1. 实验题目

实验6-2-3处理器算法调度模拟

设计一个按先来先服务的调度算法实现处理机调度

1. 实验内容与要求

假设系统中有3-5个进程，每个进程由一个进程控制块来标识。

进程名：即进程标识。

链接指针：按照进程到达系统的时间将处于就绪状态的进程连接成一个就绪队列。指针指出下一个到达的进程控制块首地址。最后一I个进程的链指针为NULL。

估计运行时间：任意指定一个值。

到达时间：进程创建时的系统时间或由用户指定，调度时，总是选择到达时间最早的进程。

进程状态：创建时就绪态‘R’，运行结束后置入完成态’C’。

设置一个队首指针head，用来指出最先进入系统的进程，各就绪进程通过链接指针连在一起。处理机调度时总是选择队首指针指向的进程投入运行。运行时，仅模拟执行（估计运行时间减一）的操作。

在所设计的程序中应有显示或打印语句，能显示正运行的进程名字。已执行时间。还剩时间。就绪队列中的进程名字。所有进程完成后，给出各进程的周转时间和平均周转时间。

1. 编程语言与编程工具

编程语言：c++

编译工具：visual studio2015

1. 程序设计

逻辑结构：单链表

存储结构：

Class process{

string name ;

char status = 'R';

int arr\_time;

int run\_time;

process\* next;

process（int n， int m，string name）;

void print\_run();

process\* manage(process &a, process &b, process &c);

};

Process类就是pcb进程控制块的类型，本题中我只考虑了3个进程，manage函数是用来对3个进程进行调度的函数，并返回最早执行的进程地址。Print\_run函数是用来显示各个进程运行的显示及模拟执行函数。

算法设计1：

1: 比较a和b的到达时间，

若a>b, 则b->next = &a;

2-1：比较b和c的到达时间，

若b>c,则c->next = &b，并返回&c

若b<c,

3-1：则比较a和c的到达时间

若a>c,则b-next = &c，c->next = &a，否则a->next= &c

返回&b

若a<b，则a->next = &b，

2-2：比较a和c的到达时间，

若a>c，则c->next= &a，并返回&c

若a<c,

3-2: 则比较b和c的到达时间

若b>c，则a-next = &c，c->next = &b，否则b->next = &c

返回&a

算法思路2：

建一个数组arr【n】，把a，b，c的到达时间放入这个数组，并进行从小到大的冒泡排序。再依次读出排好序的数据，并且将其的next指针指向其下一个元素。这样可以实现3个以上的进程调度，对到达时间的排序。但是我写完了方法1才想到这个方法

