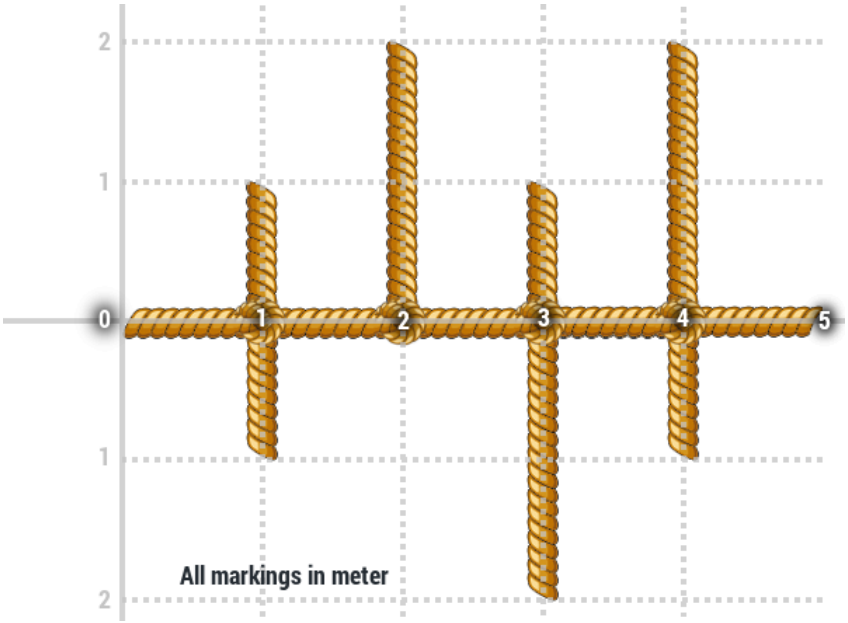
Arpita and Ropes

locked

Arpita has a rope of length L meters. This rope has several other ropes attached to it at the end of every meter (except for the end of the rope). At each meter there are two ropes attached to this main rope, let's call them upper and lower ropes. See the following example.



Arpita lit the rope on fire from the left end. This fire burns down the rope by 1 meter/minute. Your task is to find how much time (in minutes) will the fire take to burn down the entire rope.

Note:

Fix the code in the editor so that it passes all the test cases.

You can edit/modify at most `6` lines in this problem. Inserting or deleting any lines is not allowed. Blank lines will be ignored. You will get error message if you modify more than `6` lines even if your output matches the expected output.

Input Format

First line contains *T* - number of test cases. First line of each test case contains *L* length of the rope. Second line of each test case contains *L − 1* integers separated by space denoting lengths of all the upper ropes at each meter. Third line of each test case contains *L − 1* integers separated by space denoting lengths all the lower ropes at each meter.

Constraints

$1 \le T \le 10$

$2 \le L \le 1000000$

$0 \le upper[i] \le 1000000$ where $1 \le i \le L - 1$

$0 \le lower[i] \le 1000000$ where $1 \le i \le L - 1$

Output Format

Output the time (in minutes) required to burn down the entire rope for each test case in a new line.

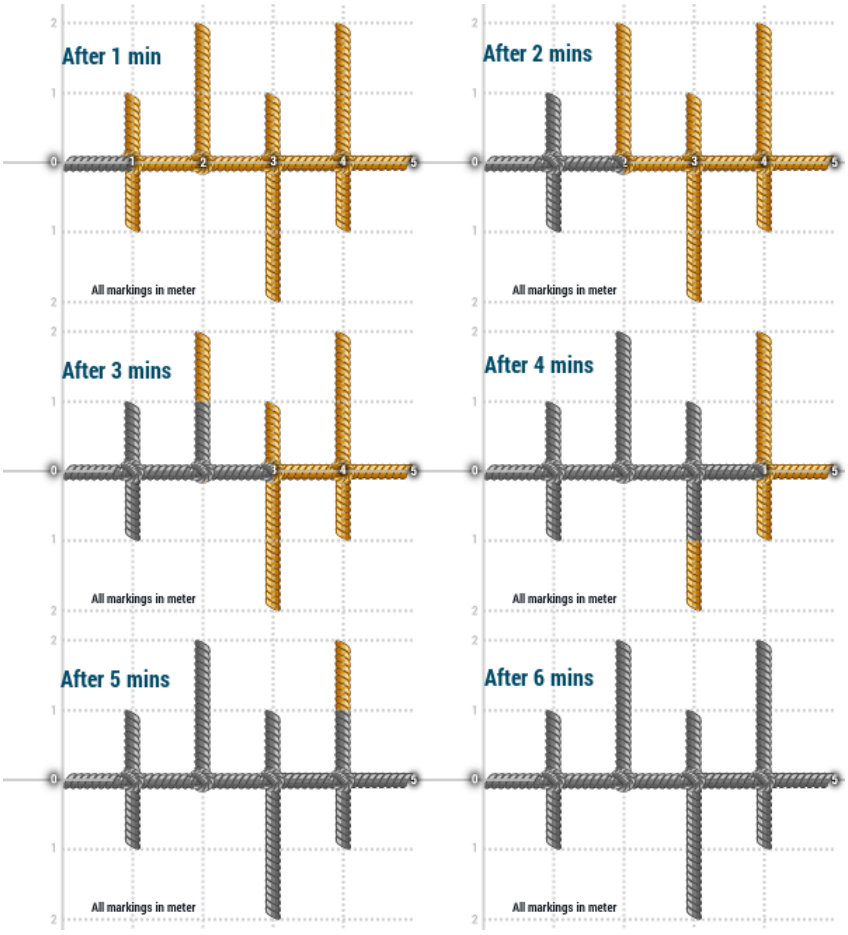
Sample Input 0

```
1
5
1 2 1 2
1 0 2 1
```

Sample Output 0

```
6
```

Explanation 0



Submissions: 32
Max Score: 100

Rate This Challenge:

More

Current Buffer (saved locally, editable) C

```c
#include<stdio.h>
#include<math.h>
int main(){
    int t,max=0,n,i,x;
    scanf("%d",&t);
    while(t--){
        scanf("%d",&n);
        max=0;
        for(i=1;i<n-1;i++){
            scanf("%d",&x);
            if((i+x)>max)
                max=i;
```

```
        }
        for(i=1;i<n-1;i++){
            scanf("%d",&x);
            if(x>max)
                max=i;
        }
        printf("%d\n",max);
    }
}
```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code