

解题报告
参考答案
<pre> #include<bits/stdc++.h> using namespace std; const int N=100009; int n,m,x,f[N]; int main() { freopen("fuli.in","r",stdin); freopen("fuli.out","w",stdout); cin>>n>>m; for(int i=0;i<n;i++)cin>>f[i]; sort(f,f+n); for(int i=0;i<m;i++){ cin>>x; int ans=upper_bound(f,f+n,x)-f; cout<<ans<<endl; } return 0; } </pre>
题目分析
题目考察二分查找的理解和运用，需要注意的是：若不使用二分搜索，很容易超时。
提示
可以对每个账号的极客币数量进行排序，然后枚举每一种方案，利用 upper_bound 得出小于等于该方案规定数字的账户数量，输出该数量。
易错点
计算“小于等于”某值的数组元素数目，应该用 upper_bound，不是 lower_bound。

解题报告
参考答案
<pre> #include<iostream> #include<algorithm> #define N 10005 using namespace std; double h[N],a,b; int main() { int n,la,lb,ua,ub; cin>>n; for(int i=0;i<n;i++)cin>>h[i]; cin>>a>>b; la=lower_bound(h,h+n,a)-h; ua=upper_bound(h,h+n,a)-h; lb=lower_bound(h,h+n,b)-h; ub=upper_bound(h,h+n,b)-h; cout<<ub-la<<" "; cout<<lb-ua<<" "; cout<<lb-la<<" "; cout<<ub-ua<<endl; return 0; } </pre>
题目分析
<p>题目考察二分查找相关库函数的理解和运用，需要注意的是：lower_bound 和 upper_bound 的区别。</p>
提示
<p>题目是课件例题的变化，可参考课件第 11、13 页的程序。</p> <p>lower_bound 可以得到“小于”某数的位置，upper_bound 可以得到“小于等于”某数的位置。</p> <p>“大于等于”某数可以转换为“不小于”某数，“大于”某数可以转换为“不小于等于”某数。</p>
易错点
<p>lower_bound 和 upper_bound 返回的都是内存地址，并不是数组的下标位置。需要减去数组首地址，才能得到下标位置（或者前方元素的个数）。</p>

解题报告
参考答案
<pre>#include<iostream> #include<algorithm> #define N 50005 using namespace std; int x[N],y[N],m,a,b,cnt=0; int main() { cin>>m>>a>>b; for(int i=0;i<a;i++)cin>>x[i]; for(int i=0;i<b;i++)cin>>y[i]; sort(y,y+b); for(int i=0;i<a;i++){ int j=upper_bound(y,y+b,m-x[i])-y; cnt+=j; } cout<<cnt<<endl; return 0; }</pre>
题目分析
题目考察二分查找的理解和运用，需要注意的是：狙击枪和手枪至少有一种要采用二分搜索，单纯的暴力枚举会超时。
提示
可以对手枪价格进行排序，然后枚举每一种狙击枪，从预算中减去这种狙击枪的价格，利用 upper_bound 得出价格小于等于剩余预算的手枪种类数，累加到最终答案中。
易错点
1) 对某种武器排序，采用二分搜索，暴力枚举的是另一种武器。 2) 计算“小于等于”某值的数组元素数目，应该用 upper_bound，不是 lower_bound。