Restock Diamonds

Time Limit: 2 Second
Memory Limit: 2048 MB

Hank, a jewelry reseller, wants to restock his black diamond inventory. In this city, there are n stores and m directed routes between them. Each store has a limited number of black diamonds, and every time Hank visits a store, he can purchase **at most 3 diamonds** due to security restrictions. Security guards in stores with make sure he did not purchase more than 3 diamonds, but he can **revisit any store** in his trip and purchase more.

Hank wants to maximize the number of black diamonds he can purchase by traversing the stores through their directed routes. Help Hank determine the maximum number of black diamonds he can purchase when his trip could start from any store and stop at any store.

Input

The first line contains an integer T, the number of test cases.

For each test case:

- The first line contains two integers: n (the number of stores) and m (the number of directed routes between stores).
- The second line contains n integers, representing the number of black diamonds available at each store.
- The next m lines each contain two integers: $x \quad y$, representing a directed route from store x to store y.

The constraints are:

- $1 \leq T \leq 3$: The number of test cases T is between 1 and 3.
- $2 \leq n \leq 10^5$: The number of stores n is between 2 and 10^5.
- $0 \leq m \leq n^2$: The number of directed routes m is between 0 and $\min(n^2, 2 \times 10^5)$
- The number of black diamonds at each store is between 0 and 100. Some stores do not have black diamonds.
- Stores are numbered from 0 to $n - 1$.

There could be multiple edges between stores, but there will be **no self-edges**.

Output

For each test case, output a single integer: the maximum number of black diamonds Hank can purchase while adhering to the constraints.

Sample Inputs

```
2
5 5
1 1 3 7 9
0 2
2 3
0 4
1 0
4 1
5 4
1 1 3 7 9
2 3
0 4
1 0
4 1
```

Sample Outputs

```
17
11
```