题号 1010：

题目描述：

Today, President of the idle, then a person go out to play with dogs, President of walk on the road, along the way there are a lot of dogs, different dogs need different time, and the dog will only appear in a period of time (assuming of dogs learned to teleport to day, is the day after a can immediately (a). President, I must be the most dogs, then statistics and calculation, the departure time on your own, dogs, and his dog day time respectively, give a dog is not necessarily in order. Such as: President of time is 8:00 first dog go out time is 9, he need 1 hour, second time is only 9:30, day he needs half an hour, the third time is at 8:30, dog day he need 50 minutes, so most of the day is the day the third and the second only.

题目大意：

会长出门时间和路上狗的数目，会长要日最多的狗，每只狗会在特定的时间出现，每只狗只会出现一段时间，会长日完一只狗马上可以日下一只狗。问会长最多能日多少只狗。

解题思路：

会长出门的时间有了，没只狗出现的时间和存在的时间有了，可以通过排序，然后找到会长能日狗的数目，因为可能性比较多，所以可以通过不同的排序方法找到不同的日狗输入，然后通过比较，找到能日的最多的狗的数目。

解题代码：

#include<iostream>

#include<algorithm>

#include<cstring>

#include<string>

#include<cmath>

#include<cstdio>

using namespace std;

struct s

{

int x1,y1,x2,y2;

int sum1,sum2,sum3;

}str[1010];

bool cmp1(const s &a, const s &b)

{

if (a.sum1!=b.sum1)

return a.sum1 < b.sum1;

else

return a.sum2<b.sum2;

}

bool cmp2(const s &a, const s &b)

{

if (a.sum2!=b.sum2)

return a.sum2 < b.sum2;

else

return a.sum1<b.sum1;

}

bool cmp3(const s &a, const s &b)

{

if (a.sum3!=b.sum3)

return a.sum3 < b.sum3;

else

return a.sum2<b.sum2;

}

int main()

{

int n;

int m1,m2,sum;

while (scanf("%d:%d %d",&m1,&m2,&n)!=EOF)

{

int shu1=0;

int shu2=0;

int shu3=0;

sum=m1\*60+m2;

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

{

scanf("%d:%d %d:%d",&str[i].x1,&str[i].y1,&str[i].x2,&str[i].y2);

str[i].sum1=str[i].x1\*60+str[i].y1;

str[i].sum2=str[i].x2\*60+str[i].y2;

str[i].sum3=str[i].sum1+str[i].sum2;

}

sort(str,str+n,cmp1);

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

{

if (str[i].sum1>=sum)

{

shu1++;

sum=str[i].sum1+str[i].sum2;

}

}

sort(str,str+n,cmp2);

sum=m1\*60+m2;

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

{

if (str[i].sum1>=sum)

{

shu2++;

sum=str[i].sum1+str[i].sum2;

}

}

sort(str,str+n,cmp3);

sum=m1\*60+m2;

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

{

if (str[i].sum1>=sum)

{

shu3++;

sum=str[i].sum1+str[i].sum2;

}

}

if (shu1>=shu2&&shu1>=shu3)

cout<<shu1<<endl;

else if (shu2>=shu1&&shu2>=shu3)

cout<<shu2<<endl;

else if (shu3>=shu1&&shu3>=shu1)

cout<<shu3<<endl;

}

return 0;

}