Report-Project4

付昊源
517021910753

2019 年 6 月 11 日

1 Scheduling Algorithms

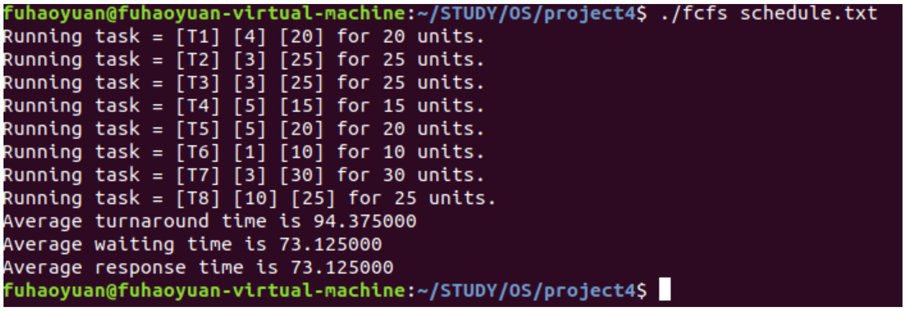
1.1 程序设计思想

在操作系统课上，我们学过的调度算法有 FCFS, SJF, Priority, Round Robin 以及 Priority with Round Robin。本 project 的目的是实现以上五种进程的调度算法。我们需要从文件中读取各个线程的名字、优先级和运行时间，存入 task 结构体中。在教材的描述中，建议我们使用单链表来对所有的线程进行连接，因此，每次读入一个 task，需要将其放入链表中。

对于 schedulers.h 文件中的 add 函数和 schedule 函数，不同的调度算法会有其不同的实现方式，这一部分我们留到核心数据结构及代码解释中详细阐述。

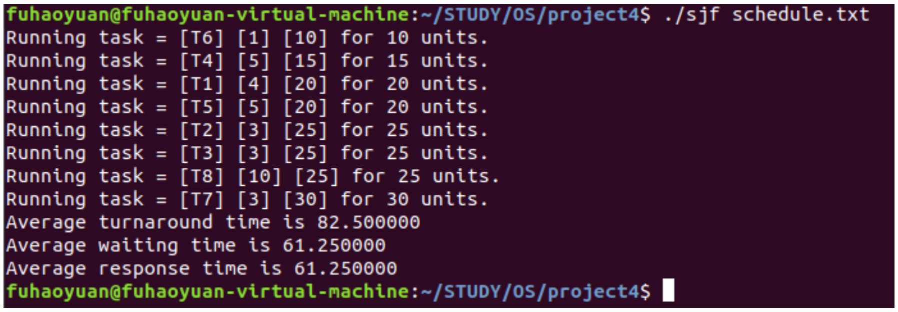
对于平均等待时间、响应时间和周转时间，我们定义了一个 Time 结构体，用于储存三种时间，并通过 schedule 函数获取返回值，在 main 函数中输出。对于不同的调度算法，三种时间的计算方式没什么区别，只需要在 schedule 函数中执行进程时进行修改。

1.2 运行结果截图



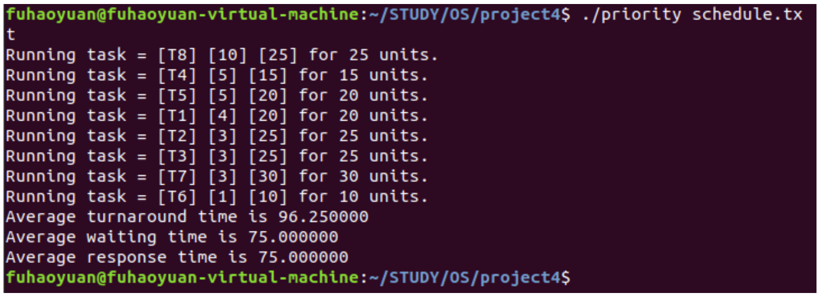
```
fuhaoyuan@fuhaoyuan-virtual-machine:~/STUDY/OS/project4$ ./fcfs schedule.txt
Running task = [T1] [4] [20] for 20 units.
Running task = [T2] [3] [25] for 25 units.
Running task = [T3] [3] [25] for 25 units.
Running task = [T4] [5] [15] for 15 units.
Running task = [T5] [5] [20] for 20 units.
Running task = [T6] [1] [10] for 10 units.
Running task = [T7] [3] [30] for 30 units.
Running task = [T8] [10] [25] for 25 units.
Average turnaround time is 94.375000
Average waiting time is 73.125000
Average response time is 73.125000
fuhaoyuan@fuhaoyuan-virtual-machine:~/STUDY/OS/project4$
```

