

제7회 천하제일 코딩대회 본선

Official Problem Set



Sponsored By:



언어 가이드

- 채점은 Intel Xeon E5-2666v3 프로세서를 사용하는 AWS EC2 c4.large 인스턴스에서 진행합니다.
- 채점 서버의 운영체제는 Ubuntu 16.04.7 LTS 입니다.
- 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 3.11.0
- PyPy3: Python 3.9.12, PyPy 7.3.9 with GCC 10.2.1 20210130 (Red Hat 10.2.1-11)
- 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 `scanf`와 `printf`를 사용하는 경우, 입출력 속도는 문제를 해결할 수 있을 정도로 충분히 빠릅니다.
- C++17에서 `cin`과 `cout`을 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 `BufferedReader`와 `BufferedWriter`를 사용하여야 합니다.
- Python 3 및 PyPy3 에서는 `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(standard input)을 통해 문제의 개수 T 를 입력받은 다음 T 줄에 걸쳐 공백으로 구분된 두 정수를 입력으로 받아 표준 출력(standard output)을 통해 두 정수의 합을 총 T 줄에 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)
```

제7회 천하제일 코딩대회 본선

Problem list

#	Problem Name	Time limit	Memory limit	Page
A	10!	2 seconds	1024MiB	6 – 6
B	고양이 카페	2 seconds	1024MiB	7 – 7
C	링크 컷 토마토	3.5 seconds	1024MiB	8 – 10
D	무한 수열	2 seconds	1024MiB	11 – 11
E	부정행위 멈춰!	2 seconds	1024MiB	12 – 12
F	사탕 나눠주기	2 seconds	1024MiB	13 – 13
G	생일 맞추기	2 seconds	1024MiB	14 – 14
H	수열의 가치	2 seconds	1024MiB	15 – 15
I	양동이 게임	2 seconds	1024MiB	16 – 16
J	크리스마스	2 seconds	1024MiB	17 – 17

문제지에 있는 문제가 총 10문제가 맞는지 확인하시기 바랍니다.

언어별 추가 시간과 추가 메모리가 주어지지 않습니다.

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

C, D, E, F, J번 문제는 입출력 양이 매우 많습니다. 2-4페이지에 있는 언어 가이드를 참고하시길 바랍니다.

A. 10!

10!초는 정확히 6주와 같다. 10!초는 $1 \times 2 \times 3 \times \dots \times 9 \times 10 = 3\,628\,800$ 초이고, 6주도 $7 \times 6 \times 24 \times 60 \times 60 = 3\,628\,800$ 초이기 때문이다.

$N! = 1 \times 2 \times 3 \times \dots \times N - 1 \times N$ 초는 몇 주인지 구하는 프로그램을 작성해 보자.

입력 형식

첫째 줄에 정수 N 이 주어진다. ($10 \leq N \leq 17$)

출력 형식

$N!$ 초가 몇 주와 동일한지 출력한다.

예제

표준 입력(stdin)	표준 출력(stdout)
10	6
17	588107520

참고

BOJ 채점 환경에서 `int` 자료형은 최대 $2^{31} - 1 = 2\,147\,483\,647$ 까지의 정수를 저장할 수 있고, C/C++의 `long long`과 Java의 `long`은 최대 $2^{63} - 1 = 9\,223\,372\,036\,854\,775\,807$ 까지의 정수를 저장할 수 있다.

B. 고양이 카페

찬우는 친구들과 고양이 카페에 가려 한다.

고양이 카페에는 N 마리의 고양이가 있다. i 번째 고양이의 무게는 w_i 이다. 찬우와 친구들은 모두 고양이를 사랑하기 때문에 무릎 위에 고양이를 정확히 2마리 데리고 있으면 행복해진다. 하지만 허약한 찬우와 친구들은 데리고 있는 두 고양이의 무게의 합이 K 를 넘는다면 버티지 못할 것이다.

각 고양이의 무게와 한 명이 버틸 수 있는 최대 무게 K 가 주어질 때 최대 몇 명이 행복해질 수 있는지 구해보자.

입력 형식

첫째 줄에 정수 N 과 K 가 공백으로 구분되어 주어진다. ($1 \leq N \leq 5\,000$; $1 \leq K \leq 10^9$)

둘째 줄에는 각 고양이의 무게를 의미하는 N 개의 정수 w_1, w_2, \dots, w_N 이 공백으로 구분되어 주어진다. ($1 \leq w_i \leq K$)

출력 형식

행복해질 수 있는 사람의 수의 최댓값을 출력한다.

