

The 40th Annual ACM
International Collegiate Programming Contest
Asia Regional – Daejeon
Nationwide Internet Competition



Practice Problem A

Registration

Time Limit: 1 Second

Print out your ICPC team name and team number.

Input

No input is given for this problem.

Output

Your program is to write to standard output. Print exactly two lines. The first line should contain your full team name even if your team name contains one or more blanks (a character of ASCII 32) and the second line should contain your team number.

The following shows sample input and output, where the team number is 123 and the team name is Your_ICPC_Team_Name. Notice that no input is given.

Sample Input	Output for the Sample Input
	Your_ICPC_Team_Name 123

The 40th Annual ACM
International Collegiate Programming Contest
Asia Regional – Daejeon
Nationwide Internet Competition



Practice Problem A

등록

Time Limit: 1 Second

자신의 ICPC 팀 이름(team name)과 팀 번호를 그대로 출력하는 프로그램을 작성하시오.

Input

이 문제는 입력이 없다.

Output

표준출력(standard output)으로 출력해야 한다. 첫 줄에 자신의 팀 이름, 둘째 줄에 팀 번호를 출력한다. 출력할 팀 이름은 공백문자(ASCII 코드 32 번인 문자)를 포함하더라도, 공백문자를 포함하여 완전한 이름을 출력해야 한다.

다음은 팀 번호가 123 번, 팀 이름(team name)이 Your_ICPC_Team_Name 인 경우의 입출력 예제이다. 참고로 입력이 없는 것에 주의한다.

입력 예제 (Sample Input)

출력 예제(Output for the Sample Input)

	Your_ICPC_Team_Name 123
--	----------------------------

The 40th Annual ACM
International Collegiate Programming Contest
Asia Regional – Daejeon
Nationwide Internet Competition



Practice Problem B

Division

Time Limit: 0.1 Second

Given two integers a and b , write a program to print out a/b .

Input

Your program is to read from standard input. The input consists of T test cases, where the positive integer T is given in the first line of input, followed by the description of each test case. Each test case contains two positive integers a and b , where $1 \leq a \leq 10,000, 1 \leq b \leq 10,000$.

Output

Your program is to write to standard output. Print exactly one line for each test case. The line should contain a real number a/b . Your output should contain two digits after the decimal point, rounded off from the third digit.

The following shows sample input and output for three test cases.

Sample Input	Output for the Sample Input
3	1.33
20 15	5.00
10 2	0.10
1049 10000	

The 40th Annual ACM
International Collegiate Programming Contest
Asia Regional – Daejeon
Nationwide Internet Competition



Practice Problem B

나눗셈

Time Limit: 0.1 Second

주어진 두 정수 a 와 b 에 대하여 a/b 를 출력하는 프로그램을 작성하시오.

Input

표준입력에서 데이터를 읽는다. 입력은 T 개의 테스트 데이터로 구성되어 있다. 입력의 첫째 줄에 입력 데이터의 개수를 나타내는 정수 T 가 주어진다. 각 테스트 데이터는 두 정수 a 와 b 를 포함하는 한 줄로 구성되어 있는데, 여기서 $1 \leq a \leq 10,000, 1 \leq b \leq 10,000$ 이다.

Output

표준출력으로 데이터를 출력한다. 각 테스트 데이터에 대해, a/b 을 소수점 셋째 자리에서 반올림하여 소수점 둘째 자리까지 출력한다.

다음은 세 개의 테스트 데이터에 대한 입력과 출력의 예이다.

입력 예제 (Sample Input)	출력 예제(Output for the Sample Input)
3	1.33
20 15	5.00
10 2	0.10
1049 10000	

The 40th Annual ACM
International Collegiate Programming Contest
Asia Regional – Daejeon
Nationwide Internet Competition



Practice Problem C

SJF

Time Limit: 0.1 Second

The Shortest Job First scheduling (SJF) is well-known scheduling algorithm that executes the process with the smallest execution time first. It is assumed that there is no gap between processes and a process begins execution after the other process is complete. The SJF scheduling minimizes the sum of waiting times for all processes to start execution.

