

unique한 문자열

N개이 문자열이 주어지면 각 문자열에서 특정문자(알파벳)가 존재하는 문자열을 찾아 그 문자열을 “unique한 문자열”로 만들어 출력하는 프로그램을 작성하세요. 특정문자는 대소문자를 구별하지 않습니다.

여기서 “unique한 문자열”이란 해당 문자열을 모두 대문자로 통일하고, 그 순서는 유지하면서 중복된 문자는 한 번만 사용되게 만든 문자열입니다. 만약 어떤 문자열이 "Remember" 라면 이 문자열의 “unique한 문자열”은 “REMB”입니다.

■ 입력설명

첫 줄에 한 줄에 자연수 $N(3 \leq N \leq 100)$ 과 특정문자가 주어집니다.

두 번째 줄에 N개의 줄에 걸쳐 문자열이 N개 입력됩니다.

■ 출력설명

첫 줄부터 특정문자가 있는 문자열을 unique한 문자열로 변환한 다음 출력합니다.

■ 입력예제 1

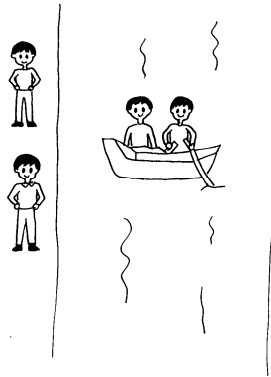
5 b
Queen
Apple
Best
Remember
Beer

■ 출력예제 1

BEST
REMB
BER

강건너기

1번, 2번, 3번, 4번으로 구분하는 네 명의 사람이 강을 건너려고 합니다. 강에는 배가 한 척밖에 없고, 그 배에는 최대 두 명이 탈 수 있습니다. 혼자서 배를 타고 노를 저어 강을 건널 경우 1번은 1분, 2번은 3분, 3번은 7분, 4번은 10분이 걸립니다. 둘이 함께 타고 건널 경우 안전을 위하여 더 천천히 노를 젓는 사람이 노를 잡습니다. 만약 1번과 2번이 함께 타고 강을 건너면 3분이 걸립니다. 네 사람이 강을 건너려면 최소 몇 분이 걸릴까요?



1번부터 N번까지의 N명의 노를 저어 강을 건너는 시간이 주어지면 N명의 사람이 모두 강을 건너는데 걸리는 최소시간을 출력하는 프로그램을 작성하세요.

■ 입력설명

첫 번째 줄에 자연수 $N(4 \leq N \leq 100)$ 이 주어지고 그 다음줄에는 1번 사람부터 차례대로 강을 건너는데 걸리는 시간이 주어집니다.

■ 출력설명

최소시간을 출력하세요. (단위는 분입니다)

■ 입력예제 1

4

1 3 7 10

■ 출력예제 1

20

전투 게임

현수네 반 학생들은 몇 개의 팀으로 나뉘어 전투게임을 합니다.

각 학생들은 반 번호, 팀, 공격력을 가지고 있습니다. 각 학생들은 다른 팀의 학생 중 자기보다 공격력이 작은 학생을 사로잡을 수 있습니다. 다른 팀의 학생을 사로잡았다고 해서 본인의 공격력이 올라가는 것은 아닙니다. 공격력은 변하지 않습니다.

다음 예는 학생이 5명이 있을 때의 경우입니다.

번호	팀	공격력
1	a	20
2	b	12
3	a	10
4	c	11
5	e	12

1번 학생의 경우 2, 4, 5번 학생을 사로잡을 수 있고, 3번 학생은 같은 팀이라 사로잡을 수 없습니다. 2번 학생은 3, 4번 학생을 사로잡을 수 있습니다. 각 학생은 자신이 사로잡을 수 있는 학생의 공격력만큼 점수를 획득합니다. 1번 학생의 경우 2, 4, 5번 학생을 사로잡을 수 있으므로 2, 4, 5번의 공격력을 합한 $12+11+12=35$ 만큼 점수를 획득할 수 있습니다.

모든 학생의 팀과 공격력이 주어지면, 각 학생마다 얻을 수 있는 점수를 출력하는 프로그램을 작성하세요.

입력설명

첫 줄에는 반 학생수를 나타내는 자연수 N 이 주어진다($1 \leq N \leq 200,000$).

다음 N 줄에 걸쳐 i 번째 줄에는 i 번째 학생의 팀($a \sim z$)과 공격력($1 \sim 100$)이 주어집니다.

출력설명

N 개의 줄을 출력합니다. N 개의 줄 중 i 번째 줄에는 i 번째 학생이 사로잡을 수 있는 학생들의 공격력의 총합을 출력합니다.

