【算法分析】

既然要计算最少扣多少分，就眼前利益考虑必然要先把超过最后时间扣分最多的作业先安排了，如果扣分一样多的话，那必然要把时间比较紧的先安排了。所以先按扣分的高低，由高向低排序，如果两门课扣分相同就按他们的结束时间由低向高排序，然后选择即可。

参考代码如下所示。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51 | //赶作业  #include <bits/stdc++.h>  using namespace std;  #define MAX 1002  struct node  {  int deadline;  int score;  } c[MAX];  int flag[MAX];  int cmp(const node &a,const node &b)  {  if(a.score!=b.score)  return b.score<a.score;  else  return a.deadline<b.deadline;  }  int main()  {  int t,n,i,sum,j;  scanf("%d",&t);  while(t--)  {  sum=0;  memset(flag,0,sizeof(flag));  scanf("%d",&n);  for(i=0; i<n; i++)  scanf("%d",&c[i].deadline);  for(i=0; i<n; i++)  scanf("%d",&c[i].score);  sort(c,c+n,cmp);  for(i=0; i<n; i++)  {  for(j=c[i].deadline;j>0;j--) //从最后的期限开始考虑前几天有没有被安排  {  if(flag[j]==0) //如果第j天有空余  {  flag[j]=1; //标记该天被使用  break;  }  }  if(j==0) //如果一直到结束都没有空余时间，只能扣分  sum+=c[i].score;  }  printf("%d\n",sum);  }  return 0;  } |