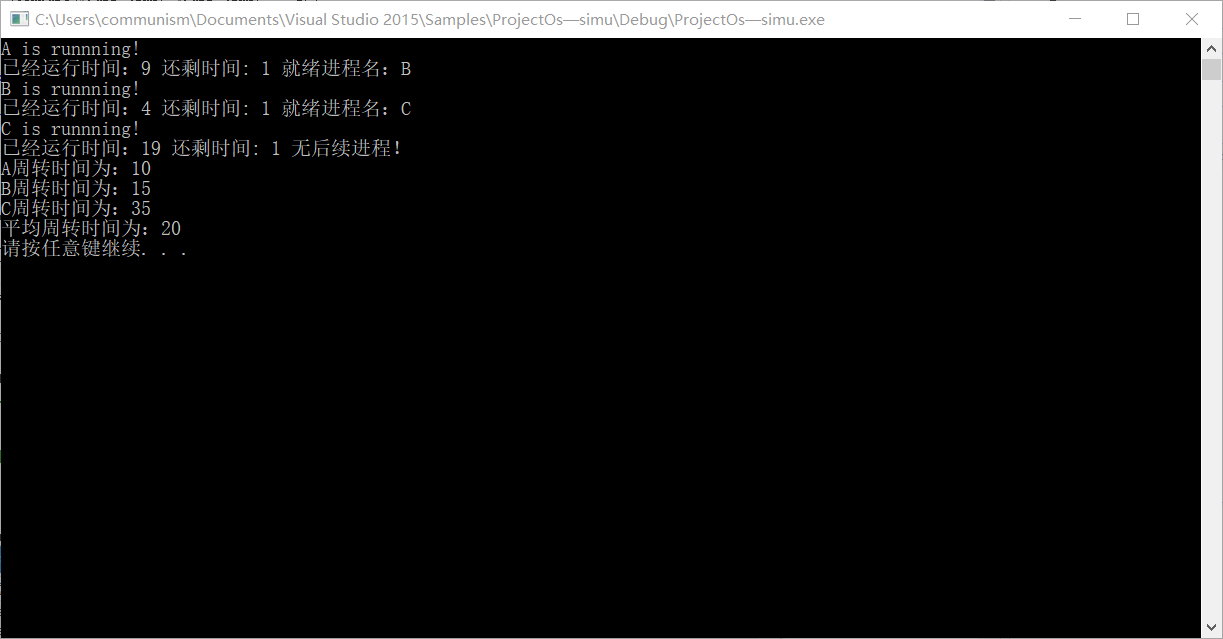
此处采用的方法1，没有写算法2

1. 程序运行

测试数据1：

process a(1,10,"A"), b(2,5,"B"), c(3,20,"C");

测试结果：符合预期 ——a-b-c顺序

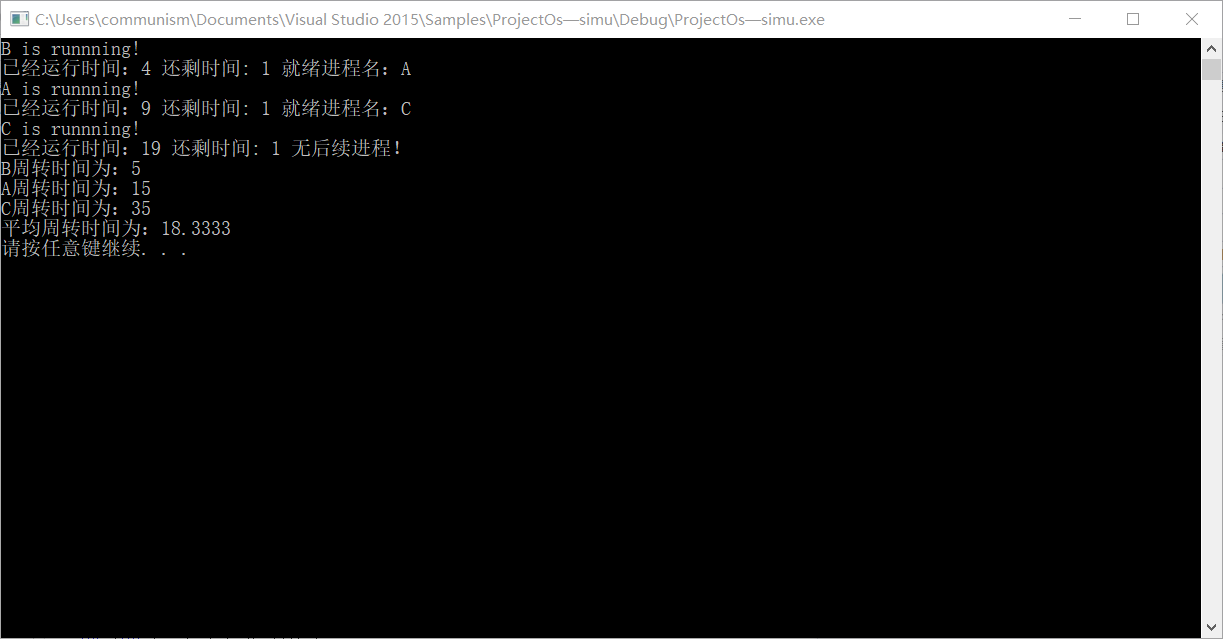


测试数据2：

process a(2,10,"A"), b(1,5,"B"), c(3,20,"C");

测试结果：符合预期

b-a-c顺序



测试数据3：

process a(2,10,"A"), b(3,5,"B"), c(1,20,"C");

测试结果：符合预期

c-a-b顺序

