Problem A: The insider

Javid is on a mission to win money as a professional competitive programmer. Instead of solving the problems himself during the contest, he has a much better plan: bribe the coach to get a sneak peek at the problems, and then he can solve them all beforehand! Since he's competing for cash, he wants to determine how many problems he should bribe the coach to see.

The contest has N problems of varying value. Javid must bribe the coach with \$420 for every problem that he wants to solve. Fortunately placing higher during the contest is never worth less money, so there's incentive to solve lots of the high-value problems.

Javid knows that his C different competitors are each going to have some score at the end of the contest S_i. Since life is unfair, in order to place higher than another person Javid needs **strictly more points**: Javid will place at the bottom of all tied scores.

What is the maximum profit that Javid can attain during this contest?

Input Specification:

The input begins with an integer $T \leq 100$, the number of test cases. Each test case begins with two space-separated integers $1 \leq N \leq 100$, the number of problems, and $1 \leq C \leq 100$, the number of other contestants. Following this is a line of N space-separated integers $1 \leq V_i \leq 100$, representing the point values of the problems. The next line contains the space-separated integers of the scores S_i that the C other contestants are guaranteed to get $(0 \leq S_i \leq \sum V_i)$. Finally we have a line of $C + 1$ space-separated integers in non-increasing order, representing the prize money P_i for placing i^{th} in the contest $(0 \leq P_i \leq 100000)$.

Output Specification:

For each test case output an integer on its own line: the maximum profit attainable.

Sample Input:

```
1
3 7
1 2 3
0 1 2 3 4 5 6
10000 1000 900 800 500 450 5 1
```

Sample Output:

```
80
```