 개의 양의 정수로 구성된 두 배열 A, B 가 주어진다. A 에 있는 수를 모두 순서대로 공백 없이 이어서 썼을 때 얻게 되는 수를 X , B 에 있는 수를 같은 방식으로 이어서 썼을 때 얻게 되는 수를 Y 라고 했을 때, X 와 Y 중 더 작은 값을 구하는 프로그램을 작성하시오.

입력 형식

첫째 줄에 각 배열의 원소 개수 N 이 주어진다.

둘째 줄에 배열 A 의 원소 A_1, A_2, \dots, A_N 이 차례대로 공백으로 구분되어 주어진다.

셋째 줄에 배열 B 의 원소 B_1, B_2, \dots, B_N 이 차례대로 공백으로 구분되어 주어진다.

출력 형식

X 와 Y 중 더 작은 값을 출력한다. X 와 Y 가 같은 경우 그 값을 출력한다.

제한

- $1 \leq N \leq 9$
- $1 \leq A_i \leq 99 (1 \leq i \leq N)$
- $1 \leq B_j \leq 99 (1 \leq j \leq N)$
- 입력으로 주어지는 수는 모두 정수이다.

예제

표준 입력(stdin)	표준 출력(stdout)
5 5 4 3 2 1 1 2 34 5 6	54321
9 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99	999999999999999999

참고

X 와 Y 는 $2^{31} - 1$ 보다 클 수 있으므로 C/C++에서는 long long 타입, Java에서는 long 타입을 이용해 저장하는 것을 권장한다.

D. 미로 탈출

주원이는 방 탈출을 하던 중 미로에 도착했다. 미로는 일렬로 놓인 N 개의 방으로 이루어져 있으며, i 번 방과 $i+1$ 번 방은 서로 인접해 있다($1 \leq i < N$). 또한, 각 방에 스위치가 하나씩 있어, 주원이는 입구가 있는 S 번 방에서 출발하여 모든 스위치를 누르고, 출구가 있는 E 번 방으로 이동하여 탈출해야 한다.

주원이는 두 가지 이동 방법을 사용할 수 있다. 첫 번째는 0 만큼의 비용을 소모해 인접한 방으로 이동하는 것이고, 두 번째는 1 만큼의 비용을 소모해 1번 또는 N 번 방으로 순간 이동하는 것이다. 두 이동 방법 모두 매번 1 만큼의 시간이 소요된다. 스위치를 누르는 건 매번 0 만큼의 시간이 소요된다.

방 탈출에서 탈출하기 위해선 시간을 잘 관리해야 하므로 주원이는 최대한 빠르게 미로를 탈출하는 방법을 찾으려고 한다. 기록을 세우고 싶은 주원이를 위해서 가장 빠르게 미로를 탈출하는 방법 중 최소 비용으로 탈출하는 방법을 찾아주자.

입력 형식

첫째 줄에 테스트 케이스의 개수 T 가 주어진다.

다음 줄부터 각 테스트 케이스마다 N, S, E 이 한 줄에 공백으로 구분되어 주어진다.

출력 형식

각 테스트 케이스에 대해 가능한 최소 비용을 한 줄에 출력한다.

제한

- $1 \leq T \leq 10^4$
- $2 \leq N \leq 10^9$
- $1 \leq S, E \leq N$
- $S \neq E$
- 입력으로 주어지는 수는 모두 정수이다.

예제

표준 입력(stdin)	표준 출력(stdout)
3	0
5 5 1	1
3 1 2	2
7 3 5	

3번 테스트 케이스의 경우, $3 \rightarrow 4 \Rightarrow 1 \rightarrow 2 \Rightarrow 7 \rightarrow 6 \rightarrow 5$ 순서대로 이동해서 최단 시간일 때의 최소 비용으로 미로에서 탈출할 수 있다. 두 줄로 표시된 화살표가 순간이동을 쓴 시점이다.

E. 수식 고치기

T | F | T & F | T & F

위 식과 같이 값은 T와 F, 연산자는 |과 &로만 구성된 수식이 주어진다. | 연산은 양쪽에 있는 두 피연산자 중 하나 이상이 T이면 결괏값이 T이고, 그렇지 않으면 F이다. & 연산은 양쪽에 있는 두 피연산자가 모두 T이면 결괏값이 T이고, 그렇지 않으면 F이다. | 연산과 & 연산의 우선순위는 같고, 원쪽부터 오른쪽으로 차례로 계산한다고 가정한다.

