ETH Zurich
Institut für Theoretische Informatik
Prof. Dr. Angelika Steger
Prof. Dr. Emo Welzl
Prof. Dr. Peter Widmayer

Algorithms Lab

Exercise – *High School Teams*

You probably recall how teams were built in physical education back in high school. The classical method works as follows: first, the gym teacher chooses two people to be team captains. Then the captains take turns choosing a not yet chosen student, until nobody is left. Usually, this way, the strong students get chosen first while weak and/or unpopular students get chosen last.

This method has several obvious disadvantages. For one, it is very bad for the self-esteem of the people who get chosen last. Additionally, there is no guarantee that it results in teams that are even remotely of the same strength. For example, if one player is much better than all others, it would make sense to balance the teams by giving more good players to the team not containing that player. Generally speaking, it is much better if the gym teacher chooses the teams in a way that seems more or less fair.

Your gym teacher buddy wants to switch to this new method. But there is one problem: how can he know whether the teams are really fair? So he asks you for help. By observing his students, he has managed to assign each student an integer that represents their skill level (some students are so bad that they have a negative skill level). The final goal is to choose two teams (team red and team blue) such that the sum of the skills are equal for both teams. It is ok if some students are not selected at all (that is, they do not belong to any team). Those students can act as referees for the game. It is even ok if some team (or both) does not have any players at all. Such an empty team has a skill level of zero. Of course, it would be stupid to have too many referees and almost no players. Therefore, your friend gives you as a side constraint that at most k students should be non-players.

More formally, by a *fair choice of teams* we mean an ordered pair (R, B) where R and B are disjoint subsets of the student set satisfying the following two conditions:

label=(0) The sum of the skill levels of the players in R equals the sum of the skill levels of those in B, and

lbbel=(0) at most k students do not belong to the union of R and B.

Your task is to count the number of possible fair choices of teams, given k and the skill levels of the students.

Input The first line of the input contains the number $t \leq 10$ of test cases. Each of the t test cases is described as follows.

- It starts with a line that contains two integers `n k`, separated by a space. They denote
  - n, the number of students ($1 \leq n \leq 22$);
  - k, the upper bound on the number of non-players ($0 \leq k \leq n$).
- The following line defines the skill levels of the students. It contains n integers $s_0, \ldots, s_{n-1}$, separated by a space and such that $\sum_{i=0}^{n-1} |s_i| < 2^{31}$.

Output For each test case output one line with a single integer that denotes the number of different ways to make a fair choice of teams. We guarantee that this number can be represented by a signed 64 bit integer.

Points There are five groups of test sets, worth 100 points in total.

1. For the first group of test sets, worth 10 points, you may assume that $n \leq 12$.
2. For the second group of test sets, worth 20 points, you may assume that $n \leq 18$ and that $k \leq 3$.
3. For the third group of test sets, worth 40 points, you may assume that $k \leq 3$.
4. For the fourth group of test sets, worth 10 points, there are no additional assumptions.
5. For the fifth group of test sets, which is hidden and worth 20 points, there are no additional assumptions.

Corresponding sample test sets are contained in `test`i`.in/out`, for $i \in \{1, 2, 3, 4\}$.

Sample Input

```
3
5 0
5 0 -1 10 9
6 3
67108864 0 67108864 8 16 -32
4 4
0 1 0 0
```

Sample Output

```
0
4
27
```