图 1: FCFS



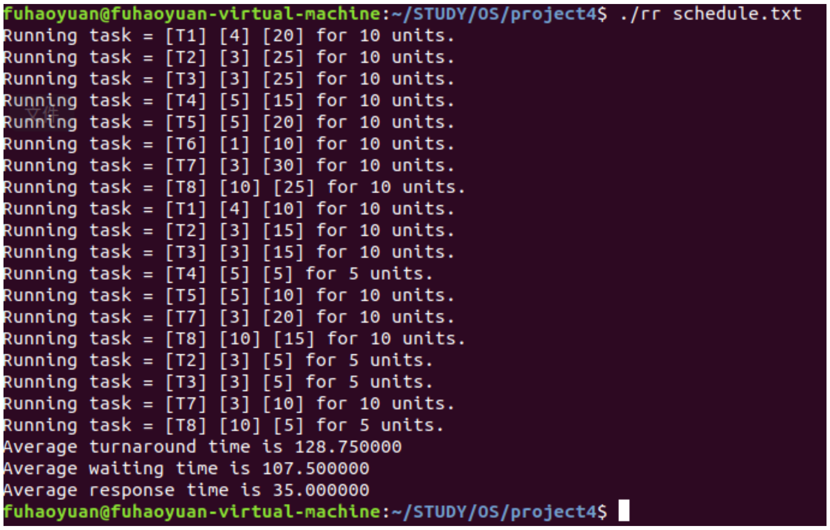
```
fuhaoyuan@fuhaoyuan-virtual-machine:~/STUDY/OS/project4$ ./sjf schedule.txt
Running task = [T6] [1] [10] for 10 units.
Running task = [T4] [5] [15] for 15 units.
Running task = [T1] [4] [20] for 20 units.
Running task = [T5] [5] [20] for 20 units.
Running task = [T2] [3] [25] for 25 units.
Running task = [T3] [3] [25] for 25 units.
Running task = [T8] [10] [25] for 25 units.
Running task = [T7] [3] [30] for 30 units.
Average turnaround time is 82.500000
Average waiting time is 61.250000
Average response time is 61.250000
fuhaoyuan@fuhaoyuan-virtual-machine:~/STUDY/OS/project4$
```

图 2: SJF



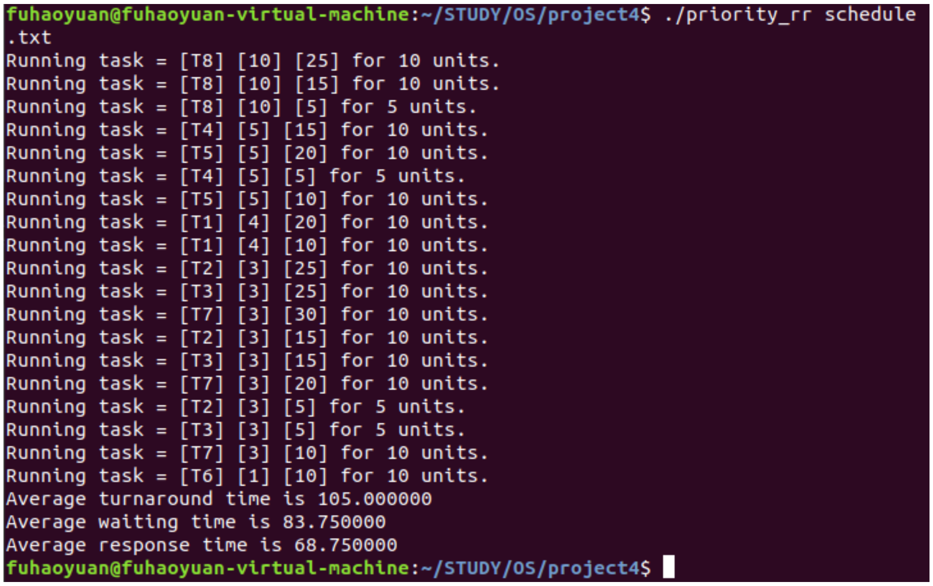
```
fuhaoyuan@fuhaoyuan-virtual-machine:~/STUDY/OS/project4$ ./priority schedule.tx
t
Running task = [T8] [10] [25] for 25 units.
Running task = [T4] [5] [15] for 15 units.
Running task = [T5] [5] [20] for 20 units.
Running task = [T1] [4] [20] for 20 units.
Running task = [T2] [3] [25] for 25 units.
Running task = [T3] [3] [25] for 25 units.
Running task = [T7] [3] [30] for 30 units.
Running task = [T6] [1] [10] for 10 units.
Average turnaround time is 96.250000
Average waiting time is 75.000000
Average response time is 75.000000
fuhaoyuan@fuhaoyuan-virtual-machine:~/STUDY/OS/project4$
```

