

西安科技大学

**操作系统**

**《进程同步与线程同步》实验报告**

**学院：** **计算机科学与技术学院**

**专业：** **网络工程系**    
 **班级：** **网络工程1801班**

**姓名：** **吴斌**

**学号：** **18408020129**

**日期：**  **2020年11月**

**进程同步与线程同步**

一、实验目的说明：

通过编程实现两个进程之间以及两个线程之间的同步。体会进程同步和线程同步时的执行过程。

二、实验内容和方法：

下为a.c：

#include<stdio.h>

#include<windows.h>//引入必要的头文件

int main(void){

HANDLE h;

h = CreateSemaphore(NULL,0,10,"ScanfSuccess");

while(1)

{

char x;

scanf("%c",&x);//获取键盘输入的字符串

ReleaseSemaphore(h,1,NULL);

}

return 0;

}

下为b.c

#include<stdio.h>

#include<windows.h>//引入必要的头文件

int main(void){

HANDLE h;

int count = 0;

h = CreateSemaphore(NULL,0,10,"ScanfSuccess");

while(1)

{

WaitForSingleObject(h,INFINITE);

count = count++;

printf("%d",count);

//循环接受a.c里输入的字符，并体现在count上并打印出来

}

return 0;

}

上述a.c和b.c构成进程同步

下为c.c（线程同步）

#include<stdio.h>

#include<windows.h>

HANDLE m;

DWORD WINAPI ThreadFunc1(LPVOID\_lpParam){

int i;

for(i = 0;i < 5; i++)

{

WaitForSingleObject(m,INFINITE);

printf("thread1 step1 int the critical section\n");

printf("thread1 step2 int the critical section\n");

ReleaseMutex(m);

}

return 0;

}

DWORD WINAPI ThreadFunc2(LPVOID\_lpParam){

int i;

for(i = 0;i < 5; i++)

{

WaitForSingleObject(m,INFINITE);

printf("thread2 step1 int the critical section\n");

printf("thread2 step2 int the critical section\n");

ReleaseMutex(m);

}

return 0;

}

int main(){

char c;

m = CreateMutex(NULL,FALSE,NULL);

CreateThread(NULL,0,ThreadFunc1,0,0,0);

CreateThread(NULL,0,ThreadFunc2,0,0,0);

scanf("%c\n",&c);

}

三、实验结果：



