1. 月饼

解决思路：每次只卖单价最高的月饼

1. 组合最小数

解决思路：分为第一个数和其余数

1. To Fill or Not to Fill

解题思路：

1. Magic Coupon

解题思路：

1. Sort with Swap(0,i)

解题思路：

1. Recover the Smallest Number

解题思路：

贪心算法：局部最优 -> 全局最优

区间贪心：

开区间不相交问题：每次选取左端点最大的点

#include<stdio.h>

#include<algorithm>

using namespace std;

typedef struct{

int left;

int right;

}QuJian;

bool cmp(QuJian q1, QuJian q2);

int main()

{

int n;

scanf("%d",&n);

QuJian qujian[105];

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

{

scanf("%d %d",&qujian[i].left, &qujian[i].right);

}

sort(qujian ,qujian+n,cmp);

int result = 1;

int nowLeft = qujian[0].left;

for(int i = 1; i< n;i++)

{

if(qujian[i].right <= nowLeft)

{

nowLeft = qujian[i].left;

result++;

}

}

printf("%d",result);

return 0;

}

bool cmp(QuJian q1, QuJian q2)

{

if(q1.left != q2.left) return q1.left>q2.left;

else{

return q1.right<q2.right;

}

}

闭区间选点问题：每次选取左端点最大的点

#include<stdio.h>

#include<algorithm>

using namespace std;

typedef struct{

int left;

int right;

}QuJian;

bool cmp(QuJian q1, QuJian q2);

int main()

{

int n;

scanf("%d",&n);

QuJian qujian[105];

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

{

scanf("%d %d",&qujian[i].left, &qujian[i].right);

}

sort(qujian ,qujian+n,cmp);

int result = 1;

int nowLeft = qujian[0].left;

for(int i = 1; i< n;i++)

{

if(qujian[i].right < nowLeft)

{

nowLeft = qujian[i].left;

result++;

}

}

printf("%d",result);

return 0;

}

bool cmp(QuJian q1, QuJian q2)

{

if(q1.left != q2.left) return q1.left>q2.left;

else{

return q1.right<q2.right;

}

}

1. 看电视

解题思路：区间不相交问题

1. 出租车费

解题思路：

1. 找零钱

解题思路：