여러분은 연산자와 값을 적절히 바꾸어 원하는 계산 결과가 나오도록 해야 한다. 이때, 연산자와 값을 바꾸는 횟수를 최소화해야 한다. 식과 원하는 계산 결과가 주어졌을 때, 원하는 계산 결과가 나오도록 연산자와 값을 고치는 최소 횟수를 구하여라.

입력 형식

첫째 줄에 식에 있는 값의 개수와 연산자의 개수를 더한 정수 N 이 주어진다.

둘째 줄에 N 개의 문자 A_1, A_2, \dots, A_N 이 공백으로 구분되어 주어진다. i 가 허수면 A_i 는 값이고, i 가 짹수면 A_i 는 연산자이다.

셋째 줄에 원하는 계산 결과를 나타내는 문자 C 가 주어진다.

출력 형식

원하는 계산 결과가 나오도록 값과 연산자를 고치는 최소 횟수를 출력하라.

제한

- $1 \leq N \leq 1999$, N 은 허수
- i 가 허수면 $A_i \in \{T, F\}$ ($1 \leq i \leq N$)
- i 가 짹수면 $A_i \in \{\&, |\}$ ($1 \leq i \leq N$)
- $C \in \{T, F\}$

예제

표준 입력(stdin)	표준 출력(stdout)
9 T & F F T & F T	1

A_9 를 T로 수정하면 된다.

표준 입력(stdin)	표준 출력(stdout)
3 F & F T	2
1 T T	0
1 T F	1

F. 피보나치 기념품



피보나치 수열은 다음과 같이 정의되는 수열이다.

- $F_1 = 1$
- $F_2 = 1$
- $F_n = F_{n-1} + F_{n-2}$ (단, $n \geq 3$)

정휘는 이집트 룩소르의 한 시장에서 피보나치 수가 적혀 있는 N 개의 기념품을 구매했다. x ($1 \leq x \leq N$) 번 기념품에는 F_x 가 적혀 있다.

정휘는 피보나치 수열을 좋아하는 세림이와 성주에게 기념품을 선물하려고 한다. 하지만 한 명에게 너무 많은 기념품을 주면 기념품을 적게 받은 사람이 슬퍼할 수 있기 때문에, 두 명이 받게 될 기념품에 적힌 피보나치 수의 합이 같아지도록 선물을 분배하려고 한다. 구매한 기념품의 개수 N 에 따라 N 개의 기념품을 전부 나눠주지 못할 수 있는데, 이때는 최대한 많은 개수의 기념품을 나눠주려고 한다.

나눠주는 기념품의 개수를 최대화하면서, 두 명이 받는 기념품에 적힌 수의 합이 같도록 기념품을 나눠주는 방법을 구해보자. 두 사람에게 1개 이상의 기념품을 나눠주는 방법은 항상 존재한다.

입력 형식

첫째 줄에 정휘가 구매한 기념품의 개수 N 이 주어진다.

출력 형식

첫째 줄에 세림이가 받을 기념품의 개수 X 를 출력한다.

둘째 줄에 세림이가 받을 기념품들의 번호 A_1, A_2, \dots, A_X 를 공백으로 구분해서 출력한다.

셋째 줄에 성주가 받을 기념품의 개수 Y 를 출력한다.

넷째 줄에 성주가 받을 기념품들의 번호 B_1, B_2, \dots, B_Y 를 공백으로 구분해서 출력한다.

가능한 분배 방법이 여러 가지면 그중 아무거나 하나만 출력하라.

제한

- $2 \leq N \leq 2000$
- $X, Y \geq 1$
- $1 \leq A_i \leq N (1 \leq i \leq X)$
- $1 \leq B_j \leq N (1 \leq j \leq Y)$
- $A_1, A_2, \dots, A_X, B_1, B_2, \dots, B_Y$ 는 서로 다른 수이다.
- 입력으로 주어지는 수는 모두 정수이다.

예제

표준 입력(stdin)	표준 출력(stdout)
2	1 1 1 2
4	2 1 3 1 4

G. 시간표 만들기

찬솔이는 이번 학기에 22학점을 들을 계획이다. 시간표를 만들기 위해 찬솔이는 다음과 같이 들을 예정인 과목들을 그룹으로 정리했다.

