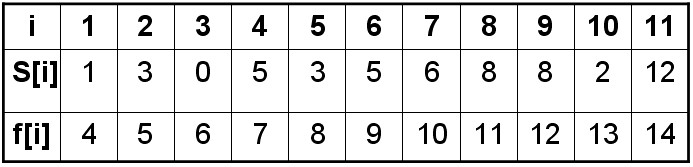
**1：活动时间安排的问题**

**设有N个活动时间集合，每个活动都要使用同一个资源，而且同一时间内只能有一个活动使用，每个活动都有一个使用活动的开始si和结束时间fi，即他的使用区间为（si,fi）,现要求你分配活动占用时间表，即哪些活动占用该会议室，哪些不占用，使得他们不冲突，要求是尽可能多的使参加的活动最大化。**

****

**上图为每个活动的开始和结束时间。**

实现代码：

#include

using namespace std;

void GreedyChoose(int len,int \*s,int \*f,bool \*flag);

int main(int argc, char\* argv[])

{

int s[11] ={1,3,0,5,3,5,6,8,8,2,12};

int f[11] ={4,5,6,7,8,9,10,11,12,13,14};

bool mark[11] = {0};

GreedyChoose(11,s,f,mark);

for(int i=0;i<11;i++)

if(mark[i])

cout<<i<<" ";

system("pause");

return 0;

}

void GreedyChoose(int len,int \*s,int \*f,bool \*flag)

{

flag[0] = true;

int j = 0;

for(int i=1;i

if(s[i] >= f[j])

{

flag[i] = true;

j = i;

}

}

**得出结果： 0 3 7 10**