

作业9 调度理论 参考答案

同济大学计算机系操作系统 国豪2023 拔尖班2023

姓名

学号

一 ~ 三 略 已有参考答案

四、单项选择题

1. 下面对进程的描述中，错误的是_____。
- A. 进程是动态的概念 B. 进程执行需要处理机
C. 进程是有生命期的 D. 进程是指令的集合（有数据的，还有进程的执行状态）

进程 = 代码 + 数据 + 进程控制块 (PCB)

Unix V6++的进程控制块 = Process结构 + User结构

2. 多道程序环境下，操作系统分配资源以_____为基本单位。

A. 程序 B. 指令 C. 进程 D. 作业

3. 操作系统通过_____对进程进行管理。

A. 进程 B. 进程控制块 C. 进程启动程序 D. 进程控制区

4. 系统感知进程的唯一实体（进程存在的唯一标志）是_____。

A. JCB B. FCB C. PCB D. SJT

5. 在进程管理中，当_____时，进程从阻塞状态变为就绪状态。

A. 进程被进程调度程序选中 B. 进程开始等待某一事件
C. 进程等待的事件发生 D. 进程的时间片用完

6. 分配到必要的资源并获得处理机时的状态是_____。

A. 就绪状态 B. 执行状态 C. 阻塞状态 D. 撤销状态

7. 进程的三个基本状态之间在一定条件下是可以相互转化的，进程由就绪状态变为运行状态的条件是：_____① D_____；由运行状态变为阻塞状态的条件是_____② B_____。

A. 时间片用完 B. 需要等待某事件发生
C. 等待的某事件已发生 D. 被进程调度程序选中

8. 下列的进程状态变化中，_____变化是不可能发生的，

A. 运行→就绪 B. 运行→等待 C. 等待→运行 D. 等待→就绪

怪我，没花时间修这份题目。这份习题，阻塞就是睡眠就是等待。

这3个术语，等待表述得最为不清晰。因为未能强调进程因等待资源丧失使用CPU运行的权力。

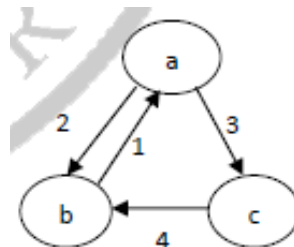
9. 一个运行中的进程用完了分配给它的时间片后，它的状态变为_____。

A. 就绪 B. 等待 C. 运行 D. 不确定

10. 某系统的进程状态如下图所示；a 是_____① B_____状态，b 是_____② D_____状态，c 是_____③ E_____状态。1 表示_____④ B_____，2 表示_____⑤ A_____，3 表示发生了等待事件，4 表示等待事件结束。

① ②③： A. 挂起 B. 运行 C. 等待 D. 就绪 E. 睡眠

④⑤： A. 落选(被剥夺) B. 选中 C. 等待



11. 下面所述步骤中, _____不是创建进程所必需的。

- A. 由调度程序为进程分配CPU
B. 建立一个进程控制块
C. 为进程分配内存
D. 将进程控制块链入就绪队列

12. 设某系统中有 n ($n > 2$) 个进程，且当前不在执行进程调度程序，上述情况中，不可能发生的情况是：_____

- A. 没有运行进程，有2个就绪进程，~~n~~个进程处于等待状态；
B. 有1个运行进程，没有就绪进程，n-1进程处于等待状态；
C. 有1个运行进程，有1个就绪进程，n-2进程处于等待状态；
D. 有1个运行进程，n-1个就绪进程，没有进程处于等待状态。

