

Maximize the sum of selected numbers from an array to make it empty

给定数组arr，重复下面步骤直到数组为空：选取一个数arr[i]，求选取的arr[i]的最大和——要求在选取arr[i]之后，从数组中删除arr[i] + 1，arr[i] - 1，以及本次选取的arr[i]

首先建立map m用于记录arr中每一个数出现的次数，maxEle记录数组中最大的数

遍历i = 2 to maxEle：

对于当前遍历的i，如果本次选取i，则 $m[i] = m[i - 2] + m[i] * i$ ，如果本次不选取i，那么 $m[i] = m[i - 1]$ ，m[i]取两者较大的那个

最终m[maxEle]即为所求

```
#include <bits/stdc++.h>

using namespace std;

int maxSum(vector<int> &arr, int &N) {
    unordered_map<int, int> m;
    for (int a: arr) m[a]++;
    int maxEle = *max_element(arr.begin(), arr.end());
    for (int i = 2; i <= maxEle; ++i) {
        m[i] = max(m[i - 1], m[i - 2] + m[i] * i);
        printf("m[%d] = %d\n", i, m[i]);
    }

    return m[maxEle];
}

int main() {
    int T;
    scanf("%d", &T);
    while (T--) {
        int N;
        scanf("%d", &N);
        vector<int> arr(N);
        for (int i = 0; i < N; ++i) scanf("%d", &arr[i]);
        printf("%d\n", maxSum(arr, N));
    }
}
```