

프로그래밍 과제 01

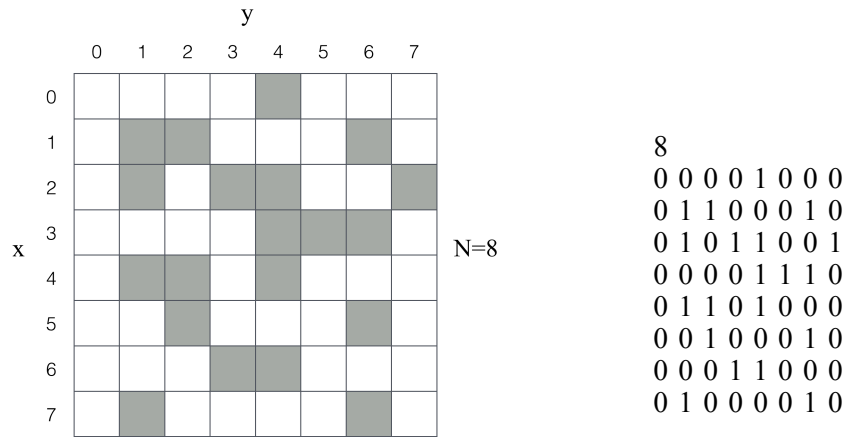
1. [Rank] 입력으로 $N < 1000$ 개의 정수 a_1, a_2, \dots, a_N 과 또 다른 하나의 정수 K 가 주어진다. 정수들은 정렬되어 있지 않고, 동일한 값이 중복될 수도 있다. 임의의 정수 K 의 rank란 N 개의 정수 a_1, a_2, \dots, a_N 중에서 K 보다 작은 것의 개수 + 1을 말한다. 정수 K 의 rank를 구해서 출력하는 프로그램을 작성하라. 단, 처음 N 개의 정수들을 입력받아 배열에 저장하는 부분을 제외한 다른 어떤 부분에서도 **for**문, **while** 문 등의 반복문을 사용해서는 안된다. 입력의 형식은 먼저 정수의 개수 N 이 주어지고, 이어서 N 개의 정수가 주어진다. 마지막으로 정수 K 가 주어진다. 시간복잡도가 $O(N)$ 을 초과해서는 안된다.

입력 예	출력
10 2 5 3 8 6 7 8 7 2 1	9
8 25 88 1 9 99 24 99 1 45 17 21 23 73 31 37 6 38 9 91 3 99 47 58 55 73 51	16
50 8 1 1 1 1 2 2 2 2 2	5

2. [Nearest] 오름차순으로 정렬된 $N < 1000$ 개의 정수 $a_0 \leq a_1 \leq \dots \leq a_{N-1}$ 과 또 다른 하나의 정수 K 가 입력으로 주어진다. 정수들 중에서 K 와 가장 가까운 정수를 찾는 함수 **nearest**를 구현하라. 가장 가깝다는 말을 차이의 절대값이 최소라는 의미이다. 답이 2개 이상인 경우에는 그 중 작은 값을 출력해야 한다. 함수 **nearest**의 시간복잡도는 $O(\log N)$ 이어야 한다. 처음 N 개의 정수들을 입력받아서 배열에 저장하는 부분을 제외한 나머지 부분에서는 **for**문이나 **while**문과 같은 반복문을 사용해서는 안된다.

입력 예	출력
10 1 2 3 4 5 6 7 8 9 10	10
11 25 1 3 6 9 13 17 21 23 24 31 37 38 44 45 47 51 55 58 71 73 88 91 99 101 102	71
72 40 1 17 19 23 25 28 41 44 49 50 61 64 65 67 71 77 79 81 82 83 84 90 91 92 96 99 101 103 109 121 128 132 133 150 152 161 165 167 177 178	109
111 2 2 2	2
1 2 2 2 3	2

3. [경로의 개수] 미로 찾기 문제에서 입구에서 출구까지 가는 길이가 K 이하인 서로 다른 경로의 개수를 계산하여 출력하는 프로그램을 순환(recursion)을 이용하여 작성하라. 경로의 길이는 이동 횟수를 의미한다. 단, 같은 위치를 2번 이상 방문하는 경로는 카운트하지 않는다. 입구의 위치는 $(0,0)$, 출구는 $(N-1, N-1)$ 이다. N 은 16 이하이다. (경로의 개수가 **int**의 표현범위를 넘는 경우는 없다고 가정해도 된다.)



입력 형식

표준입력파일(키보드)로 부터 입력을 받는다. 입력의 첫 줄에는 미로의 크기 N 이 주어진다. 이어진 N 줄에는 각 줄마다 N 개의 0 혹은 1이 한 칸씩 띄어져서 주어진다. 0은 통로, 1은 지나갈 수 없는 벽을 표시한다. 이어진 줄에는 정수 K 의 값이 주어진다.










