

2018년 9월 29일  
제12회 E-PPER 프로그래밍 경진대회 (자바 부문)

학번 \_\_\_\_\_ 이름 \_\_\_\_\_ 비밀번호 \_\_\_\_\_

1. 사용자로부터  $n$ 개의 숫자를 입력받아  $i$ 번째 수부터  $j$ 번째 수까지 합을 구하는 프로그램을 작성하시오.

[입력 형식]

- 첫 번째 줄에 입력받을 숫자의 개수  $n$ 을 입력한다.  
( $1 \leq n \leq 100$ )
- 두 번째 줄에는  $n$ 개의 수를 공백으로 구분하여 입력한다.  
입력하는 수는 1000이하의 자연수이다.
- 세 번째 줄에 합을 구할 구간  $i$ 와  $j$ 를 입력한다.

[출력 형식]

- 두 번째 줄에 입력한  $n$ 개의 수중  $i$ 번째 수부터  $j$ 번째 수까지의 합을 출력한다.

[입력 예]

```
5
5 4 3 2 1
1 3
```

[출력 예]

```
12
```

2. 신입생 오리엔테이션에 참석할 학생들에게 (1번부터 시작되는) 번호표를 주었다. 이 번호표는 학생들이 머무르는 방의 번호를 결정하고, (게임 등을 하기 위해) 같은 방을 쓰는 학생들에게도 내부적으로 일련번호를 주기 위해 쓰인다. 각 방은 15명씩 배정되고, 같은 방에 배정된 학생들의 경우는 번호표 순으로 방안에서의 번호가 결정된다.

학생에게 줄 번호표를 입력받아 그 학생의 방 번호와 방안에서의 번호를 결정해 주는 프로그램을 작성하시오.

[입력 형식]

- 번호표  $n$ 을 입력한다. ( $1 \leq n \leq 1000$ )

[출력 형식]

- 방 번호와 방안에서의 번호를 공백으로 구분하여 순서대로 출력한다.

[입력 예1]

```
7
```

[출력 예1]

```
1 7
```

[입력 예2]

```
30
```

[출력 예2]

```
2 15
```

3. 영어 알파벳과 하이픈(-)으로 구성되고 첫 시작 문자와 하이픈 뒤에 시작하는 첫 번째 문자는 항상 대문자로 시작하는(나머지 문자는 모두 소문자) 메모가 있다. 이를 대문자들로만 구성된 간략한 형태의 메모로 변환하는 프로그램을 작성하시오.

[입력 형식]

- 최대 100글자의 영어 알파벳과 하이픈(-)으로 구성된 메모를 한 줄에 입력한다.

[출력 형식]

- 대문자들로 구성된 간략한 형태의 메모를 출력한다.

[입력 예]

```
My-Name-Is-Ewha
```

[출력 예]

```
MNIE
```

4. 홀수인 양의 정수  $n$ 이 주어지면, 별 문자 \*를 이용하여 가장 긴 행의 \* 문자의 수가  $n$ 개인 다이아몬드 모양을 출력하는 프로그램을 작성하시오.

[입력 형식]

- 첫째 줄에 홀수인 양의 정수  $n$ 을 입력한다.  
( $1 \leq n \leq 100$ )

[출력 형식]

- 다이아몬드 모양을 출력한다.

[입력 예]

```
5
```

[출력 예]

```

*
***
*****
***
*

```

5. 다장조는 c d e f g a b C, 총 8개 음으로 이루어져있다. 이 8개 음은 다음과 같이 숫자로 바꾸어 표현한다. c는 1로, d는 2로, ..., C를 8로 바꾼다. 1부터 8까지 차례대로 연주한다면 ascending, 8부터 1까지 차례대로 연주한다면 descending, 둘 다 아니라면 mixed 이다. 연주한 음의 순서가 숫자로 주어졌을 때, 이것이 ascending인지, descending인지, 아니면 mixed인지 판별하는 프로그램을 작성하시오.

[입력 형식]

- 8개의 숫자를 공백으로 구분하여 입력한다. (1부터 8까지의 숫자가 한 번씩 등장한다.)

[출력 형식]

- ascending, descending, mixed를 판별하여 출력한다.

[입력 예1]

8 7 6 5 4 3 2 1

[출력 예1]

descending

[입력 예2]

8 1 7 2 6 3 5 4

[출력 예2]

mixed

6. 두 개의 문자열이 있을 때, 하나의 문자열을 재배열하면 다른 문자열이 될 수 있는지를 체크하는 프로그램을 작성하시오.

[입력 형식]

- 두 개의 문자열을 공백으로 구분하여 입력한다. (각 문자열은 공백을 포함하지 않고, 문자열 길이가 100이하이다.)

[출력 형식]

- 재배열이 가능하면 Yes를 출력하고, 불가능하면 No를 출력한다. (대소문자 상관없이 재배열하여 만들 수 있는지 체크함)

[입력 예1]

Listen Silent

[출력 예1]

Yes

[입력 예2]

Ewha What

[출력 예2]

No

7. 아홉 개의 숫자 중 합이 100이 되는 일곱 개의 숫자를 찾는 프로그램을 작성하시오.

[입력 형식]

- 1 이상 100 미만인 아홉 개의 숫자를 공백으로 구분하여 입력한다. 모든 숫자는 서로 다르고 항상 답은 유일하다.