예제

표준 입력(stdin)	표준 출력(stdout)
5 20 8 16 11 2 4	2
6 10 9 5 6 7 4 3	2
4 100 100 99 98 97	0

C. 링크 컷 토마토

토마토가 꼭지를 안테나처럼 사용해 연결을 형성하고 끊으며 네트워크를 이룬다는 사실은 잘 알려져 있다. 토마토 간의 연결은 날짜가 바뀌는 순간에만 형성되거나 끊어질 수 있으며, 임의의 두 토마토 사이의 연결 상태는 하루에 두 번 이상 바뀌지 않는다.

토마토 네트워크를 전공한 농부 존은 토마토의 연결 상태와 숙성도의 상관관계를 발견했다. 존의 발견에 따르면, 익은 토마토와 덜 익은 토마토가 하루 간 연결되어 있다면 덜 익은 토마토는 익은 토마토의 영향을 받아 익게 된다.

끈질긴 연구 끝에 존은 토마토 사이의 연결 상태를 관찰하는 기기를 개발했다. 이 기기를 사용하면 0일에 어떤 토마토가 익어 있고 어떤 토마토들끼리 연결되어 있는지 알 수 있으며, 임의의 두 토마토 사이의 연결이 언제 형성되고 끊어지는지도 추적할 수 있다.

찬우는 천하제일 코딩대회 출제비를 탈탈 털어 존의 기기와 1부터 N 까지의 번호가 하나씩 붙은 토마토 N 개를 샀지만, 어떤 토마토가 언제 익는지 알아내지 못했다. 찬우가 잘 익은 토마토를 먹을 수 있도록 존의 기기에서 얻은 정보로 각 토마토가 익는 날짜를 계산하는 프로그램을 작성해 주자.

입력 형식

첫째 줄에 토마토의 개수 N , 0일에 연결되어 있는 토마토 쌍의 수 M , 0일에 익은 토마토의 수 K , 연결 상태가 변하는 횟수 Q 가 공백으로 구분되어 주어진다. ($1 \leq K \leq N \leq 200\,000$; $0 \leq M, Q \leq 200\,000$)

둘째 줄부터 M 개의 줄에 걸쳐 서로 다른 토마토들의 초기 연결 상태가 중복 없이 주어진다. 각 줄에는 0일에 연결되어 있는 두 토마토의 번호 a, b 가 공백으로 구분되어 주어진다. ($1 \leq a < b \leq N$)

그 다음 줄에는 0일에 익어 있는 서로 다른 토마토의 번호 X_1, X_2, \dots, X_K 가 공백으로 구분되어 주어진다. ($1 \leq X_i \leq N$)

그 다음 줄부터 Q 개의 줄에 걸쳐 연결 상태의 변화가 일어나는 순서대로 주어진다. 각 줄에는 날짜 T 와 두 토마토의 번호 x, y 가 공백으로 구분되어 주어진다. $T - 1$ 일에 두 토마토가 연결되어 있었다면 T 일 0시부터는 연결이 끊어지고, 연결되어 있지 않았다면 T 일 0시부터 두 토마토는 새롭게 연결된다. 주어지는 (T, x, y) 쌍은 모두 다르다. ($1 \leq T \leq 200\,000$; $1 \leq x < y \leq N$)

입력으로 주어지는 모든 수는 정수이다.