출력 형식

경로의 개수를 표준출력파일(화면)로 출력한다.

입력 예	출력 예
<pre> 8 0 0 0 0 1 0 0 0 0 1 1 0 0 0 1 0 0 1 0 1 1 0 0 1 0 0 0 0 1 1 1 0 0 1 1 0 1 0 0 0 0 0 1 0 0 0 1 0 0 0 0 1 1 0 0 0 0 1 0 0 0 0 1 0 20 // N=8 // K=20 </pre>	6
<pre> 3 0 0 0 0 0 0 0 0 0 8 // N=3 // K=8 </pre>	12
<pre> 3 0 0 0 0 0 0 0 0 0 7 </pre>	10
<pre> 3 0 0 0 0 1 0 0 0 0 10 </pre>	2
<pre> 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 8 </pre>	56

5	43
0 0 0 0 0	
0 0 0 1 0	
0 1 0 0 0	
0 0 0 0 0	
0 0 1 0 0	
10	

4. [폭탄] 아래 그림과 같이 폭탄이 설치된 미로가 있다. 폭탄이 설치된 곳을 지나갈 수는 있지만 대신 부상을 당하는데, K 번을 초과하여 부상을 당하면 죽는다. 출구까지 죽지 않고 갈 수 있는지 검사하여 Yes 혹은 No를 출력하는 프로그램을 순환함수(recursion)를 이용하여 작성하라.

출발							
							
							
							
							도착

N=8

```

8
0 0 0 2 1 0 0 0
0 1 1 0 0 0 1 0
0 1 0 1 1 0 2 1
0 2 0 2 0 1 0 1
0 1 1 0 0 2 1 0
2 2 1 0 0 2 1 0
0 0 0 1 1 0 0 0
0 1 0 0 2 0 1 0

```

입력 형식

표준입력파일(키보드)로부터 입력을 받는다. 입력의 첫 줄에는 미로의 크기 N 이 주어진다. 이어진 N 줄에는 각 줄마다 N 개의 0, 1, 혹은 2가 한 칸씩 띄어져서 주어진다. 0은 통로, 1은 지나갈 수 없는 벽, 그리고 2는 폭탄을 표시한다. 마지막 줄에는 정수 K 의 값이 주어진다.

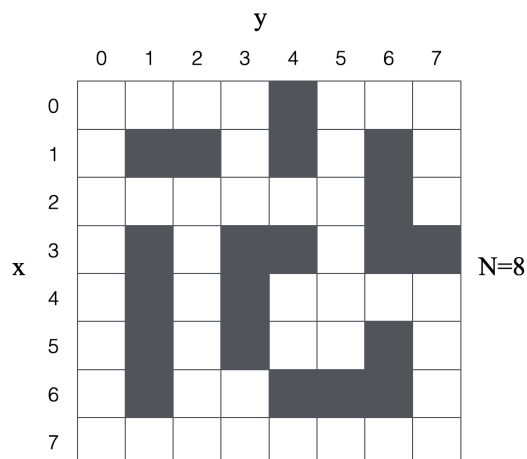
출력 형식

Yes 혹은 No라고 출력한다.

입력 예	출력 예
<pre> 8 // N=8 0 0 0 2 1 0 0 0 0 1 1 0 0 0 1 0 0 1 0 1 1 0 2 1 0 2 0 2 0 1 0 1 0 1 1 0 0 2 1 0 2 2 1 0 0 2 1 0 0 0 0 1 1 0 0 0 0 1 0 0 2 0 1 0 2 // K=2 </pre>	Yes

8 0 0 0 2 1 0 0 0 0 1 1 0 0 0 1 0 0 1 0 1 1 0 2 1 0 2 0 2 0 1 0 1 0 1 1 0 0 2 1 0 2 2 1 0 0 2 1 0 0 0 0 1 1 0 0 0 0 1 0 0 2 0 1 0 1	No
4 0 2 2 2 2 2 2 2 2 2 2 2 0 2 2 0 4	Yes
5 0 0 2 2 2 0 2 2 1 2 0 1 0 0 0 0 2 2 2 0 0 0 0 1 0 1	Yes

