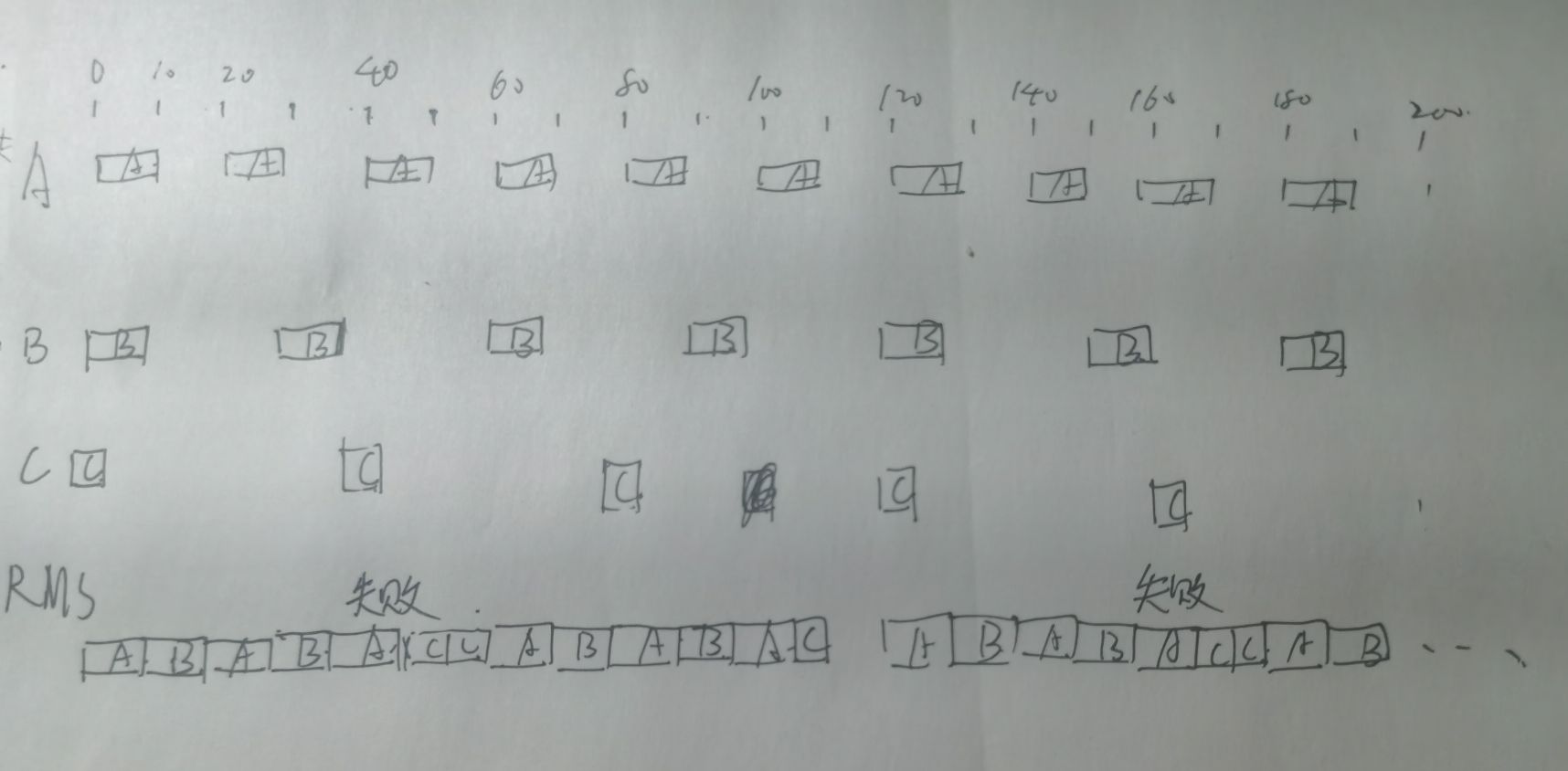
