

Ch9

班级

01

学号

2022040906023

姓名

梁书恺

成绩

一、选择题（每空 5 分，共 30 分）

1. 在多任务系统中，为了保证公共变量的完整性，各进程应互斥进入临界区。临界区是指（ D ）。
A、一个缓冲区 B. 一段数据区 C、同步机制 D、一段程序
2. （ C ）不是实时系统的基本特点。
A、高可靠 B、保障及时 C、良好交互 D、多路控制
3. （ D ）不是操作系统的主要功能。
A、提供用户与计算机硬件之间的接口
B、有效地控制和管理计算机的软硬件资源
C、合理地调度计算机的工作流程，改善计算机系统的性能
D、为软件开发提供可视化集成开发环境。
4. 下面所述步骤中，（ A ）不是创建进程所必需的。
A、为进程分配 CPU B、建立一个进程控制块
C、为进程分配内存 D、将进程控制块插入就绪队列
5. 当（ B ）时，进程从执行状态转变为就绪状态。
A、进程被调度程序选中 B、时间片 C、等待某一事件 D、等待的事件发生
6. 下列有关进程状态转化说法错误的是（ D ）
A 从就绪状态可以直接转换到执行状态
B 从执行状态可以直接转换到阻塞状态
C 从执行状态可以直接转换到就绪状态
D 从就绪状态可以直接转换到阻塞状态

二、简答题（共 8*7=56 分）

1. 什么是操作系统？操作系统的作用是什么？有哪五大功能？

操作系统是计算机系统中直接控制和管理各种软硬件资源，以方便用户充分而有效地利用这些资源的程序的集合。

操作系统的作用：

1. 提供用户与计算机之间的接口
2. 有效管理计算机的软、硬件资源
3. 合理地调度计算机的工作流程，提高系统效率

操作系统的功能：

1. 处理机管理
2. 存储器管理
3. 设备管理
4. 文件管理
5. 作业管理（用户接口）

2. 什么是多道程序设计技术？操作系统中引入多道程序设计技术带来哪些好处？

多道程序设计技术是指在内存中同时存放若干个作业，并让它们同时运行，且共享系统中的资源的技术。

1. 提高 CPU 的利用率；
2. 提高内存和 I/O 设备的利用率；
3. 增加系统吞吐量。

3. 为什么要引入进程？PCB 的作用是什么？进程与程序有什么区别？

程序的概念已无法描述动态执行过程中的并发活动，引入进程来描述程序的一次执行，使并发执行的程序保持“可再现性”。

PCB 可唯一标识一个进程，其中的信息为进程的控制提供依据，将程序变成了进程，是进程在系统中存在的唯一标志。

1. 程序是指令的集合，是静态概念；进程是程序的执行过程，是动态概念。
2. 程序可作为软件资源长期保存；进程只是一次短暂活动或过程。
3. 一个程序可对应多个进程；一个进程可包含多段程序

4. 什么是响应时间？影响响应时间的因素有哪些？时间片过大或过小有什么问题？

响应时间指的是用户从发出请求到接收完响应之间的总耗时。

影响因素有时间片的大小和就绪队列进程数。

时间片过小，时间片轮转频繁，消耗运算资源；时间片过大，响应时间延长，实时性减弱。

5. 进程有哪些基本状态？状态转换的原因和过程是什么？

进程有就绪、执行、阻塞三种状态。

就绪→执行：进程调度；

执行→就绪：中断或时间片用完；

执行→阻塞：I/O 请求或等待某事件；

阻塞→就绪：I/O 完成或事件发生。

6. 什么是重定位？静态重定位与动态重定位有何区别？试比较它们的优缺点。

重定位指操作系统将逻辑地址转变为物理地址的过程。

静态重定位装入：装入时，由系统分配内存空间，并完成装入模块中相对地址变换为绝对地址。

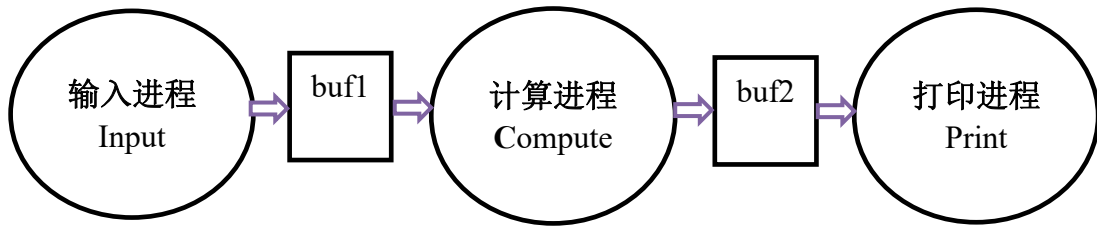
动态重定位装入：装入模块中使用相对地址，装入时不变换地址，在执行时才完成地址的变换。

7. 操作系统通过哪些途径提高内存利用率？

1. 允许多道程序动态共享主存，提高内存空间的利用率
2. 引入虚拟存储

三、算法设计（共 14 分）

用 P、V 操作及信号量写出下图三个进程的同步算法，其中 buf1 与 buf2 均应互斥访问。
(设 buf1 和 buf2 的大小分别是 n1 和 n2)



```
Input()
{
    P(buf1_empty);
    P(buf1_mutex);
    输入 buf1;
    V(buf1_mutex);
    V(buf1_full);
}
```

```
Compute()
{
    P(buf1_full);
    P(buf1_mutex);
    计算并取出 buf1;
    V(buf1_mutex);
    V(buf1_empty);

    P(buf2_empty);
    P(buf2_mutex);
    输入 buf2;
    V(buf2_mutex);
    V(buf2_full);
}
```

```
Print()
{
    P(buf2_full);
    P(buf2_mutex);
    打印并取出 buf2;
    V(buf2_mutex);
    V(buf2_empty);
}
```