

2018-CS240-Autumn 卷学渣解析:

(答案可能有误, 欢迎各位学霸们指正!!!)

### 1. State five main tasks that an operating system performs.

答案来源: Lecture 01: P20-P22

- ① Allocation and Management of Resources among competing user processes  
(在相互竞争的用户流程之间的资源分配和管理)
- ② Maximising Resource Utilisation with the goal of improving overall system throughput  
(最大化资源利用率, 以提高整体系统吞吐量)
- ③ Providing a user interface and an application interface to the machine  
(为机器提供用户界面和应用程序界面)
- ④ Coordinating the many concurrent activities and devices, handling input and output from attached hardware and ensuring correct synchronisation and communication is achieved  
(协调许多并发活动和设备, 处理来自附加硬件的输入和输出, 并确保实现正确的同步和通信)
- ⑤ Acting as a Resource Guardian to protect various resources of the computer system from malicious or accidental misuse  
(作为资源监护者, 保护计算机系统的各种资源免受恶意或意外滥用)
- ⑥ Accounting for periods of resource usage by user processes, enforcing quotas or restrictions as appropriate  
(根据用户进程计算资源使用期间, 适当地执行配额或限制)
- ⑦ Power and Thermal Management  
(电力和热管理)

### 2. Explain the scheduling requirements of a real-time system and how they differ from a general purpose system.

Discuss ways in which a system could schedule real-time tasks effectively.

答案来源: Lecture 05: P22-P23

Explanation:

Real-time systems can usually be divided into hard real time systems and soft real time systems.

(实时系统通常可以分为硬实时系统和软实时系统)

Hard Real Time Systems are required to complete a critical task within a guaranteed amount of time. The execution time of code needs to be analysed before a task may be accepted by the scheduler.

(硬实时系统需要在有保证的时间内完成一个关键任务。在调度器可以接受任务之前, 需要分析代码的执行时间)

Soft Real Time Systems are ones which endeavour to meet scheduling deadlines but where missing an occasional deadline may be tolerable.

(软实时系统是那些努力满足最后期限, 但错过偶尔的最后期限是可以容忍的)

答案来源: CSDN

Difference:

- ① Set priority (设置优先级)

The general purpose system does not always perform strictly following the priority set by the program

(通用系统不会始终严格按照程序设置的优先级执行)

The real-time system can strictly follow the priority program set by the programmer

（实时系统可严格按照程序员设置的优先级执行程序）

## ② Interrupt Latency（中断延迟）

The time required for general purpose system response interruption is not fixed and real time system must ensure response interruption at a fixed time

通用系统响应中断需要的时间不固定，实时系统必须确保在一个固定的时间内响应中断

## ③ temporal characteristics（时间特性）

Real time system have more accurate and reliable temporal characteristics

（实时系统具有更精确和更可靠的时间特性）

出处原图：

### 3. 探究事实：实时操作系统与通用操作系统有哪些不同之处

Windows、Mac OS等操作系统是开发和运行非实时测量和控制应用程序的优秀平台。这些操作系统使用于实时操作系统之外的使用场合。对于要求精确定时和长时间稳定运行的系统而言，通用操作系统不是理想的平台。下面将阐述两种操作系统之间的区别，以及编写实时应用程序时的注意事项。

#### 设置优先级

编写应用程序时，大多数操作系统都允许程序员指定应用程序的总体优先级，以及应用程序中不同任务（线程）的优先级。这些优先级设置将程序和任务的重要性告知操作系统。如两个任务同时运行，操作系统将优先运行优先级设置较高的任务。

在实际应用中，通用操作系统不会始终严格按照程序设置的优先级执行。因为通用操作系统可同时运行多个应用程序和进程，所有任务都会被分配到一些处理时间。在某些情况下，低优先级任务的临时优先级可能会比高优先级任务更高。这样，每个任务都会分配到一定的运行时间。这会违背程序设计人员的设计初衷。

