Gotta ~ Catch 'Em All

Time limit: *1000 ms*
Memory limit: *256 MB*

Diglett is a ground-type Pokémon. Diglett digs regularly through the earth at a shallow depth, leaving perfectly tilled soil in its wake. Diglett lives in the forest, in caves under the earth, where it feeds on tree roots.

A greedy construction company is planning to build a tunnel near the Viridian city and they know that there are a lot of Digletts in the Viridian forest near the Viridian city. Instead of spending money on machines to dig a tunnel for the company, the company has decided to capture Digletts from the Viridian forest in order to use them for digging the tunnels so that they can reduce spending and increase profit margins. However, Digletts do not want to dig for a greedy construction company and want to be free.

Digletts can only be captured if they are on the surface and can escape by going into a hole. Whenever the Diglett senses a human presence, it goes into a hole. There are M holes and N Digletts. Initially, a hole can only accommodate a single Diglett. However, a Diglett can dig deeper in the hole in order to accommodate another Diglett. The time taken to dig deeper is T. A hole can accommodate two Digletts at most no matter how deep it is.

You are given the time required by each Diglett to travel from its current location to reach each hole. Your job is to find the minimum amount of time it will take before at least L Digletts are hiding in holes and are safe from the construction company.

Standard input

The first line contains the number of test cases K, followed by the data of each test case.

The first line of each test case contains 4 integers: M (number of holes), N (number of Digletts), L (minimum number of Digletts that need to go to holes) and T (time required by a Diglett to dig the hole deeper in order to hold another Diglett).

The next N lines contains M integers denoting time taken for each Diglett to reach each hole.

Standard output

Output a single integer denoting the minimum amount of time it will take before at least L Digletts are hiding in holes and are safe from the construction company.

Constraints and notes

- $1 \leq K \leq 10$
- $1 \leq M \leq 100$
- $1 \leq N \leq 100$
- $1 \leq L \leq \min(N, 2M)$
- $1 \leq T \leq 200$
- $1 \leq$ Distance from Diglett to hole ≤ 100

Input	Output	Explanation
1 3 1 1 100 100 100 50	50	There is only one test case with 3 holes, 1 Diglett, 1 Diglett to save and $T = 100$ as cost to dig a hole deeper. The distances from Diglett to the three holes are 100, 100 and 50. The closest hole is 50 away. Hence, output is 50.
2 3 3 2 50 10 20 50 25 30 40 100 60 30 3 4 3 10 10 11 100 11 12 100 15 100 100 100 100 100	25 20	There are two test cases. For the first test case, there are 3 holes, 3 Diglett, 2 Diglett to save and $T = 50$ as the cost to dig a hole deeper. For the first Diglett, the distances from Diglett to the three holes are 10, 20 and 50. For the second Diglett, the distances from Diglett to three the holes are 25, 30 and 40. For the third Diglett, the distances from Diglett to the three holes are 100, 60 and 30. The first Diglett can go to the second hole and the second Diglett can go to the first hole. Hence, time required is 25. For the second test case, there are 3 holes, 4 Diglett, 3 Diglett to save and $T = 10$ as the cost to dig a hole deeper. For the first Diglett, the distances from Diglett to the three holes are 10, 11 and 100. For the second Diglett, the distances from Diglett to the three holes are 11, 12 and 100. For the third Diglett, the distances from Diglett to the three holes are 15, 100 and 100. For the fourth Diglett, the distances from Diglett to the four holes are 100, 100 and 100. The first Diglett goes to the first hole and starts digging so that the 3rd Diglett can also be accommodated. The second Diglett goes to the second hole. Hence, minimum time is 20.