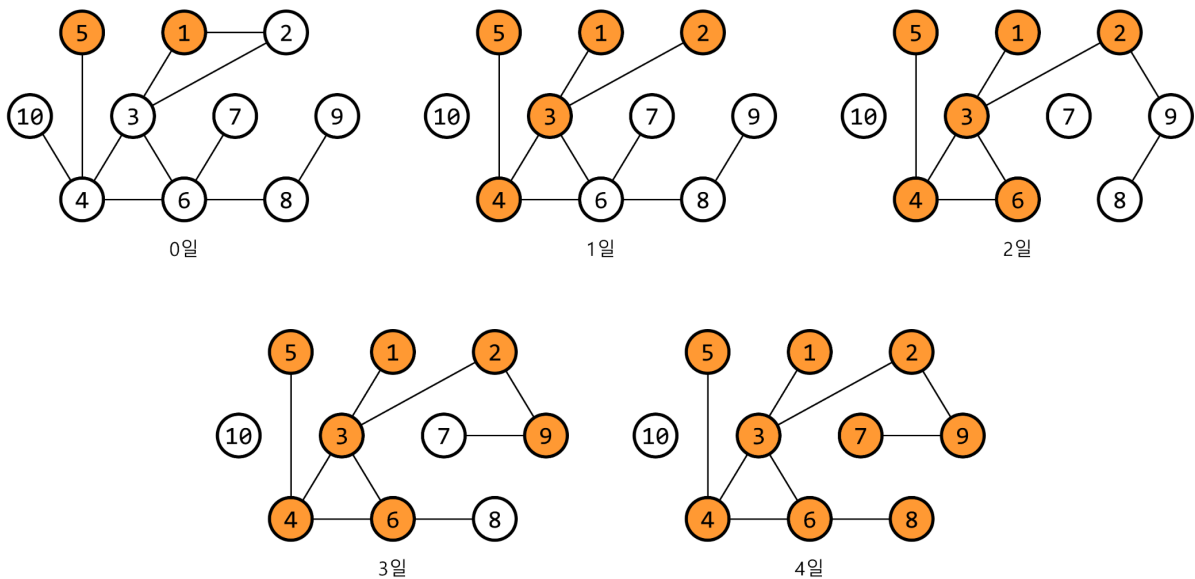
출력 형식

첫째 줄에 N 개의 정수를 공백으로 구분하여 출력한다. i 번 토마토가 영원히 익지 않는다면 i 번째 정수로 -1 을, 언젠가 익는다면 익는 날짜를 출력한다.

예제

표준 입력(stdin)	표준 출력(stdout)
10 11 2 8 1 2 6 7 2 3 3 4 4 6 1 3 4 5 3 6 8 9 6 8 4 10 1 5 1 1 2 1 4 10 2 6 8 2 2 9 2 6 7 3 6 8 3 7 9 3 8 9	0 1 1 1 0 2 4 4 3 -1

아래는 각각 0일, 1일, 2일, 3일, 4일의 토마토의 연결 상태를 나타낸 그림이다. 해당 날짜에 익어 있는 토마토만 색칠되어 있다. 3일 이후 연결 상태가 변하지 않기 때문에 10번 토마토는 영원히 익지 않는다.



표준 입력(stdin)	표준 출력(stdout)
6 5 1 0 1 2 3 4 1 4 5 6 2 3 1	0 1 2 1 -1 -1

연결 상태가 변하지 않는다.

표준 입력(stdin)	표준 출력(stdout)
5 0 2 4 1 2 1 1 5 1 3 4 3 2 3 3 4 5	0 0 4 4 2

0일에 연결되어 있는 두 토마토가 없다.

참고

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

D. 무한 수열

세훈이는 생일 기념으로 N 개의 정수로 이루어진 수열 A 를 선물로 받았다. 하지만 수열의 길이가 너무 짧다고 생각한 세훈이는 다음과 같은 방식으로 무한히 긴 수열을 만들기로 했다.

우선 수열 A 를 무한히 이어 붙인 뒤, 모든 양의 정수 i 에 대해 i 번째 수에서 i 를 뺀다.

수열의 길이가 길어졌음에 만족한 세훈이는 이렇게 만들어진 수열에서 연속된 몇 개의 수를 선택해서 더해보기로 했다. 물론 큰 것을 좋아하는 세훈이는 이러한 합을 최대로 만들고 싶어졌다.

예를 들면, 수열 A 가 $[9, 1, 7]$ 이었다면, 새로 만들어진 수열은 $[8, -1, 4, 5, -4, 1, \dots]$ 가 되고, 이 수열의 최대 연속합은 1번째 수부터 4번째 수까지의 합인 $8 - 1 + 4 + 5 = 16$ 이 된다.

입력 형식

첫째 줄에는 수열 A 의 길이 N 이 주어진다. ($1 \leq N \leq 500\,000$)

둘째 줄에는 수열 A 의 원소 A_1, A_2, \dots, A_N 이 공백으로 구분되어 주어진다. ($|A_i| < 2^{31}$)

출력 형식

문제의 정답을 출력한다.

예제

표준 입력(stdin)	표준 출력(stdout)
3 9 1 7	16
2 -12 -17	0

첫 번째 예시는 지문에 설명되어 있는 경우이다.