实时操作系统可严格按照程序员设置的优先级执行程序。在多数实时操作系统上，如果一个高优先级任务占用率100%的处理器资源，低优先级任务将一直等待直到高优先级任务完成。因此，设计实时应用程序时，必须谨慎、合理设置优先级。在一个典型的实时应用程序中，设计者应该将实时代码放置在高优先级的部分。写入磁盘、网络通信等较低优先级的代码应该放在较低优先级的部分。

#### 中断延迟

设备生成一个中断和设备中断发生之间的时间，称为中断延迟。通用操作系统响应中断需要的时间不固定，实时操作系统必须确保在一个固定的时间内响应中断。也就是说，实时操作系统的延迟必须是已知的固定的值。

#### 性能

一个常见的误解是实时操作系统的性能高于通用操作系统。在某些情况下，因为不需要在多个应用程序和服务之间进行多任务操作，实时操作系统的性能更好。这并不是绝对的规律。实际应用程序的性能取决于CPU主频、内存架构、程序特征，等等。

即使实时操作系统对执行速度没有直接影响，但是它与通用操作系统相比具有更精确和更可靠的时间特性。

答案来源：Lecture 05: P25-P26

### ① Using Earliest Deadline First:

When an event is detected, the handling process is added to the ready queue. The list is kept sorted by deadline, which for a periodic event is the next time of occurrence of the event. The scheduler services processes from the front of the sorted queue serving the most urgent tasks first

(当检测到一个事件时，该处理过程将被添加到就绪的队列中。列表按截止日期排序，对于周期事件，截止日期是事件下一次发生的时间。调度器服务从已排序队列的前面进行处理，首先服务于最紧急的任务)

### ② Using Least Laxity Algorithm:

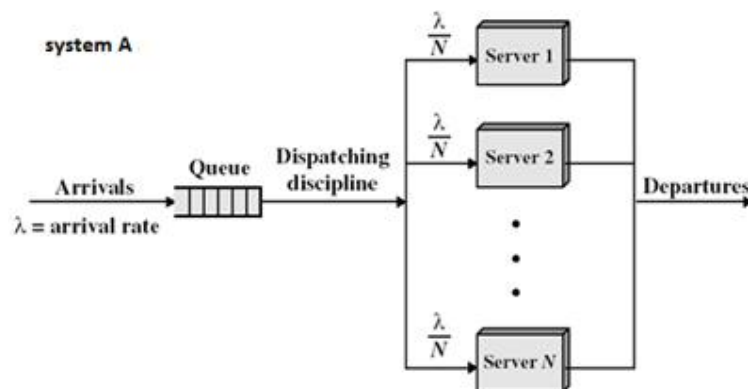
If a process requires 200msec and must finish in 250msec, then its laxity is 50msec. The Least Laxity Algorithm chooses the process with the smallest amount of time to spare. This algorithm might work better in situations where events occurred aperiodically.

（如果一个过程需要 200 毫秒，并且必须在 250 毫秒内完成，那么它的松弛度是 50 毫秒。最小松弛度算法选择空闲时间最小的过程。该算法可能在事件不定期发生的情况下工作得更好）

### 3. Give an argument as to why a single queue may perform better than multiple queues in a multiprocessor system.

A multiprocessor system can be composed of a heterogeneous or homogenous collection of processors. With homogeneous processors, a single queue of tasks may be serviced by all processors. Using a single queue is now more efficient than using multiple queues.

（多处理器系统可以由异构或同构的处理器集合组成。对于同构处理器，所有处理器都可以为单个任务队列提供服务。此时，使用单个队列比使用多个队列更有效）



### 4. In disk scheduling, explain what advantages the C-LOOK algorithm has over the SCAN algorithm? Support your answer with an example.

答案来源：CSDN

Explanation:

Because every head in SCAN goes to the end of the track, and the actual process does not require the head to go to the end, but reaches the last request in that direction, using C-LOOK can avoid some unnecessary head movement.