▣ 입력예제 1

7

a 17

b 12

a 10

c 11

b 24

a 25

b 12

▣ 출력예제 1

35

21

0

10

38

59

21

스도쿠(DFS 활용)

스도쿠는 9×9 크기의 보드가 있을 때, 각 행과 각 열, 그리고 9개의 3×3 크기의 보드에 1부터 9까지의 숫자가 중복 없이 나타나도록 보드를 채우는 퍼즐이다.

1	2	3	4	5	6	7	8	9
4	5	6	7	8	9	1	2	3
7	8	9	1	2	3	4	5	6
2	1	4	3	6	5	8	9	7
3	6	5	8	9	7	2	1	4
8	9	7	2	1	4	3	6	5
5	3	1	6	4	2	9	7	8
6	4	2	9	7	8	5	3	1
9	7	8	5	3	1	6	4	2

위 그림은 스도쿠를 정확하게 풀어낸 것이다. 각 행에 1부터 9까지의 숫자가 중복 없이 나오고, 각 열에 1부터 9까지의 숫자가 중복 없이 나오고, 각 3×3짜리 사각형(9개이며, 위에서 색깔로 표시되었다)에 1부터 9까지의 숫자가 중복 없이 한 번씩 나오기 때문이다.

완성되지 않은 스도쿠 퍼즐이 주어졌을 때, 스도쿠를 정확하게 완성하는 프로그램을 작성하세요.

■ 입력설명

9*9 보드정보가 주어집니다. 아직 숫자가 채워지지 않은 칸은 0으로 표시됩니다.

■ 출력설명

9개의 줄에 9개의 숫자로 답을 출력한다. 답이 여러 개 있다면 그 중 오름차순으로 제일 먼저 발견된 것을 출력하면 됩니다. 즉, 81자리의 수가 제일 작은 경우로 출력하는 것입니다.

■ 입력예제 1

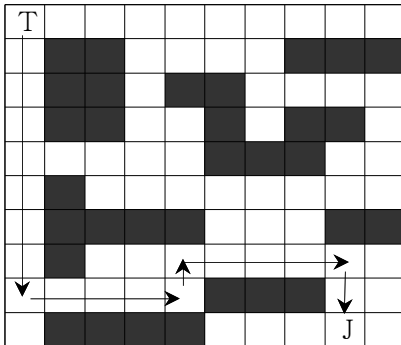
```
0 2 3 0 5 0 7 8 9
0 5 6 0 8 9 1 0 3
0 8 9 1 0 3 0 5 6
0 1 0 0 6 0 8 9 0
3 0 5 0 9 7 0 1 4
0 9 7 0 1 0 0 6 5
5 3 0 6 0 2 9 7 8
6 0 2 9 0 8 5 3 1
9 0 8 0 3 0 6 0 2
```

■ 출력예제 1

```
1 2 3 4 5 6 7 8 9
4 5 6 7 8 9 1 2 3
7 8 9 1 2 3 4 5 6
2 1 4 3 6 5 8 9 7
3 6 5 8 9 7 2 1 4
8 9 7 2 1 4 3 6 5
5 3 1 6 4 2 9 7 8
6 4 2 9 7 8 5 3 1
9 7 8 5 3 1 6 4 2
```

통과 제리(BFS) :

모두가 잘 알고 있듯이 통과 제리는 별로 가까운 사이가 아니다. 항상 통은 제리를 쫓아다녔지만 그때마다 번번이 실패하고 말았다. 곰곰이 꾀를 내던 통은 제리가 자고 있을 때 가장 빠른 길로 잰싸게 달려가면 제리를 잡을 수 있겠다 싶었다. 그런데 그러기 위해서는 가장 적은 횟수로 방향을 바꾸어서 제리가 있는 곳까지 가는 방법을 알아야만 한다. 통과 제리가 있는 정사각형 모양의 방의 상태와, 둘의 위치가 주어질 때 이와 같은 경로를 찾아내는 프로그램을 작성하세요.



입력설명

첫째 줄에 방의 너비를 나타내는 100이하의 자연수 n 이 주어지고 다음 n 줄에는 방의 상태가 입력된다. 지나갈 수 있는 위치는 0으로, 장애물이 있는 위치는 1로 표현된다. 다음 마지막 두 줄에는 통의 위치와 제리의 위치가 주어진다. 각각의 위치는 행과 열 순이다.

출력설명

첫 줄에 방향을 바꾸는 회수를 출력한다. 만약 갈 수 있는 길이 하나도 없을 경우에는 -1을 출력한다.

입력예제 1

```
10
0 0 0 0 0 0 0 0 0 0
0 1 1 0 0 0 0 1 1 1
0 1 1 0 1 1 0 0 0 0
0 1 1 0 0 1 0 1 1 0
0 0 0 0 0 1 1 1 0 0
0 1 0 0 0 0 0 0 0 0
0 1 1 1 1 0 0 0 1 1
0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 0 0
0 1 1 1 1 0 0 0 0 0
1 1
10 9
```

출력예제 1

```
4
```