13. 进程调度是从 选择一个进程投入运行。

- A. 就绪队列 B. 等待队列 C. 作业后备队列 D. 提交队列

14. 一个进程在某一时刻具有_____。

- A. 一种状态 B. 两种状态 C. 三种状态 D. 四种状态

15. “可抢占”和“不可抢占”的优先级调度算法相比。

- A. 前者开销小 B. 前者开销大
C. 两者开销大致相同 D. 两者开销不能相比

解释：因为“可抢占”优先级调度始终保证在处理机上运行的是优先级最高的进程，这样，当处理机正在运行某个进程时，很可能被其他优先级更高的进程“抢占”引起处理机调度，和“不可抢占”算法相比，前者的调度次数会更频繁，而每调度一次都会引起保护现场、恢复现场的工作，并且会弄脏 CPU cache 和 快表，降低 CPU 执行进程的速度，（存储管理理论会说）所以“可抢占”的优先级调度算法开销更大。

16. 优先权是在创建进程时确定的，确定之后在整个进程运行期间不再改变。

- A. 先来先服务 B. 静态 C. 动态 D. 短作业

静态优先权反映作业的特征，前台、后台，是 CPU bound 还是 IO bound

17. 各种作业调度算法中, 若所有作业同时到达, 则平均等待时间最短的算法是

○

- A. 先来先服务
作业优先

18. 是指从作业提交给系统到作业完成的时间间隔。

- A. 周转时间 B. 响应时间 C. 等待时间 D. 运行时间

19. 作业从进入后备队列到被调度程序选中的时间间隔称为_____。

- ~~A. 周转时间~~ ~~B. 响应时间~~ ~~C. 等待时间 (就这个, 不纠结)~~ ~~D. 触发时间~~

20. 假设下述四个作业同时到达, 当使用最高优先数优先调度算法时, 作业的平均周转时间为 _____ 小时。

作业	所需运行时间	优先数
1	2	
2	5	
3	8	
4	3	

- A. 4.5 B. 10.5 C. 4.75 D. 10.25

解释：作业执行次序 2, 4, 1, 3

运行时间 5, 3, 2, 8

周转时间	5, 8, 10, 18
------	--------------

(周转时间=等待+运行, 等待时间=Σ序列中排在前面的所有运行时间)

周转时间求均值。不怕烦就不累, 不会错。

21. 设有4个作业同时到达, 每个作业的执行时间均为2小时, 它们在一台处理机上按单道方式运行, 则平均周转时间为_____。

- A. 1小时 B. 5小时 C. 2.5小时 D. 8小时

解释: 作业执行次序 * (2), * (2), * (2), * (2) // 这是作业的任意排列

周转时间依次是 2 4 6 8

求均值

22. 现有3个同时到达的作业J1、J2和J3, 它们的执行时间分别是T1、T2和T3, 且 $T_1 < T_2 < T_3$ 。系统按单道方式运行且采用短作业优先算法, 则平均周转时间是_____。

- A. $T_1 + T_2 + T_3$ B. $(T_1 + T_2 + T_3) / 3$

- C. $(3T_1 + 2T_2 + T_3) / 3$ D. $(T_1 + 2T_2 + 3T_3) / 3$

作业执行序列是J1、J2和J3

23. 作业周转时间为_____。

- A. 作业开始时间 - 作业提交时间

- B. 作业等待时间 + 作业执行时间

- C. 作业等待时间

- D. 作业执行时间

或, 完成时刻 - 提交时刻

二、填空题

1. 程序顺序执行时有顺序性、封闭性和可再现性的特点。~~(程序顺序执行是指程序独占计算机资源运行的情况)~~

程序顺序执行时独占计算机, 没有其它进程与其竞争资源。

不会走走停停, 顺序性;

独占计算机, 指运行在封闭运行环境;

可再现, 因为无论何时执行, 运行环境恒定不变。

2. 进程是一个动态①态概念, 而程序是一个静态②态概念。

3. 在一个单处理机系统中, 若有5个进程, 则处于就绪状态的用户进程最多有① 4个, 最少有② 0个。

4. 一个就绪进程都没有的状态叫做CPU空转。此时刻, 现运行进程是0#进程。

5. 进程的调度方式有两种, 一种是可剥夺的①, 另一种是不可剥夺的②。

6. 一种最常用的进程调度算法是把处理机分配给具有最高优先权的进程。而确定优先权的方法概括起来不外乎是基于静态①特性和动态②特性两种方法。前者所得到的

静态③优先权, 后者所得到的动态④优先权。

7. 设有一组作业, 它们的提交时间及运行时间如下:

作业号 提交时间 运行时间 (分钟)

1 9:00 70

2 9:40 30

3 9:50 10

4 10:10 5

在单道方式下，采用短作业优先调度算法，作业的执行顺序是 1、4、3、2。

9点钟，只有作业1，运行它。

8、设有3个作业，A，B，C。其运行时间分别为1，2，3小时。假定这些作业同时到达，并在一台处理器上单道运行，则平均周转时间最小的执行序列是 A，B，C。

解释：短作业优先调度理论最优，我们有证明。

9、导致CPU切换现运行进程的事件包括 现运行进程时间片用完，或者有高优先级进程就绪（典型地，系统唤醒了一个高优先级的进程），现运行进程入睡，现运行进程终止。

10、多道系统，资源分配的单位是 进程。（可选项：进程，作业，线程）

四、综合题

（一）情景分析，基于多级反馈队列的调度系统。

1、上班时间，n个IO密集型的交互式作业和1个CPU密集型的长作业同时进入系统。分析所有作业的运行情况。

2、凌晨3点（没有人上机与系统交互了）。。。系统中的作业和它们的运行情况。

3、某个进程，从IO密集阶段进入CPU密集阶段。。之后，回到IO密集阶段。

4、第二天早上，上班时间到。

参考答案：

1、上班时间，n个IO密集型的交互式作业（队列1管理。始终用不完时间片）和1个CPU密集型的长作业并行。分析所有作业的运行情况。

答：n+1个作业开始运行时，全部位于队列1。CPU密集型长作业因用满时间片降级至队列2，并最终降级至最低优先级队列。n个IO密集型的交互式作业，若就绪则位于队列1；然而，这些进程的属性决定了它们几乎总是处在阻塞状态。故，系统的运行情况是，交互式作业几乎总是在等待IO操作完成或是在等待使用IO（外部）设备。交互式作业阻塞时，CPU密集型长作业使用CPU运行。IO完成时，IO密集型作业立即抢占CPU，很短时间后，CPU密集型进程恢复运行。IO计算并行。系统工作得非常好。

2、凌晨3点。。系统中的作业和它们的运行情况。

答：没有交互式作业。如果有，它们也全部入睡。若干CPU密集型长作业在最低优先级队列时间片轮转。系统中起作用的时间片很长，调度开销小，平均周转时间短。

其实，最低优先权队列，FIFS系统性能可能会更好（一种观点）。

3、某个进程，从CPU密集阶段进入IO密集阶段。。之后回到CPU密集阶段。

答：每隔T，系统会将除队列1之外的所有进程升级至队列1。欲从CPU密集阶段进入IO密集阶段的进程，至多等待T秒会升级至队列0，得到与交互式进程轮转的机会，并最终用不完时间片的事实证明自己已进入IO bound阶段。IO bound阶段，这个进程在队列1，享用高优先级接受系统调度。

进程从IO密集阶段回到CPU密集阶段，系统的反应非常直接，用满时间片直接降级。

4、第二天早上，上班时间到，系统开始人机交互。多级反馈队列中会出现交互式进程，只要有需要，优先使用CPU。

(二) 参考OSTEP 第8章。一个CPU bound, 可以通过巧妙的方式垄断CPU。怎么做? 系统怎样防作弊。

参考答案: 被唤醒的进程, 回原先就绪队列, 排末尾。若重置其时间片, 进程可以调节自己的运行节奏, 在时间片快到期的时候, 执行一次短暂 IO 操作。完成后, 它会回到原队列, 使用完整时间片。用这种方法, 进程可以多占用 CPU 份额。垄断。。。可能很难做到。。。我的考虑是 IO 总是很耗时, 它 IO 的时候会被迫放弃CPU。