（因为 SCAN 中每次磁头都要走到磁道尽头，而实际过程中并不需要要求磁头走到尽头，而是到达该方向的最后一个请求后即可返回，使用 C-LOOK 可以避免一些不必要的磁头移动）

出处原图：

## 五.LOOK和C-LOOK

### 1.方法

LOOK和C-LOOK就是对应SCAN和C-SCAN的改进算法。因为SCAN和C-SCAN中每次磁头都要走到磁道尽头，而实际过程中并不需要要求磁头走到尽头，而是到达该方向的最后一个请求后即可返回，这样可以避免一些不必要的磁头移动。

### 2.优点

避免一些不必要的磁头移动

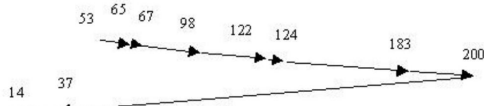
答案来源：Lecture 06: P18、P21

Example:

98, 183, 37, 122, 14, 124, 65, 67

#### SCAN Algorithm

The head moves in one direction, either to the center or to the outermost track, servicing requests as it passes the corresponding tracks. It then turns around and repeats this in the other direction. With this algorithm, assuming the head is moving in the direction of increasing track numbers and track numbers are from 0..200, we would get the following head movements:-

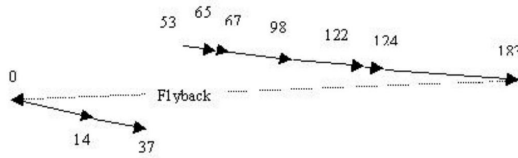


98, 183, 37, 122, 14, 124, 65, 67

#### C-LOOK Algorithm

The C-LOOK algorithm is the same as C-SCAN except that where C-SCAN always continues to the last track in a particular direction before flying back, the C-LOOK algorithm simply checks to see if there are anymore requests in that direction before flying back.

With this algorithm we get the following head movements:-



5. Compare and contrast the merits of socket based communication in Java with the Remote Method Invocation (RMI) mechanism.

答案来源: Zoukankan.com

① Compared to RMI, socket's network programming is more flexible and controllable

与 RMI 相比, socket 的网络编程更具灵活性和可控性

② In transmission data, RMI transmits more data. Thus, socket network programming is more suitable for big data transmission

在传输数据中, RMI 会额外传输更多的数据。由此可见, socket 的网络编程更适用于大数据传输

出处原图:

#### 三、RMI Vs Socket

RMI技术比较socket的网络编程主要有以下几个方面:

第一、.RMI是面向对象的,而后者不是。

第二、.RMI是与语言相绑定的。比如当你使用Java RMI技术的时候,客户端与服务端都必须使用Java开发。而socket的网络编程是使用独立于开发语言的,甚至独立于平台。基于socket的网络编程,客户端与服务端可以使用不同开发语言和不同的平台。

第三、从网络协议栈的观点来看,RMI与socket的网络编程处于不同层次上。基于socket的网络编程位于TCP协议之上,而RMI在TCP协议之上,又定义了自己的应用协议,其传输层采用的是Java远程方法协议(JRMP)。可见,在网络协议栈上,基于RMI的应用位置更高一些,这也决定了,与socket的网络编程相比,RMI会丧失一些灵活性和可控性,但是好处是它带给了应用开发者更多的简洁、方便和易用。比如:如果你用的是RMI,你不需要关心消息是怎么序列化的,你只需要像本地方法调用一样,使用RMI。代价是:应用开发者无法很好地控制消息的序列化机制。

第四、这是最后一点不同,我认为也是比较重要的一点,就是两种方法的性能比较,其往往决定着你将使用那种技术来开发你的应用。以下引用Adrian Reber在Network-programming with RMI文中对TCP和RMI所做的一个比较,其做的实验主要是对两者在网络传输的带宽上作的对比: 在网络上传输2 byte的有效数据,对于TCP而言,总共有478 byte被额外传输,而对于RMI,1645byte被额外传输。

6. Give a psuedo code software solution to the n-process mutual exclusion problem indicating the entry code and exit code to be executed by each process. Explain the components of your code.