두 번째 예시에서 합을 구할 때 아무런 수를 선택하지 않을 수 있음에 주의하자.

참고

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

E. 부정행위 멈춰!

김채완 교수는 “프로그래밍과 봉사” 과목의 기말고사를 준비하고 있다. 고사실은 가로 M 칸, 세로 N 칸인 직사각형 형태이며, 모든 칸마다 정확히 한 명의 학생이 앉게 된다. 김채완 교수는 학생들의 부정행위를 막기 위해 모든 학생이 인접한 칸의 학생과 다른 시험지를 받을 수 있도록 여러 종류의 시험지를 준비하려 한다. 이때 인접한 학생은 가로, 세로, 또는 대각선으로 붙어 있는 칸에 있는 학생을 의미한다.

김채완 교수는 많은 종류의 시험지를 준비할 시간이 없었기 때문에 시험지의 종류를 최소화하고자 한다. 가능한 한 적은 종류의 시험지를 사용하여 조건을 만족하도록 시험지를 배부하는 프로그램을 작성하시오.

입력 형식

첫째 줄에 정수 N, M 이 공백으로 구분되어 주어진다. ($1 \leq N, M \leq 1000$)

출력 형식

첫째 줄에 사용한 시험지의 종류의 수 K 를 출력한다.

둘째 줄부터 N 개의 각 줄에 1 이상 K 이하의 정수 M 개를 공백으로 구분하여 출력한다. 각 정수는 해당 칸에 배부된 시험지의 번호를 의미한다. 즉, $y+1$ 번째 줄의 x 번째 정수는 위에서 y 번째, 왼쪽에서 x 번째 칸에 배부된 시험지의 번호이다.

예제

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

참고

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

F. 사탕 나눠주기

소수전공 수업을 마무리한 찬우는 축하의 의미로 학생들에게 사탕을 나누어 주려 한다. 구체적으로, 기준이 되는 음이 아닌 정수 X 를 정한 뒤 최종 점수가 X 점을 넘는 학생들에게 점수가 높은 만큼 많은 사탕을 줄 것이다. 즉, $X + 1$ 점을 받은 학생은 1개, $X + 2$ 점을 받은 학생은 2개, $T(T > X)$ 점을 받은 학생은 $T - X$ 개의 사탕을 받게 된다.

찬우는 학생들에게 최대한 많은 사탕을 나누어주고 싶기 때문에 기준 점수 X 를 가능한 한 낮게 정하려 한다. 하지만, 지금 가지고 있는 돈으로는 사탕을 K 개까지만 살 수 있기 때문에 사탕의 총 개수가 K 개를 넘으면 안 된다.

찬우의 수업은 총 N 명이 수강했고, i 번째 학생은 A_i 점을 받았다. 수강생의 수와 각 학생의 점수, 사탕의 최대 개수 K 가 주어질 때 찬우를 위해 가능한 X 의 최솟값을 구하는 프로그램을 작성해 주자.

입력 형식

첫째 줄에 정수 N, K 가 공백으로 구분되어 주어진다. ($1 \leq N \leq 5 \times 10^5$; $0 \leq K \leq 10^{12}$)

둘째 줄에 N 개의 정수 A_1, A_2, \dots, A_N 이 공백으로 구분되어 주어진다. ($0 \leq A_i \leq 10^{12}$)

출력 형식

첫째 줄에 가능한 기준 X 의 최솟값을 출력한다.

예제

표준 입력(stdin)	표준 출력(stdout)
4 80 80 100 50 40	50
4 61 80 100 50 40	60

참고

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

G. 생일 맞추기

많은 사람이 아이디를 지을 때 생일과 관련된 숫자를 넣지만, 찬우의 아이디에서 등장하는 숫자는 태어난 달이나 태어난 일에 등장하는 숫자들과 하나도 겹치지 않는다. 찬우의 아이디에 어떤 숫자가 포함되어 있는지 주어진다면, 찬우의 생일이 될 수 있는 날짜의 수를 구해보자.

찬우가 태어난 해는 윤년이다. 즉, 찬우의 생일은 2월 29일이 될 수도 있다. 또한 태어난 달이나 일이 한 자리라면 십의 자리는 빈 칸으로 생각한다. 따라서 찬우의 생일이 6월 7일이라면 생일에 0이 등장하지 않는 것이고, 6월 10일 또는 10월 7일이라면 생일에 0이 등장하는 것이다.

