1037. Magic Coupon (25)

时间限制

100 ms

内存限制

65536 kB

代码长度限制

16000 B

判题程序

Standard

作者

CHEN, Yue

The magic shop in Mars is offering some magic coupons. Each coupon has an integer N printed on it, meaning that when you use this coupon with a product, you may get N times the value of that product back! What is more, the shop also offers some bonus product for free. However, if you apply a coupon with a positive N to this bonus product, you will have to pay the shop N times the value of the bonus product... but hey, magically, they have some coupons with negative N's!

For example, given a set of coupons {1 2 4 -1}, and a set of product values {7 6 -2 -3} (in Mars dollars M$) where a negative value corresponds to a bonus product. You can apply coupon 3 (with N being 4) to product 1 (with value M$7) to get M$28 back; coupon 2 to product 2 to get M$12 back; and coupon 4 to product 4 to get M$3 back. On the other hand, if you apply coupon 3 to product 4, you will have to pay M$12 to the shop.

Each coupon and each product may be selected at most once. Your task is to get as much money back as possible.

**Input Specification:**

Each input file contains one test case. For each case, the first line contains the number of coupons NC, followed by a line with NC coupon integers. Then the next line contains the number of products NP, followed by a line with NP product values. Here 1<= NC, NP <= 105, and it is guaranteed that all the numbers will not exceed 230.

**Output Specification:**

For each test case, simply print in a line the maximum amount of money you can get back.

**Sample Input:**

4

1 2 4 -1

4

7 6 -2 -3

**Sample Output:**

43

这道题做的时候的不当之处在于在循环中加入了复杂的STL操作，这使得整个程序的时间复杂度骤升。

由此问题想到了优化代码计划，不过这个想法可以等到PAT刷完再说（遥遥无期；

#include<iostream>

#include<string>

#include<algorithm>

#include<queue>

#include<vector>

#include<sstream>

#define ll long long

#include<stack>

using namespace std;

vector<int>poscoupon;

vector<int>posprice;

vector<int>negcoupon;

vector<int>negprice;

int comp(int a, int b)

{

return a > b;

}

int main()

{

int total=0;

int coucase, pricase;

cin >> coucase;

for (int i = 0; i < coucase; i++)

{

int temp;

scanf("%d",& temp);

if (temp > 0)

poscoupon.push\_back(temp);

else

negcoupon.push\_back(temp);

}

cin >> pricase;

for (int i = 0; i < pricase; i++)

{

int temp;

scanf("%d", &temp);

if (temp > 0)

posprice.push\_back(temp);

else

negprice.push\_back(temp);

}

sort(posprice.begin(), posprice.end(),comp);

sort(poscoupon.begin(), poscoupon.end(),comp);

sort(negprice.begin(), negprice.end());

sort(negcoupon.begin(),negcoupon.end());

for (int i = 0; i < posprice.size() && i < poscoupon.size(); i++)

{

total += posprice[i] \* poscoupon[i];

}

for (int i = 0; i < negprice.size() && i < negcoupon.size(); i++)

{

total += negprice[i] \* negcoupon[i];

}

cout << total;

}