5. [휴식횟수] 크기가 $N \times N$ 인 미로와 하나의 양의 정수 k 가 입력으로 주어진다. 흰색은 통로, 검은색은 벽이고 입구의 위치는 $(0,0)$ 이고 출구는 $(N-1, N-1)$ 이다. 이동하는 규칙은 다음과 같다. 현재 위치에서 일직선으로 최대 k 칸까지 한 번에 이동할 수 있다. 예를 들어 아래 그림에서 $k=3$ 이고 현재 위치가 $(2, 3)$ 이라면, $(2, 0)$, $(2, 1)$, $(2, 2)$, $(2, 4)$, $(2, 5)$, $(1, 3)$, $(0, 3)$ 으로는 한 번에 이동할 수 있다는 의미이다. 한 번 이동하면 반드시 휴식을 취해야 한다. 입구에서 출발하여 출구까지 가는 휴식 횟수가 최소인 경로를 찾아 최소 휴식 횟수를 출력하는 프로그램을 작성하라. 출구까지 가는 경로가 존재하지 않으면 -1 을 출력한다. 입력파일의 이름은 **maze.txt**이고, 파일의 첫 줄에는 $N \leq 20$ 의 값이 주어지고, 이어진 N 줄에는 각 줄마다 N 개의 0 혹은 1이 한 칸씩 띄어져서 주어진다. 0은 통로, 1은 벽을 나타낸다. 마지막으로 양의 정수 k 가 주어진다. 출력은 화면으로 한다.

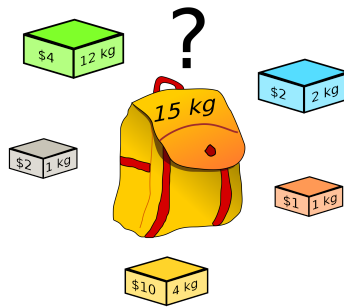


입력의 예	출력
8 2 4 3 2 1 0 4 3	No
5 2 1 0 2 3	No

7. [이진수열] 입력으로 하나의 양의 정수 $n \leq 16$ 을 받는다. 길이가 n 인 이진수열 중에서 0이 연속해서 등장하지 않는 이진수열의 개수를 계산하여 출력하는 프로그램을 작성하라. 예를 들어 $n = 2$ 라면 01, 10, 11로 3개의 수열이 존재한다. 입력은 키보드로 주어지고, 출력은 화면으로 한다.

입력의 예	출력
1	2
2	3
5	13
6	21
7	34

8. [Knapsack] 배낭채우기(knapsack) 문제는 다음과 같이 정의된다. $N \leq 16$ 개의 아이템(item)이 있다. 각 아이템은 가격과 무게가 정해져 있다. w_0, w_1, \dots, w_{N-1} 을 아이템의 무게라고 하고 v_0, v_1, \dots, v_{N-1} 을 각 아이템의 가격이라고 하자. 배낭의 용량은 W 이다. N 개의 아이템들 중에서 무게의 총량이 W 를 초과하지 않으면서 가격의 합이 최대가 되는 부분집합을 찾아서 가격의 최대 합을 출력하는 프로그램을 작성하라.



입력 형식

입력은 `input.txt` 파일로부터 받는다. 파일의 첫 줄에는 아이템의 개수 $N \leq 16$, 다음 줄에는 W 가 주어지고, 다음 줄에는 N 개의 아이템의 무게가 주어지고, 다음 줄에는 N 개의 아이템의 가격이 주어진다. 무게와 가격은 모두 1,000 이하의 양의 정수이다.

출력 형식

가격의 최대 합을 화면으로 출력한다.

입력 파일의 예	출력 예
6 // N 30 // W 5 10 8 6 9 3 // w 20 40 30 25 35 15 // v	120 [1, 1, 0, 1, 1, 0] (각 아이템의 선택 여부를 표시. 이것을 출력할 필요는 없다)
10 45 2 6 12 8 15 10 9 11 4 7 10 25 35 20 45 30 28 33 15 23	151 [1, 1, 0, 0, 1, 0, 0, 1, 1, 1]

15 60 4 12 8 5 10 14 7 11 6 9 15 3 13 2 1 15 35 25 20 30 40 22 33 18 28 45 10 38 8 5	196 [1, 0, 1, 1, 1, 0, 1, 1, 0, 1, 0, 1, 0, 1, 1] 0, 1, 1]
12 40 6 8 5 11 3 10 9 7 12 4 13 2 25 30 20 35 15 33 28 22 40 18 38 10	158 [1, 1, 1, 0, 1, 0, 0, 0, 1, 1, 0, 1] 1, 1, 0, 1]