입력 형식

첫째 줄에 테스트 케이스의 수 T 가 주어진다. ($1 \leq T \leq 1234$)

둘째 줄부터 T 개의 줄에 걸쳐 한 줄에 하나씩 테스트 케이스가 주어진다.

각 줄에는 10개의 정수가 공백으로 구분되어 주어진다.

해당 테스트 케이스에서 $i - 1$ 이 찬우의 아이디에 등장한다면 i 번째 정수는 1, 등장하지 않는다면 0이다. ($1 \leq i \leq 10$)

출력 형식

각 테스트 케이스의 정답을 한 줄에 하나씩 출력한다.

예제

표준 입력(stdin)	표준 출력(stdout)
4	25
1 1 1 1 1 0 0 0 0 0	108
0 0 0 0 0 1 1 1 1 1	36
0 1 0 1 0 1 0 1 0 1	303
1 0 0 0 0 0 0 0 0 0	

첫 번째 테스트 케이스는 아이디에 0, 1, 2, 3, 4가 포함된 경우이다. 태어난 달은 5월부터 9월 사이, 태어난 일은 5일부터 9일 사이여야 하므로 생일이 될 수 있는 날짜는 총 25개이다.

두 번째 테스트 케이스는 5, 6, 7, 8, 9가, 세 번째 테스트 케이스는 1, 3, 5, 7, 9가, 네 번째 테스트 케이스는 0만 아이디에 포함된 경우이다.

표준 입력(stdin)	표준 출력(stdout)
1	4
1 1 1 1 1 1 0 0 1 1	

H. 수열의 가치

어떤 정수 수열 X 의 가치는 다음과 같이 정의된다:

- X 에서 감소하지 않는 부분 수열 P 와 증가하지 않는 부분 수열 Q 를 임의로 선택했을 때, (P 의 모든 원소의 합) + (Q 의 모든 원소의 합)의 최댓값

길이가 N 인 정수 수열 A 가 주어진다. A 를 원하는 대로 재배열하여 수열의 가치를 최대화하고 싶다. 재배열하여 만들 수 있는 수열의 가치의 최댓값과 이 때의 수열을 찾아보자.

감소/증가하지 않는 부분 수열이 무엇인지 잘 모르는 친구들은 친절한 정휘가 준비한 아래 정의를 읽어보도록 하자.

- **부분 수열**이란 주어진 수열에서 1개 이상의 원소를 골라 원래 순서대로 나열하여 얻은 수열을 말한다.
- **감소하지 않는 부분 수열**은 맨 처음 원소를 제외한 모든 원소가 바로 이전 원소보다 **크거나** 같은 부분 수열을 말한다.
- **증가하지 않는 부분 수열**은 맨 처음 원소를 제외한 모든 원소가 바로 이전 원소보다 **작거나** 같은 부분 수열을 말한다.

어떤 수가 두 부분 수열 P 와 Q 모두에 포함되도록 P 와 Q 를 선택할 수 있으며, 둘 모두에 포함된 수는 P 의 원소의 합을 구할 때와 Q 의 원소의 합을 구할 때 모두 더해진다.

입력 형식

첫째 줄에 N 이 주어진다. ($1 \leq N \leq 1000$)

둘째 줄에 정수 A_1, A_2, \dots, A_N 이 공백으로 구분되어 주어진다. ($1 \leq A_i \leq N$)

출력 형식

A 를 재배열하여 만들 수 있는 수열의 가치의 최댓값을 첫 번째 줄에 출력한다.

그 때의 수열 B_1, B_2, \dots, B_N 을 두 번째 줄에 공백으로 구분하여 출력한다. 가능한 수열이 여러 가지라면 그 중 아무거나 출력한다.

예제

표준 입력(stdin)	표준 출력(stdout)
4	12
2 3 1 2	1 3 2 2

I. 양동이 게임

찬우는 천하제일 코딩대회 참가자들과 간단한 게임을 하기로 했다. 게임은 '양동이 게임'으로, 맨 위의 양동이에 물을 100만큼 부어 흘려보냈을 때 최종적으로 가장 많은 물이 담긴 양동이를 선택하면 이기는 게임이다. 양동이를 고르기 전까지는 연결 상태를 알 수 없다. 하지만 찬우는 양동이들이 어떻게 연결되어 있는지 이미 알고 있는 상태였다! 찬우가 게임을 이기기 위해서 골라야 하는 양동이에는 얼마만큼의 물이 들어있는지 알아보자.