同 19 年 Autumn 卷, 此处不赘述。

7. The first readers/writers concurrency problem prioritises readers and requires that no reader be kept waiting unless a writer has already obtained permission to use the shared item. Define a psuedo code solution to this coordination problem using semaphores.

答案出处: Lecture 15&16

```
public class DataAccessPolicyManager
{
    private int readerCount;
    private Semaphore mutex;
    private Semaphore wrt;

    public DataAccessPolicyManager () {
        readerCount = 0;
        mutex = new Semaphore(1);
        wrt = new Semaphore(1);
    }

    public void acquireReadLock() {
        mutex.acquire();
        ++readerCount;
        if (readerCount == 1) // first reader
            wrt.acquire();
        mutex.release();
    }

    public void releaseReadLock() {
        mutex.acquire();
        --readerCount;
        if (readerCount == 0) // Last reader
            wrt.release();
        mutex.release();
    }

    public void acquireWriteLock() {
        wrt.acquire();
    }

    public void releaseWriteLock() {
        wrt.release();
    }
}
```

8. Describe the operation of a paged memory architecture using diagrams to support your answer. Explain clearly

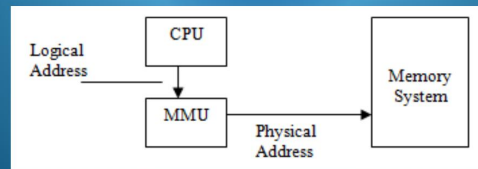
the process of memory translation and the benefits of paged memory.

答案出处: Lecture 19&20: P22、P24、P25

### Paged Architecture supports logical to physical mapping

The process has a **logical view of its memory space** as consisting of a number of **logical page numbers** from 0 to a maximum.

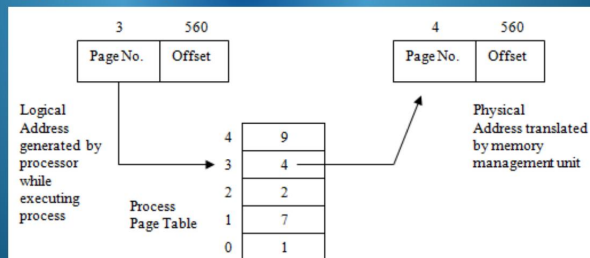
These pages are **mapped by the memory management unit** of the processor to the corresponding **physical pages** in the memory.



### Paged Architecture

A memory address is divided into two parts, a **page number** and an **offset** within that page. The **MMU exchanges** the page number from the logical address generated by the processor, with the corresponding physical page found in the page table.

Say pages are 1000 bytes in size



### Paged Architecture - Advantages

**Protection** - As processes only have a logical view of their address spaces, it is not possible for them to access the address space of other processes because the page table won't contain pointers to pages for any other process.

The memory space assigned to a process can be composed of pages which are scattered randomly throughout the physical memory and **do not all have to be adjacent** to one another. Easy to keep track of free space and to allocate space of requested size.

It is easy to **relocate or expand the address space** of a process by altering the mapping of pages into its page table.

9. In the context of a virtual memory system what is meant by the terms “working set” and “page replacement algorithm”? Briefly discuss two practical page replacement algorithms.

答案出处: Lecture 19-20: P33、P34

What:



### Working Set

Each process must be allocated a number of physical memory pages in which the memory manager attempts to store pages from its logical address space corresponding to its current locality of reference. The **set of physical pages allocated to a process is known as its working set**.

### Page Replacement Algorithms

Sometimes, there may be **no free pages left** in memory to accommodate an incoming page, or perhaps the operating system **restricts the size of the working set** for a process, in order to be fair to other processes in the allocation of physical memory space.

答案出处: Lecture 19-20: P37、P39

Discuss:

### Page Replacement Algorithms

The **Optimal Page Replacement** algorithm, which guarantees a minimum number of page faults is to replace the page which won't be used for the longest time.

### Page Replacement Algorithms

The **First In First Out** Algorithm replaces the page that has been in the memory the longest.