For example, let P_1 , P_2 , and P_3 be given and their execution times are 6, 7, and 3 milliseconds, respectively. Then, according to the SJF algorithm, the shortest process P_3 runs without waiting. After 3 milliseconds, P_1 begins execution. Finally, P_2 starts execution after waiting $(3 + 6) = 9$ milliseconds. Thus, the total waiting time is $(0 + 3 + 9) = 12$ milliseconds.

Given a set of processes and their execution times, write a program to compute total waiting time using the SJF scheduling algorithm.

Input

Your program is to read from standard input. The input consists of T test cases, where the positive integer T is given in the first line of input, followed by the description of each test case. Each test case starts with a line containing an integer, n ($1 \leq n \leq 1,000$), where n is the number of processes. In the following line, n positive integers are given each of which denotes the execution time of a process and is separated by a blank. Each execution time is between 1 and 1,000,000,000, inclusively.

Output

Your program is to write to standard output. Print exactly one line for each test case. The line should contain an integer, which represents the total waiting time using the SJF scheduling. Note that the answer can exceed the range of 4-byte integer type. You can use long long in C++ or long in JAVA.

The following shows sample input and output for three test cases.

Sample Input	Output for the Sample Input
3	12
3	0
6 7 3	3000000020
1	
100	
5	
1000000000 1 999999999 1000000000 6	

The 40th Annual ACM
International Collegiate Programming Contest
Asia Regional – Daejeon
Nationwide Internet Competition



Practice Problem C

최단 작업 우선

Time Limit: 0.1 Second

최단 작업 우선 (SJF) 스케줄링은 실행 시간이 가장 짧은 프로세스를 먼저 실행하는 스케줄링 알고리즘이다. 한 프로세스는 이전 프로세스가 종료된 후에 시작하며 프로세스 사이에 빈 시간은 없다고 가정한다. SJF 스케줄링은 모든 프로세스의 대기 시간의 합을 최소화한다.

예를 들어, 프로세스 P_1, P_2, P_3 의 실행 시간이 각각 6, 7, 3 밀리세컨드라고 하자. SJF 스케줄링에 따르면 실행 시간이 가장 짧은 프로세스 P_3 은 대기 시간 없이 먼저 실행된다. 다음으로 실행 시간이 짧은 P_1 은 P_3 가 실행되는 3 밀리세컨드를 대기한 후 실행된다. 마지막으로 P_2 는 $(3 + 6) = 9$ 밀리세컨드를 대기한 후 실행된다. 따라서 대기 시간의 합은 $(0 + 3 + 9) = 12$ 밀리세컨드이다.

주어진 프로세스들의 실행 시간에 대해 SJF 스케줄링 알고리즘을 사용하였을 때 대기 시간의 합을 출력하는 프로그램을 작성하시오.

Input

입력 데이터는 표준입력을 사용한다. 입력은 T 개의 테스트 데이터로 구성된다. 입력의 첫 번째 줄에는 입력 데이터의 수를 나타내는 정수 T 가 주어진다. 각 테스트 데이터의 첫 줄에는 프로세스의 수를 나타내는 정수 n ($1 \leq n \leq 1,000$)이 주어진다. 다음 줄에는 n 개의 프로세스의 실행시간이 공백을 사이에 두고 주어진다. 각 프로세스의 실행 시간은 1 이상 1,000,000,000 이하의 정수이다.

Output

표준출력으로 데이터를 출력한다. 각 테스트 데이터에 대해, SJF 스케줄링에 의한 총 대기 시간을 한 줄에 출력한다. 여기서, 결과가 32 비트 정수의 범위를 넘어갈 수 있음을 주의해야 한다. C++의 경우에는 long long을, JAVA의 경우에는 long을 통해 64 비트 정수를 사용할 수 있다.

다음은 세 개의 테스트 데이터에 대한 입력과 출력의 예이다.

입력 예제 (Sample Input)	출력 예제 (Output for the Sample Input)
3	12
3	0
6 7 3	3000000020
1	
100	
5	
1000000000 1 999999999 1000000000 6	

Practice Problem D Grid Graphs

Time Limit: 0.1 Second

The $m \times n$ rectangular grid is a graph whose vertices correspond to the points in the plane with x -coordinates being integers in the range $0, \dots, n - 1$ and y -coordinates being integers in the range $0, \dots, m - 1$, and two vertices are joined by an edge whenever the corresponding points are at unit distance apart. For example, a 4×6 rectangular grid is shown in Figure 1. The grid has n vertices appearing in each of m rows and m vertices in each of n columns. The vertex in row i and column j is denoted by (i, j) , where $0 \leq i \leq m - 1$ and $0 \leq j \leq n - 1$.