1번 양동이가 항상 맨 위에 있으며, 1의 양만큼 물이 채워져 있는 양동이에 $K > 0$ 개의 나가는 방향 호스가 연결되어 있으면 양동이가 비워질 때까지 호스당 $1/K$ 씩 흐른다. 또한 양동이의 번호가 더 작을수록 위에 있기 때문에, 물은 항상 번호가 작은 양동이에서 큰 양동으로 흐른다.

입력 형식

첫째 줄에 양동이의 개수 N 과 호스의 개수 M 이 공백으로 구분되어 주어진다. ($1 \leq N \leq 50; 0 \leq M \leq 100$)

둘째 줄부터 M 개의 줄에 걸쳐 호스의 정보를 나타내는 두 정수 v, w 가 공백으로 구분되어 주어진다. ($1 \leq v < w \leq N$) 이는 v 번 양동이에서 w 번 양동으로 물이 흐르는 호스가 있다는 뜻이다.

연결하는 양동이가 같은 호스가 여러 개 주어지지 않는다. 즉, 주어지는 모든 (v, w) 쌍은 서로 다르다.

출력 형식

1번 양동이에 물을 100만큼 채워서 흘려보냈을 때, 한 양동이에 가장 많이 들어있는 물의 양을 출력한다. 실제 정답과 출력값의 절대오차 또는 상대오차가 10^{-6} 이하이면 정답으로 처리한다.

예제

표준 입력(stdin)	표준 출력(stdout)
5 4 1 2 1 3 2 4 2 5	50.0
9 7 1 2 1 3 2 4 6 8 3 5 6 9 3 6	50.0

참고

절대오차는 실제 정답과 출력값의 차이, 상대오차(오차율)는 절대오차를 실제 정답으로 나눈 값을 의미한다. 즉, 실제 정답을 a , 출력값을 b 라고 하면, 절대오차는 $|a - b|$, 상대오차는 $\frac{|a-b|}{a}$ 로 계산한다.

J. 크리스마스

20XX년의 크리스마스가 다가옴에 따라 산타클로스는 선린 마을에 선물을 나눠주려고 한다. 선린 마을에는 1부터 N 까지의 번호가 매겨진 집이 있다. 집은 번호가 증가하는 순으로 원형을 이루고 있다. 즉, $1 \leq i < N$ 인 모든 i 에 대해 i 번 집과 $i+1$ 번 집은 이웃해 있고, N 번 집과 1번 집 또한 이웃해 있다. 이웃한 집 사이의 거리는 1이다. 산타클로스는 1번 집부터 시작해서 모든 집에 방문해 선물을 나눠준 뒤, 다시 1번 집으로 돌아올 것이다.

산타클로스는 마을 사람들이 잠에서 깨지 않도록 하기 위해 아래 규칙을 따라 이동하려 한다.

- 시계 방향 또는 반시계 방향으로 현재 집에서 거리가 2 이하인 집에만 갈 수 있다.
- 세 번 연속 같은 방향으로 갈 수 없다.
- 같은 집을 두 번 연속으로 방문할 수 없다.

같은 집을 두 번 연속으로 방문할 수 없지만, 다른 집을 거친 다음에 다시 방문하는 것은 가능하다.

1번 집에서 출발하여 가능한 한 적게 이동하며 모든 집을 방문하고 1번 집으로 돌아올 때 이동 횟수와 방문 순서를 출력하여야.

입력 형식

첫째 줄에 마을에 있는 집의 수 N 이 주어진다. ($5 \leq N \leq 1\,000\,000$)

출력 형식

첫 번째 줄에는 가장 적게 이동했을 때 이동 횟수 K 를 출력한다.

두 번째 줄에는 이동하며 방문한 집의 번호를 나타내는 $K+1$ 개의 정수를 공백으로 구분하여 출력한다. 집의 번호는 방문한 순서대로 출력한다.

반드시 1번 집에서 출발하여 모든 집을 방문하고 1번 집으로 돌아온 뒤 이동을 마쳐야 한다. 즉, 두 번째 줄에 출력하는 첫 정수와 마지막 정수는 항상 1이다.

가능한 답이 여러 가지라면 아무거나 출력한다.

예제

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

참고

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