- 그룹 A : [전공종합설계1(가), 전공종합설계2(나), 컴퓨터공학특강1]
- 그룹 B : [네트워크프로그래밍(가), 네트워크프로그래밍(나)]
- 그룹 C : [멀티미디어응용]
- 그룹 D : [SW융합세미나1, 정보기술세미나1]
- 그룹 E : [파일처리(가), 파일처리(나)]
- 그룹 F : [디지털공학(나), 디지털공학(다)]
- 그룹 G : [문제해결]
- 그룹 H : [프로그래밍언어(가)]

그룹마다 그룹에 속한 강의 중 **최대 하나의 강의를 선택**해서 시간표를 구성한다. 각 강의에는 강의가 진행되는 요일, 강의 시작 시각, 강의 종료 시각이 있고, 선택한 강의끼리 진행 시간이 겹치면 안 된다. 요일이 다르거나 강의가 끝나는 동시에 다른 강의가 시작하는 것은 시간이 겹치는 것이 아니다.

꼭 모든 그룹에서 강의를 하나씩 선택해야 하는 것이 아니며, 하나의 그룹에 학점이 다른 강의가 있을 수도 있음에 유의하라. 또한, 모든 강의는 일주일에 한 번만 진행된다. 즉, 입력으로 주어지는 강의는 모두 서로 다른 강의이다.

찬솔이는 들을 예정인 과목들을 위와 같이 그룹으로 정리했을 때, 선택한 강의의 학점 합이 정확히 **22가 되도록** 시간표를 만들 수 있는 경우의 수가 궁금해졌다.

입력 형식

첫째 줄에 그룹의 개수 N 이 주어진다.

둘째 줄부터 N 개의 그룹과 각 그룹에 포함된 과목의 정보가 주어진다. 그룹에 포함된 과목의 개수 A_i 가 먼저 주어진다. 이어서 A_i 개의 줄에 걸쳐 각 과목의 학점 수 C , 요일 D , 강의 시작 시각 S , 강의 종료 시각 E 가 공백으로 구분되어 주어진다.

강의 시작 시각 S 와 종료 시각 E 는 HH시 MM분이 HH:MM 형식으로 주어진다.

출력 형식

주어진 입력으로 조건을 만족하며 만들 수 있는 22학점 시간표의 개수를 출력한다.

제한

- $1 \leq N \leq 15$
- $1 \leq A_i \leq 15$
- $\sum_{i=1}^N A_i \leq 15$
- $1 \leq C \leq 22$
- $1 \leq D \leq 7$
- $00 \leq \text{HH} \leq 23$
- $00 \leq \text{MM} \leq 59$
- $S < E$, 즉, 강의가 자정을 넘어서까지 진행되거나, 시작하자마자 종료하는 경우는 없다.
- 입력으로 주어지는 수는 모두 정수이다.

예제

표준 입력(stdin)	표준 출력(stdout)
8 3 3 1 13:30 14:45 3 1 15:00 16:15 2 4 14:00 14:50 2 3 2 10:30 11:45 3 2 12:00 13:15 1 3 2 15:00 16:15 2 1 2 16:30 17:20 1 4 16:30 17:20 2 3 4 12:00 13:15 3 4 15:00 16:15 2 3 1 09:00 10:15 3 1 10:30 11:45 1 3 1 09:00 10:15 1 3 1 13:30 14:45	8

H. 아이템 2

좌우로 무한히 긴 수직선 위에 N 개의 아이템이 떨어져 있다. i 번째 아이템의 위치는 i 이며, A_i 의 가치를 가지고 있다.

주원이에겐 길이가 K 인 집게가 있는데, 이 집게를 이용하면 주원이가 지정한 정수 좌표 x ($-10^9 \leq x \leq 10^9$)를 기준으로 x 부터 $x + K - 1$ 까지의 범위에 있는 아이템을 모두 줍는다. 주운 아이템은 그 자리에서 사라지며 주운 아이템은 다시 내려놓을 수 없다. 주원이는 집게를 원하는 만큼 사용해 주운 아이템의 가치의 합이 최대가 되도록 만들고 싶다.

주원이가 주운 아이템 가치의 합으로 가능한 값 중 최댓값을 찾아보자.