防作弊的方法: 被唤醒的进程, 回原先就绪队列, 排末尾。不重置其时间片。

五、证明题

(一) 若被调度进程集合恒定, 考察指标平均周转时间。求证: SJF 策略生成的调度方案理论最优 (也就是: SJF 给出的进程排列平均周转时间理论最短)。

证明:

证明 1:

假设被调度进程集合有 n 个元素。对这 n 个进程的任意排列 $\{P_0, P_1, P_2, \dots, P_{n-1}\}$, 便是一种调度方案。设 $\{T_0, T_1, T_2, \dots, T_{n-1}\}$ 是这 n 个进程的执行时间。

$$T_1 = [M_0 + (M_0 + M_1) + \dots + (M_0 + M_1 + \dots + M_{n-1})] / n;$$

M_i 是第 $i+1$ 个执行的进程的运行时间。

对 M_0, M_1, \dots, M_{n-1} 递增排序, 得 $M_{k_0}, M_{k_1}, \dots, M_{k_{n-1}}$

$$\text{令 } T_2 = [M_{k_0} + (M_{k_0} + M_{k_1}) + \dots + (M_{k_0} + M_{k_1} + \dots + M_{k_{n-1}})] / n$$

对照构成 T_1 和 T_2 的每个累加项, 后者永远不大于前者。故, 能够产生 T_2 的调度方案必为理论最优。 T_2 调度方案是 SJF 生成的, 得证。

证明 2(反证法):

假设被调度进程集合有 n 个元素。排列 $\{P_0, P_1, P_2, \dots, P_{n-1}\}$ 最优, 总周转时间最短。设 $\{T_0, T_1, T_2, \dots, T_{n-1}\}$ 是这 n 个进程的执行时间, 存在 $T_i > T_j$ 且 $i < j$ 。交换 $P_i > P_j$, 可使总调度时间缩短 $((j-i+1)T_i + (j-i)T_{i+1} + \dots + T_j) - ((j-i+1)T_j + (j-i)T_{i+1} + \dots + T_i) = (j-i)(T_i - T_j) > 0$ 。矛盾。最优调度方案中不存在 $T_i > T_j$ 且 $i < j$ 的元素。SJF 理论最优。

证明 3:

证明 SJF 的平均周转时间 T 最优

证: 设有进程执行时间序列 $\{T_{ci}\} \mid 1 \leq i \leq n$ 且用 SJF $T_{ci} \leq T_{ci+1}$

则 $T_{SJF} = \frac{1}{n} (\sum_{i=1}^n S_n)$ S_n 为 T_{ci} 的前 n 项和

T_{SJF} 简化为: $T_{SJF} = \frac{1}{n} \sum_{i=1}^n (n+1-i) T_{ci}$ (交换到首位)

任意交换 $\{T_{ci}\}$ 中的元素, 设交换后为 $\{T_{c\sigma i}\}$, 则交换后

$T_{\sigma} = \frac{1}{n} \sum_{i=1}^n (n+1-i) T_{c\sigma i}$ $T_{\sigma} - T_{SJF} = \frac{1}{n} \sum_{i=1}^n (n+1-i) (T_{c\sigma i} - T_{ci})$

由对称性: 当 $\sigma_i = i$ 时 $T_{c\sigma i} - T_{ci} = 0$

$\therefore T_{\sigma} - T_{SJF} = \frac{1}{2n} [\sum_{\sigma_i < i} (n+1-i) (T_{c\sigma i} - T_{ci}) + \sum_{\sigma_i > i} (n+1-\sigma_i) (T_{ci} - T_{c\sigma i})]$

$= \frac{1}{2n} \sum_{\sigma_i < i} (\sigma_i - i) (T_{c\sigma i} - T_{ci})$ \therefore 交换前 $T_{ci} \leq T_{ci+1}$ \therefore 当 $\sigma_i > i$ 时 $T_{c\sigma i} > T_{ci}$

当 $\sigma_i < i$ 时 $T_{c\sigma i} < T_{ci}$ \therefore 总有 $T_{\sigma} - T_{SJF} > 0$ \therefore SJF 最优 证毕