9. [팀구성] $N < 16$ 명의 선수 후보들이 있다. 이 중 $k \leq N$ 명을 선택하여 출전 팀을 구성하려고 한다. 각 선수들은 자신의 고유한 능력치가 있지만 또한 누구와 팀이 되느냐에 따라서 능력이 증진되기도 하고 저하되기도 한다. 이 정보가 하나의 $N \times N$ 배열 A 로 주어진다. 선수 i 의 고유한 능력치는 $A[i][i]$ 에 저장되고, $A[i][j], i \neq j$,는 선수 i 가 선수 j 와 동시에 팀에 속했을 때 선수 i 의 능력치의 증가(양수) 혹은 감소(음수) 정도를 나타낸다. $A[i][j]$ 와 $A[j][i]$ 는 다를 수 있다. 예를 들어 $N = 8, k = 4$ 이고, 선발된 선수들이 $\{0, 2, 4, 7\}$ 이라면 이 팀의 능력치는

$$\begin{aligned}
 &A[0][0] + A[0][2] + A[0][4] + A[0,7] \\
 &+ A[2][0] + A[2][2] + A[2][4] + A[2,7] \\
 &+ A[4][0] + A[4][2] + A[4][4] + A[4,7] \\
 &+ A[7][0] + A[7][2] + A[7][4] + A[7,7]
 \end{aligned}$$

이 된다. k 명으로 이루어진 능력이 최대가 되는 팀을 구성하여 이 팀의 능력치를 출력하는 프로그램을 작성하라.

입출력 형식

입력은 input9.txt 파일로 부터 받는다. 파일의 첫 줄에는 정수 N 이 주어지고 이어진 N 줄에는 각각 N 개의 정수들이 주어진다. 마지막 줄에는 정수 k 가 주어진다. 출력은 화면으로 한다.

입력 예	출력	참고(출력할 필요 없음)
4 2 3 1 5 1 4 -2 2 0 -1 3 5 2 4 2 1 3	24	정답의 선수 구성: [0, 1, 3] (선수 번호는 0번에서 시작)
4 1 -4 -2 -5 0 2 -3 -2 -2 -3 4 -2 -1 -3 -4 6 3	-5	정답의 선수 구성: [0, 2, 3]
8 6 -2 8 -5 3 6 0 -2 8 4 -12 8 3 6 -18 23 -8 7 2 16 4 -8 -2 7 6 4 1 9 8 2 9 3 -2 8 3 6 4 -2 -8 6 3 -2 4 7 2 8 9 12 2 8 3 -6 4 -2 8 6 -4 3 2 9 4 6 9 8 5	149	정답의 선수 구성: [1, 3, 4, 5, 7]

입력 예	출력	참고(출력할 필요 없음)
10 2 -3 4 8 6 -2 9 -3 4 6 -9 2 3 4 2 4 -2 3 -9 7 4 -2 7 3 4 7 -2 -9 3 -4 7 2 9 3 8 7 4 8 9 2 2 -3 -8 7 4 6 -2 8 7 4 -6 1 2 9 8 4 9 1 2 8 4 7 9 -4 2 -3 9 4 7 2 3 9 4 7 2 9 8 3 4 -7 9 2 8 3 7 -4 9 2 3 7 2 3 -8 9 4 9 8 2 7 3 6	181	정답의 선수 구성: [0, 3, 4, 6, 8, 9]

10. [이사] 현재 살고 있는 집에는 총 $N \leq 20$ 개의 물건이 있다. 각 물건에 대해서 부피 w_i , 가격 v_i , 그리고 그 물건을 버릴 때 드는 폐기 비용 c_i 가 주어진다. 현재 집에 있는 N 개의 물건들 중 일부를 새로운 집으로 가지고 가려고 한다. 가지고 가지 않는 물건들은 폐기 비용을 지불하고 폐기해야 하며, 가져갈 물건의 부피의 총량은 W 를 초과해서는 안된다. 가져가는 물건의 가격의 합에 가져가지 않을 물건의 폐기 비용을 뺀 값의 최대값을 찾는 프로그램을 작성하라.

입출력 형식

입력은 input10.txt 파일로 부터 받는다. 파일의 첫 줄에는 정수 N 이 주어지고 이어진 N 줄에는 각 줄마다 하나의 물건에 대한 정보가 3개의 정수로 주어진다. 세 정수는 물건의 부피(w_i), 가격(v_i), 그리고 폐기비용(c_i)의 순서이다. 마지막 줄에는 정수 W 가 주어진다. 출력은 화면으로 한다.

입력 예	출력	참고
4 2 3 1 // 부피, 가격, 폐기비용 1 4 2 1 1 3 2 4 2 4	8	정답에서 선택된 물건: [1, 2, 3] (물건 번호는 0번에서 시작)
8 3 7 9 5 19 8 8 6 1 13 1 9 1 3 8 7 11 7 4 16 1 11 5 6 35	47 51	정답에서 선택된 물건: [0, 1, 2, 4, 5, 6]
12 2 3 4 8 6 2 9 3 4 6 9 2 4 2 4 2 3 9 7 4 2 7 3 4 7 2 9 3 4 7 2 9 3 8 7 4 50	29 42	정답에서 선택된 물건: [0, 1, 3, 5, 8, 9, 10, 11]