If we add an edge joining two vertices $(i, 0)$ and $(i, n - 1)$ of the $m \times n$ rectangular grid for every row $i \in \{0, \dots, m - 1\}$ and moreover, add an edge between two vertices $(0, j)$ and $(m - 1, j)$ for every column $j \in \{0, \dots, n - 1\}$, then each row forms a cycle of length n and each column forms a cycle of length m , as illustrated in Figure 2. The resulting graph is often called an $m \times n$ toroidal grid, because it can be drawn on a torus without edge crossings.

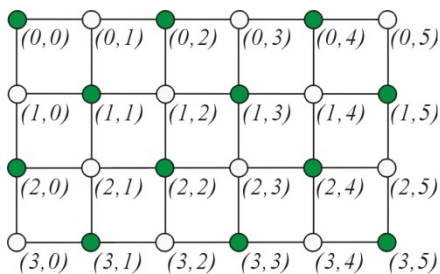


Figure 1. A 4×6 rectangular grid.

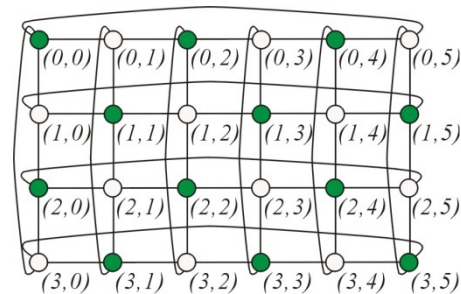


Figure 2. A 4×6 toroidal grid.

Given an $m \times n$ toroidal grid, you are to write a program to find a cycle that visits every vertex exactly once. Here, the required cycle may be represented as a sequence, $(v_1, v_2, \dots, v_{mn})$, of mn distinct vertices of the graph such that v_k and v_{k+1} are adjacent for all $k \in \{1, \dots, mn - 1\}$ and moreover, v_{mn} and v_1 are adjacent.

Input

Your program is to read from standard input. The input consists of T test cases. The number of test cases T is given in the first line of the input. Each test case consists of a single line containing two integers, m and n , where $3 \leq m, n \leq 100$, indicating that the input graph is an $m \times n$ toroidal grid.

Output

Your program is to write to standard output. For each test case, the first line of the results must contain an integer indicating whether there exists a feasible solution. If yes, the integer must be 1; otherwise -1. If the first line is 1 and only 1, then it must be followed by mn lines, describing a sequence of vertices of the required cycle. In case multiple solutions are possible, just output any one of them. No whitespace characters (blanks and/or tabs) are allowed inside a line.

The following shows sample input and output for two test cases.

Sample Input	Output for the Sample Input
2 3 4 3 3	1 (0,0) (0,1) (1,1) (1,2) (0,2) (0,3) (1,3) (2,3) (2,2) (2,1) (2,0) (1,0) 1 (2,2) (2,0) (1,0) (0,0) (0,1) (0,2) (1,2) (1,1) (2,1)

Practice Problem D

그리드 그래프

Time Limit: 0.1 Second

$m \times n$ 직사각 그리드(rectangular grid)는, x -좌표의 범위가 0부터 $n - 1$ 까지인 정수이고 y -좌표의 범위가 0부터 $m - 1$ 까지 정수인 평면상의 점들에 대응하는 정점들을 가지고, 두 정점에 대응하는 두 점 사이의 거리가 1 일 때에만 그 둘을 잇는 에지가 있는 그래프다. 예를 들어, 4×6 직사각 그리드가 그림 1에 있다. 이 그리드는 m 개 행 각각에 n 개의 정점이 있고, n 개 열 각각에 m 개의 정점이 있다. 행 i 와 열 j 에 있는 정점을 (i, j) 라고 나타낸다. 여기서 $0 \leq i \leq m - 1$ 이고 $0 \leq j \leq n - 1$ 이다.