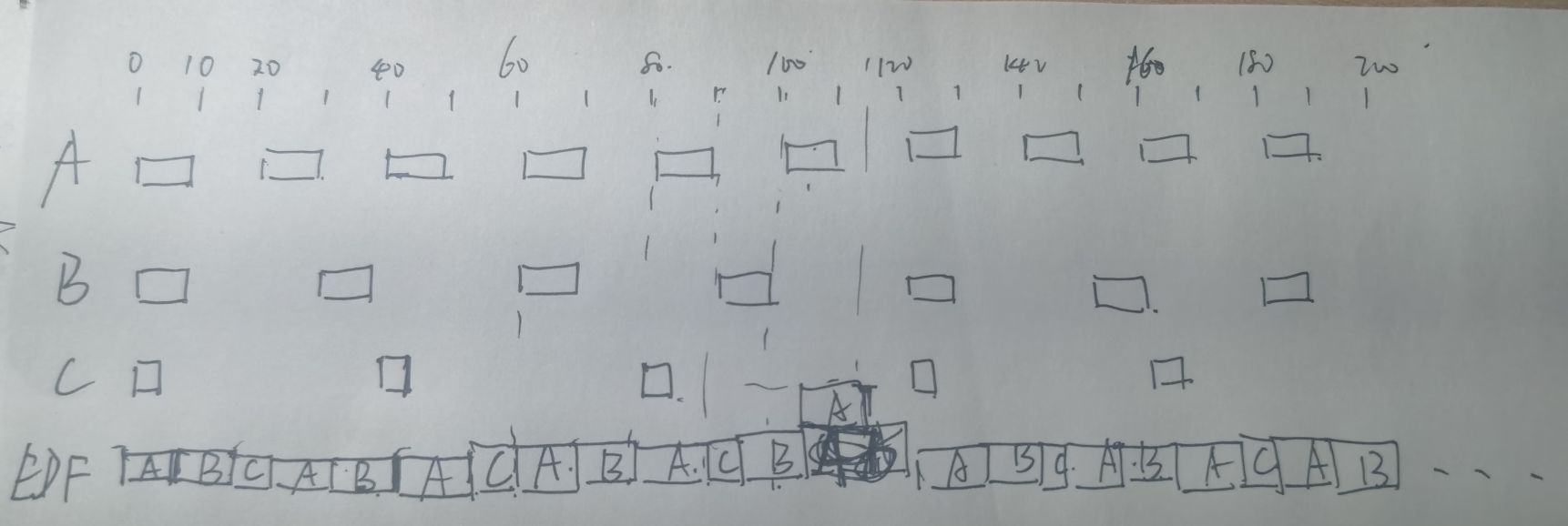
操作系统第五次作业