입력 형식

첫째 줄에 아이템의 개수 N 와 집게의 길이 K 가 주어진다.

둘째 줄에 아이템의 가치 A_1, A_2, \dots, A_N 이 차례대로 공백으로 구분되어 주어진다.

출력 형식

주운 아이템 가치의 합으로 가능한 값 중 최댓값을 출력한다.

제한

- $1 \leq K \leq N \leq 500\,000$
- $-500\,000 \leq A_i \leq 500\,000$ ($1 \leq i \leq N$)
- 입력으로 주어지는 수는 모두 정수이다.

예제

표준 입력(stdin)	표준 출력(stdout)
9 3 -2 3 -1 5 4 0 -10 5 -5	11
5 5 1 -10 -10 -10 1	2

두 번째 예제에서 $x = -3$ 과 $x = 5$ 에서 한 번씩 집게를 사용하면 주운 아이템의 가치의 합을 2로 만들 수 있다.

참고

정답이 32비트 정수 범위를 벗어날 수 있으므로 C/C++에서는 long long 타입, Java에서는 long 타입을 사용하는 것을 권장한다.

I. 불꽃놀이의 아름다움

봄 축제 때 정보과학관에서는 아무 행사도 진행되지 않는다는 것에 화가 난 찬솔이는 정보과학관 입구 앞에서 직접 불꽃놀이 행사를 진행하려고 한다.

정보과학관 입구 앞에 폭죽 또는 스위치를 설치할 수 있는 N 개의 공간과, 서로 다른 두 공간을 연결하는 도화선 $N - 1$ 개가 준비되어 있다. N 개의 공간은 도화선을 통해 모두 서로 연결되어 있다. 찬솔이는 N 개의 공간 중 한 곳에 스위치를 설치하고, 나머지 $N - 1$ 개의 공간에는 폭죽을 설치한다. 만약 i 번째 공간에 스위치가 설치되지 않았다면, i 번째 공간에는 W_i 개의 폭죽이 설치된다.

스위치를 설치한 공간의 번호를 a 라고 하자. 공간 x 와 y 사이의 거리 $D(x, y)$ 는 x 에서 y 까지 이동하는데 거쳐 가는 도화선의 최소 개수로 정의된다. 폭죽이 설치된 각 공간 b 에서 터지는 폭죽의 아름다움은 $W_b \times D(a, b)$ 이다.

불꽃놀이의 아름다움은 스위치가 설치된 공간 a 를 제외하고, 폭죽이 설치된 모든 공간 b 에서 터지는 폭죽의 아름다움의 합으로 정의된다. 스위치를 설치할 공간을 잘 정해서 얻을 수 있는 불꽃놀이의 아름다움의 최댓값을 구해보자.

입력 형식

첫째 줄에 공간의 수 N 이 주어진다.

둘째 줄부터 $N - 1$ 개의 줄에 걸쳐 도화선이 연결하는 두 공간의 번호 a, b 가 공백으로 구분되어 주어진다.

$N + 1$ 번째 줄에 N 개의 정수 W_1, W_2, \dots, W_N 이 공백으로 구분되어 주어진다.

출력 형식

가능한 불꽃놀이의 아름다움 중 최댓값을 출력한다.

제한

- $2 \leq N \leq 200\,000$
- $1 \leq a, b \leq N$
- $1 \leq W_i \leq 10^6$
- N 개의 공간은 도화선을 통해 서로 연결되어 있다.
- 입력으로 주어지는 수는 모두 정수이다.

예제

표준 입력(stdin)	표준 출력(stdout)
5	25
2 3	
3 1	
4 1	
1 5	
1 2 2 3 4	

2번 공간에 스위치를 설치하면 1, 3, 4, 5번 공간에서 **폭죽의 아름다움**이 각각 $1 \times 2 = 2$, $2 \times 1 = 2$, $3 \times 3 = 9$, $4 \times 3 = 12$ 가 되어서 **불꽃놀이의 아름다움**이 25로 최대가 된다.

참고

정답이 32비트 정수 범위를 벗어날 수 있으므로 C/C++에서는 long long 타입, Java에서는 long 타입을 사용하는 것을 권장한다.

입출력 양이 많으므로 문제지 2~4페이지의 언어 가이드에 있는 빠른 입출력을 사용하는 것을 권장한다.

