

Padel Game (padel)

Two teams are playing padel and it is known that they have played N points in total, and we know the winner of each point (team 1 or 2).

Given that they have played a number of warmup points (it could be also 0 or N) followed by competitive points (the remaining points) find the maximum number of complete games they have played. A complete game ends when a team wins at least 4 competitive points and the difference is at least 2 competitive points (just like in tennis).




Figure 1: Typical rally in a padel game

For example, if we have the sequence 1 2 1 2 1 2 1 1 2 2 1 2 2, then we would have 2 complete games if we choose the first 2 points as warm-ups and the rest to be games (1 2 1 2 1 1, respectively 2 2 1 2 2).

Take note that it is not required for the players to finish playing at the end of a game, as they can be forced to leave the field even in the middle of the game (for example, another group of players could start playing).

These teams are playing in T sessions and we are required to find the maximum number of games they could have played in each of them, if they chose the length of the warmup optimally.

 Among the attachments of this task you may find a template file `padel.*` with a sample incomplete implementation.

Input

The first line of the input contains T , the number of tests ($1 \leq T \leq 10$).

Each test case contains two lines. The first line contains N , the number of points they played in the session ($1 \leq N \leq 2000$). The next line contains the N points played, for which we know the winning team (1 or 2).

Output

The output will contain T lines, for each of them we will print the maximum number of games they could





have played if they chose the length of the warmup optimally.

Constraints

- $1 \leq T \leq 10$.
- $1 \leq N \leq 2000$.

Scoring

Your program will be tested against several test cases grouped in subtasks. In order to obtain the score of a subtask, your program needs to correctly solve all of its test cases.

- Subtask 1 (0 points) Examples.

- Subtask 2 (20 points) $N \leq 20$.

- Subtask 3 (30 points) $N \leq 30$.

- Subtask 4 (50 points) No additional limitations.


Examples

input	output
5 13 1 1 1 1 1 2 2 1 2 1 2 2 2 10 1 2 1 2 1 1 1 2 1 2 20 2 2 1 1 2 1 1 2 2 2 2 2 1 2 2 1 1 1 1 2 14 1 1 2 2 1 2 2 1 1 2 1 2 2 1 12 1 1 1 1 1 2 1 1 2 2 2 1	2 1 3 1 2

Explanation

In the **first test case**, they can achieve the maximum number of games played by choosing not to warmup at all, thus playing two games.

In the **second test case**, they can't play more than one game no matter how they choose the length of the warmup.

In the **third test case**, they can achieve the maximum number of games played by choosing to warm up for the first four points, thus playing three games.