$m \times n$ 직사각 그리드의 모든 행 $i \in \{0, \dots, m - 1\}$ 에 대하여 두 정점 $(i, 0)$ 과 $(i, n - 1)$ 을 잇는 에지를 추가하고, 또한 모든 열 $j \in \{0, \dots, n - 1\}$ 에 대하여 두 정점 $(0, j)$ 와 $(m - 1, j)$ 을 잇는 에지를 추가하면, 그림 2에 보인 것과 같이 각 행은 길이 n 인 사이클을 이루고 각 열은 길이 m 인 사이클을 이루게 된다. 이렇게 만들어진 그래프를 종종 $m \times n$ 토로이드 그리드(toroidal grid)라고 부르는데, 왜냐하면 이 그래프를 토러스(torus)에 에지가 교차하지 않도록 그릴 수 있기 때문이다.

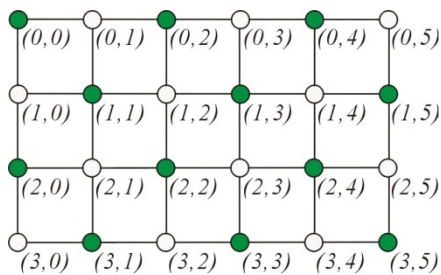


그림 1. 4×6 직사각 그리드.

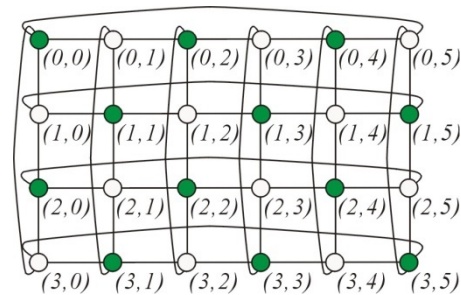


그림 2. 4×6 토로이드 그리드

주어진 $m \times n$ 토로이드 그리드에 대하여, 모든 정점을 정확히 한번씩 지나는 사이클을 찾는 프로그램을 작성하시오. 문제에서 요구하는 사이클은 그래프에 있는 서로 다른 mn 개 정점들의 열 $(v_1, v_2, \dots, v_{mn})$ 로 나타낼 수 있는데, 이때 모든 $k \in \{1, \dots, mn - 1\}$ 에 대하여 v_k 와 v_{k+1} 은 인접하며 또한 v_{mn} 과 v_1 도 인접하여야 한다.

입력(Input)

표준입력에서 데이터를 읽는다. 입력은 T 개의 테스트 데이터로 구성되어 있다. 입력의 첫째 줄에 입력 데이터의 개수를 나타내는 정수 T 가 주어진다. 각 테스트 데이터는 두 정수 m 과 n 을 포함하는 한 줄로 구성되어 있는데, 입력 그래프가 $m \times n$ 토로이드 그리드임을 가리킨다. 여기서 $3 \leq m, n \leq 100$ 이다.

출력(Output)

표준출력으로 데이터를 출력한다. 각 테스트 데이터에 대해, 출력의 첫째 줄에는 조건을 만족하는 해가 존재하는지 아닌지를 나타내는 정수를 출력해야 한다. 만약 해가 존재하면, 그 정수는 1 이어야 한다; 그렇지 않으면 -1 이다. 첫째 줄이 1 일 경우에만 추가로 문제에서 요구한 사이클의 정점 열(sequence)을 mn 개의 줄에 출력한다. 복수의 해가 가능하면, 그 중 임의의 하나를 출력하면 된다. 어떤 줄에도 공백 문자(빈칸이나 탭)은 허용되지 않는다.

다음은 두 개의 테스트 데이터에 대한 입력과 출력의 예이다.

입력 예제(Sample Input)

출력 예제(Output for the Sample Input)

2	1
3 4	(0,0)
3 3	(0,1)
	(1,1)
	(1,2)
	(0,2)
	(0,3)
	(1,3)
	(2,3)
	(2,2)
	(2,1)
	(2,0)
	(1,0)
	1
	(2,2)
	(2,0)
	(1,0)
	(0,0)
	(0,1)
	(0,2)
	(1,2)
	(1,1)
	(2,1)