J. Traveling SCCC President 2

숭실대학교 컴퓨터학부 문제해결 소모임 SCCC의 회장인 상원이는 정보과학관에서 협남공학관으로 이동해 대회 홍보 포스터를 붙이려고 한다.

숭실대학교의 캠퍼스는 1번부터 N 번까지 번호가 붙어 있는 N 개의 건물과, 서로 다른 두 건물을 연결하고 1번부터 M 번까지 번호가 붙어 있는 M 개의 도로로 구성되어 있다. i 번째 도로는 u_i 번 건물과 v_i 번 건물을 연결하고, 이 도로의 길이는 w_i 이다. 두 건물 쌍을 연결하는 도로는 최대 한 개만 존재하며, 모든 건물은 도로를 통해 서로 이동할 수 있다. 정보과학관의 건물 번호는 1번, 협남공학관의 건물 번호는 N 번이다.

일반적인 사람이라면 다른 건물로 이동할 때 사용한 도로 길이의 합 만큼의 시간이 소요되겠지만, 놀랍게도 상원이는 사용한 도로의 길이를 모두 bitwise OR 연산한 값 만큼의 시간이 소요된다. 상원이가 1번 건물에서 N 번 건물로 이동할 때 걸리는 최소 시간을 구해주자.

bitwise OR 연산이 무엇인지 잘 모르는 친구들은 문제 지문 맨 아래에 친절한 정휘가 준비해 놓은 정의를 읽어보도록 하자.

입력 형식

첫째 줄에 건물의 개수 N 과 도로의 개수 M 이 공백으로 구분되어 주어진다.

둘째 줄부터 M 개의 줄에 걸쳐, i 번 도로가 연결하는 두 건물의 번호 u_i, v_i 와 도로의 길이 w_i 가 한 줄에 하나씩 공백으로 구분되어 주어진다.

출력 형식

상원이가 1번 건물에서 N 번 건물로 이동하는 데 필요한 최소 시간을 출력한다.

제한

- $2 \leq N \leq 300\,000$
- $N - 1 \leq M \leq 300\,000$
- $1 \leq u_i, v_i \leq N, u_i \neq v_i (1 \leq i \leq M)$
- $0 \leq w_i < 2^{60} (1 \leq i \leq M)$
- $i \neq j$ 면 $(u_i, v_i) \neq (u_j, v_j), (u_i, v_i) \neq (v_j, u_j)$ 이다. 즉, 연결하는 두 건물의 쌍이 같은 도로가 여러 개 존재하지 않는다.
- 모든 건물은 도로를 통해 이어져 있다.
- 입력으로 주어지는 수는 모두 정수이다.

예제

표준 입력(stdin)	표준 출력(stdout)
4 4 1 2 1 2 3 2 3 4 3 1 4 4	3
2 1 1 2 1152921504606846975	1152921504606846975

참고

두 정수 A, B 를 bitwise OR한 값 $A \text{ OR } B$ 는 다음과 같이 정의된다.

- 이진법으로 생각했을 때, A 의 2^k 의 자릿수와 B 의 2^k 의 자릿수 중 하나 이상이 1이면 $A \text{ OR } B$ 의 2^k 의 자릿수가 1이고, 둘 다 0이면 $A \text{ OR } B$ 의 2^k 의 자릿수는 0이다.

예를 들어 $12 = 1100_{(2)}$, $10 = 1010_{(2)}$ 이므로 $12 \text{ OR } 10 = 1100_{(2)} \text{ OR } 1010_{(2)} = 1110_{(2)} = 14$ 로 계산된다.

k 개의 정수 A_1, A_2, \dots, A_k 를 bitwise OR한 결과는 $(\dots((A_1 \text{ OR } A_2) \text{ OR } A_3) \text{ OR } \dots) \text{ OR } A_k$ 로 정의된다. 이 연산의 결과는 A_1, A_2, \dots, A_k 의 순서를 바꾸더라도 변하지 않음을 증명할 수 있다.

도로의 길이와 정답이 32비트 정수 범위를 벗어날 수 있으므로 C/C++에서는 long long 타입, Java에서는 long 타입을 사용하는 것을 권장한다.

입출력 양이 많으므로 문제지 2-4페이지의 언어 가이드에 있는 빠른 입출력을 사용하는 것을 권장한다.