[출력 형식]

- 합이 100이 되는 일곱 개의 숫자를 찾아 출력한다.

[입력 예]

7 8 10 13 15 19 20 23 25

[출력 예]

7 8 10 13 19 20 23

8. 다음 체스판 위를 개미가 좌측 하단에서 시작하여 각 칸에 적힌 번호 순서대로 한 번에 한 칸씩 이동한다. 사용자가 특정 번호를 입력하면 시작점(1번 위치)에서부터 가로와 세로 방향으로 각각 얼마만큼 떨어진 위치에 있는지 출력하는 프로그램을 작성하시오.

25	24	23	22	21
10	11	12	13	20
9	8	7	14	19
2	3	6	15	18
1	4	5	16	17

[입력 형식]

- 첫째 줄에 입력받을 개수 n을 입력한다.  
- 다음 n개의 줄에 걸쳐 체스판의 번호를 입력한다.

[출력 형식]

- 입력받은 체스판 번호에 도달하기 위한 가로 칸수, 세로 칸수를 공백으로 구분하여 출력한다.

[입력 예]

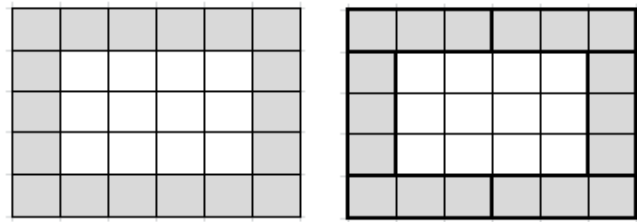
3  
8  
20  
25

[출력 예]

2 3  
5 4  
1 5

9.  $N \times M$  크기의 정원 바닥에 타일을 붙이려고 한다. 단, 타일을 아래 왼쪽 그림과 같이 바닥의 테두리에만 붙인다. 타일은  $P \times 1$  크기의 한 가지 종류만 사용한다고 할 때, 타일이 겹치는 부분 없이 바닥을 모두 덮는 것이 가능한  $P$ 를 구하는 프로그램을 작성하시오.

아래 그림의 정원을 보면,  $6 \times 5$  크기의 정원이고 오른쪽 그림의 굵은 선은  $3 \times 1$ 의 타일로 테두리를 붙인 예를 보여주고 있다. 즉  $P$ 가 3인 예이다.  $6 \times 5$  크기의 정원은  $1 \times 1$  타일,  $2 \times 1$  타일,  $3 \times 1$  타일로 겹치는 부분 없이 테두리 바닥을 모두 붙일 수 있으므로 구하고자 하는  $P$ 는 1, 2, 3 이 된다.



[입력 형식]  
- 정원 바닥의 가로  $N$ , 세로  $M$ 을 순서대로 공백으로 구분하여 입력한다. ( $1 \leq N, M \leq 5000$ )

[출력 형식]  
-  $P$ 들을 공백으로 구분하여 출력한다.

[입력 예1]                      [출력 예1]  

6 5

1 2 3

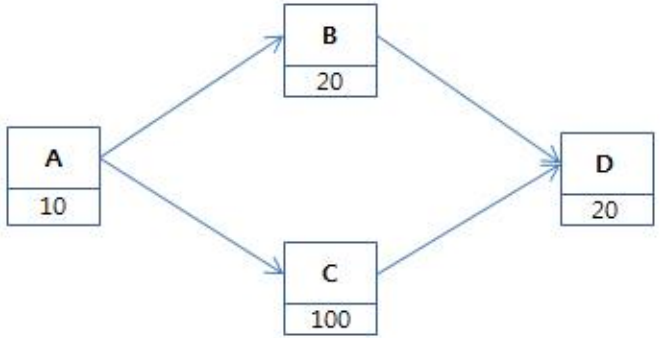
[입력 예2]                      [출력 예2]  

20 10

1 2 4

10.  $N$ 개의 작업 공정이 있다. 공정마다 소요되는 시간이 있고, 각 공정들 끼리 선후 관계가 존재할 때는 선행 공정이 끝나야만 다음 공정으로 넘어갈 수 있다.

예를 들어 아래 공정을 보면, A공정에 10이 소요되고 난 후, B와 C가 동시에 진행이 된다. 그렇게 되면 B공정이 끝나는 시점은 30이 되고, C공정은 110에 끝나게 된다. D공정은, B와 C가 모두 끝나는 시점인(=C가 끝나는 시점) 110에 시작하게 되고, 따라서 D공정은 130에 끝나게 된다.



이러한 원칙을 적용해서, 임의로 주어지는 공정에 대해 목표되는 공정까지 소요되는 최소시간을 구하는 프로그램을 작성하시오.

[입력 형식]  
- 첫 째줄에 공정수( $N$ )와 관계수( $R$ )를 입력한다. ( $1 \leq N, R \leq 100$ )  
- 그 다음 줄에 각 공정에서 소요되는 시간  $N$ 개를 입력하고, 그 다음 줄 부터 공정간의 관계가  $R$ 줄에 걸쳐 입력한다. 공정간의 관계는 연결되는 공정 번호가 "앞공정번호 뒤공정번호" 순으로 나온다.  
- 그 다음 줄에 목표되는 공정 번호를 입력한다.

[출력 형식]  
- 최소 소요시간을 출력한다.

[입력 예]                      [출력 예]  

4 4  
 10 20 100 20  
 1 2  
 1 3  
 2 4  
 3 4  
 4

130