图 3: Priority



```
fuhaoyuan@fuhaoyuan-virtual-machine:~/STUDY/OS/project4$ ./rr schedule.txt
Running task = [T1] [4] [20] for 10 units.
Running task = [T2] [3] [25] for 10 units.
Running task = [T3] [3] [25] for 10 units.
Running task = [T4] [5] [15] for 10 units.
Running task = [T5] [5] [20] for 10 units.
Running task = [T6] [1] [10] for 10 units.
Running task = [T7] [3] [30] for 10 units.
Running task = [T8] [10] [25] for 10 units.
Running task = [T1] [4] [10] for 10 units.
Running task = [T2] [3] [15] for 10 units.
Running task = [T3] [3] [15] for 10 units.
Running task = [T4] [5] [5] for 5 units.
Running task = [T5] [5] [10] for 10 units.
Running task = [T7] [3] [20] for 10 units.
Running task = [T8] [10] [15] for 10 units.
Running task = [T2] [3] [5] for 5 units.
Running task = [T3] [3] [5] for 5 units.
Running task = [T7] [3] [10] for 10 units.
Running task = [T8] [10] [5] for 5 units.
Average turnaround time is 128.750000
Average waiting time is 107.500000
Average response time is 35.000000
fuhaoyuan@fuhaoyuan-virtual-machine:~/STUDY/OS/project4$ 
```

图 4: Round Robin



```
fuhaoyuan@fuhaoyuan-virtual-machine:~/STUDY/OS/project4$ ./priority_rr schedule
.txt
Running task = [T8] [10] [25] for 10 units.
Running task = [T8] [10] [15] for 10 units.
Running task = [T8] [10] [5] for 5 units.
Running task = [T4] [5] [15] for 10 units.
Running task = [T5] [5] [20] for 10 units.
Running task = [T4] [5] [5] for 5 units.
Running task = [T5] [5] [10] for 10 units.
Running task = [T1] [4] [20] for 10 units.
Running task = [T1] [4] [10] for 10 units.
Running task = [T2] [3] [25] for 10 units.
Running task = [T3] [3] [25] for 10 units.
Running task = [T7] [3] [30] for 10 units.
Running task = [T2] [3] [15] for 10 units.
Running task = [T3] [3] [15] for 10 units.
Running task = [T7] [3] [20] for 10 units.
Running task = [T2] [3] [5] for 5 units.
Running task = [T3] [3] [5] for 5 units.
Running task = [T7] [3] [10] for 10 units.
Running task = [T6] [1] [10] for 10 units.
Average turnaround time is 105.000000
Average waiting time is 83.750000
Average response time is 68.750000
fuhaoyuan@fuhaoyuan-virtual-machine:~/STUDY/OS/project4$ 
```

图 5: Priority with Round Robin

1.3 核心数据结构及代码解释

在 task.h 文件中，我对 task 结构体进行了一些修改，在其中添加了 flag，startTime 和 lastEndTime。其中，flag 变量用于检验该线程是否已经开始执行，startTime 存有该线程的开始执行时间，lastEndTime 储存该线程上一次结束时的时间。这些变量用于计算整个系统的平均等待时间、响应时间和等待时间。

1.3.1 FCFS

FCFS 调度算法简单易懂，每次直接执行最先到达 CPU 的线程，执行结束后再执行下一条线程。由于链表的连接设计，最先存入链表的线程会放在链表末尾，故每次找到该执行的线程时，需要遍历链表找到最末的 task，完成它并将其删除。

1.3.2 SJF

SJF 调度算法的核心是每次找到运行时间最短的线程先完成，再找到剩余线程中时间最短的，以此类推。因此，每次找到该执行的线程，需要遍历整个链表，找到时间最短的 task，完成它并将其删除。

1.3.3 Priority

Priority 调度算法是每次找到优先级最高的线程先完成，再在剩余线程中先完成优先级最高的线程，以此类推。虽然算法的逻辑与 SJF 不同，但实现方式与 SJF 几乎一模一样，只需要将 task 中的判断条件由 burstTime 改为 priority。

1.3.4 Round Robin

Round Robin 调度算法相较于之前的几个算法略微复杂。它的基本思想是设置一个时间片，每次只执行一个进程的时间片的长度，再进行周转。因此在算法的实现上，为了便于在同一个位置（即链表末尾）确定下一步执行哪个线程，我选择每次执行一个线程的时间片后进行判断，若线程已完成，则删除，若未完成，则先取出线程，再存入线程，从而像队列一样改变其位置。需要注意的是，Round Robin 算法需要计算同一个线程两次事件片之间的等待时间。

1.3.5 Priority with Round Robin

Priority with Round Robin 算法与 Round Robin 算法类似，但并不是每次到链表尾寻找下一个执行的线程，而是找到优先级最高的线程。但由于相同优先级的线程要周转进行，我们仍需将该 task 先删除再插入。其余操作与 Round Robin 类似，也需要注意等待时间的计算。