2019011008 无92 刘雪枫

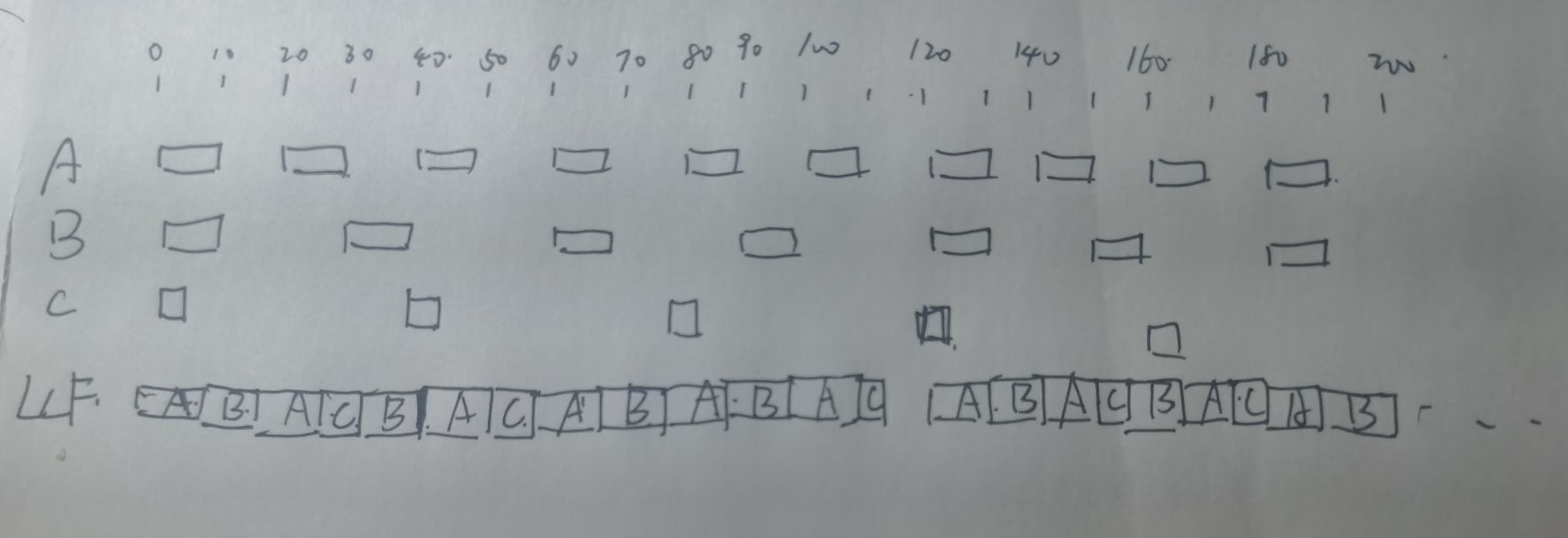
1. 最短响应时间应该采用最短作业优先的调度算法，即次序为：
   1. 若X<=3，则顺序为X、3、5、6、9；
   2. 若3<X<=5，则顺序为3、X、5、6、9；
   3. 若5<X<=6，则顺序为3、5、X、6、9；
   4. 若6<X<=9，则顺序为3、5、6、X、9；
   5. 若X>9，则顺序为3、5、6、9、X。
   6. 先来先服务：运行顺序为A、B、C、D、E。A的周转时间为10分钟，B的周转时间为11分钟，C的周转时间为13分钟，D的周转时间为14分钟，E的周转时间为19分钟。平均周转时间为 (10+11+13+14+19)/5=13.4分钟。
   7. 时间片轮转：由于时间片是1分钟，B和D在第一次获得时间片时便执行完毕，B的周转时间为2分钟，D的周转时间为4分钟。5分钟后还剩下A、C、E，剩余时间分别为9分钟、1分钟和4分钟。第二轮中C运行完毕，周转时间为7分钟。在第8秒末，还剩下A和E，时间分别为8分钟和3分钟。6分钟后E执行完毕，周转时间为14分钟。5分钟后A执行完毕，周转时间为19分钟。因此平均周转时间为 (19+2+7+4+14)/5=9.2分钟。
   8. 短作业优先：执行顺序为B、D、C、E、A，执行时间分别为1、1、2、5、10，因此周转时间分别为1、2、4、9、19。平均周转时间为 (1+2+4+9+19)/5=7分钟。
   9. 最高优先级有限，执行顺序为：B、E、A、C、D，执行时间分别为：1、5、10、2、1，因此周转时间分别为1、6、16、18、19。平均周转时间为 (1+6+16+18+19)/5=12分钟。
   10. RMS：优先级A>B>C，先用10ms执行A，再用10ms执行B，此时A任务又到达，因此再用10ms执行A，此时又有B作业到达，执行B。这时，A和第二次C作业一起到达，因此只能执行A任务。因此作业C不能及时完成。Juti执行情况如图：



* 1. EDF：A最早截止，先用10ms执行A，然后用10ms执行B。这时第二个A任务到达，A与C的截止时间相同，A和C均可执行。消耗15ms。期间有B任务到达，因此再第35秒末只有B任务未完成，因此执行10ms。之后以此类推，如下图，可以看到这种方法可以正常调度：



* 1. LLF：根据最小裕度优先，执行情况如下图：



1. 不安全。执行安全性算法，先把资源分配给P0使其归还资源，可用资源变为2、2、1；再将资源分配给P3，可用资源变为2、2、2。此时P1和P2的R2均需要3个，因此均无法满足，于是可能会发生死锁。因此是不安全状态。
   1. 是安全的。目前各个进程还需要的资源如下表：

|  |  |  |
| --- | --- | --- |
| 2 | 2 | 2 |
| 2 | 0 | 2 |
| 1 | 0 | 3 |
| 4 | 2 | 0 |

先把资源分配给P1，可用资源变为6、2、3；此时资源满足其他所有进程的最大需求，因此可以以任意顺序分配。一个可行的安全序列是P1->P0->P2->P3。

* 1. 若P0进程发出请求1、0、1，执行安全性算法：如果分配则可用资源变为1、1、1，P0还需要的请求变为1、2、1，但此后可用资源便不能满足任何一个进程的请求了，因此进入不安全状态，所以应当让P0阻塞；若P1发出请求，执行安全性算法，可用资源变为1、1、1，P1需要的资源变为1、0、1，此时可用资源还可以让P1满足要求进而回收P1的资源，可用资源变为6、2、3，可以满足其他所有进程的最大需求，因此是安全状态，所以可以分配资源给P1。