

식단 관리

인하체력단련실에서는 고객들의 칼로리 섭취를 관리하기 위해 간식 식단관리 서비스를 시행하려 한다. 특정 날짜에 고객이 섭취한 간식의 총 칼로리가 이전 d 일 동안 해당 고객이 섭취한 간식의 칼로리의 중앙값(median) 대비 2배 이상인 경우 고객에게 경보를 보낸다. n 일 동안 고객의 일일 간식 칼로리 섭취량들이 주어졌을 때, 고객이 총 몇 번의 경보를 받는지 출력하시오. 단, 중앙값은 모든 숫자를 오름차순으로 정렬하여 찾을 수 있다. 홀수 개의 원소가 있으면 가운데 숫자가 선택되며, 짝수 개의 원소가 있으면 중앙값은 두 중간 값의 평균으로 정의된다. (1, 3, 3, 6, 7, 8, 9의 중앙값은 6이며, 1, 2, 3, 4, 5, 6, 8, 9의 중앙값은 $(4+5)/2 = 4.5$ 이다.)

예를 들어, $n = 5$ 일 동안 고객의 간식 섭취량이 [10, 20, 30, 40, 50]일 때 이전 $d = 3$ 일 동안의 섭취량을 고려하여 경보를 보낸다면, 4일째부터 고객에게 경보가 보내질 수 있다. 즉, 3일 이내에는 경보가 보내지지 않는다. 4일째에 이전 3일 동안의 간식 섭취량이 [10, 20, 30]이며, 해당 섭취량의 중앙값은 20으로 4일째의 섭취량과 비교했을 때, " $40 \geq 2 \times 20$ "이 참이므로 경보를 보낸다. 다음 날인 5일째에는 이전 3일 동안의 간식 섭취량이 [20, 30, 40]으로 5일째의 섭취량과 비교했을 때, " $50 \geq 2 \times 30$ "이 거짓이므로 경보를 보내지 않는다. 따라서 5일 동안 해당 고객은 1번의 경보를 받는다.

※ 프로그램의 실행 시간은 1 초, 메모리 사용량은 512MB 를 초과할 수 없다.

사용할 수 있는 언어는 C, C++로 제한한다. C++의 경우 main 함수 내의 시작 지점에 다음 내용을 추가함으로써 cin, cout 의 입출력 속도를 개선할 수 있다.

```
ios_base::sync_with_stdio(false);
cin.tie(NULL);
cout.tie(NULL);
```

단, 위의 내용을 추가할 경우 cin, cout 만 사용해야 하며, scanf, printf 등 C 입출력을 혼용해서 사용하면 안된다. C++의 std::endl 의 경우 출력 속도가 느리므로, cout<<endl; 대신 cout<<"\n";을 사용하는 것을 권장한다.

입력

첫 번째 줄에는 테스트 케이스 수 T ($1 \leq T \leq 100$)가 주어진다.

이후 각 테스트 케이스의 정보가 다음과 같이 주어진다.

- 첫 번째 줄에 총 날짜 수 n ($1 \leq n \leq 5,000$)일과 섭취량을 고려할 날짜 범위 d ($1 \leq d < n$)일이 공백 하나를 사이에 두고 주어진다.
- 두 번째 줄에 i ($1 \leq i \leq n$)일에 섭취한 칼로리 $calorie[i]$ ($0 \leq calorie[i] \leq 200$)이 차례로 공백 하나를 사이에 두고 주어진다.

출력

각 테스트 케이스마다 고객이 몇 번의 경보를 받는지 한 줄에 출력한다.

예제 입출력

예제 입력	예제 출력
2	2
9 5	0
2 3 4 2 3 6 8 4 5	
5 4	
1 2 3 4 4	

3 int to-erase

3 4 2 3 - 2 3 4

0 1 2 3 4

0	0	2	2	1
---	---	---	---	---