Multiprocessing: scheduling

COSC 208, Introduction to Computer Systems, 2022-04-26

Outline

- Scheduling processes
- First In First Out (FIFO) scheduling
- Shortest Job First (SJF) scheduling
- Preemption
- Shortest Time-to-Completion First (STCF) scheduling
- Round Robin (RR) scheduling

Scheduling processes

- OS decides which process to run and for how long
- *What factors should the OS consider when making these decisions?*
    - Time for process to complete
    - Overhead of context switching
    - Fairness
    - User interaction
- For now, consider one of these metrics: turnaround time
    - $T_{turnaround} = T_{complete} - T_{arrive}$
- For now, assume a process starts and runs to completion—i.e., no I/O and no preemption

First In First Out (FIFO) scheduling

Process	Arrival time	Duration
A	0	15
B	5	15
C	10	15

- *What is the average turnaround time for the processes using FIFO?*

```
┌────────────┬────────────┬────────────┐
│ A          │ B          │ C          │
└────────────┴────────────┴────────────┘
0           15            30          45
```

  - Average Turnaround = (15 + 25 + 35) / 3 = 25
- *What happens if A’s duration is 60?*

```
┌───────────────────────────────────────────────┬──────────────┬──────────────┐
│ A                                             │ B            │ C            │
└───────────────────────────────────────────────┴──────────────┴──────────────┘
0                                              60             75            90
```

  - Average Turnaround = (60 + 70 + 80) / 3 = 70
- *What happens if C’s duration is 60?*

```
┌───────────┬───────────┬────────────────────────────────────────────────────┐
│ A         │ B         │ C                                                  │
└───────────┴───────────┴────────────────────────────────────────────────────┘
0           15          30                                                  90
```

  - Average Turnaround = (15 + 25 + 80) / 3 = 40
- *How can we change the schedule so the average turnaround time when A’s duration is 60 is more like the average turnaround time when C’s duration is 60?*

Shortest Job First (SJF) scheduling

Process	Arrival time	Duration
A	0	60
B	0	15
C	0	15

- *What is the average turnaround time for the above processes using SJF?*

```
┌─────────────┬─────────────┬──────────────────────────────────────────────────────┐
│ B           │ C           │ A                                                    │
└─────────────┴─────────────┴──────────────────────────────────────────────────────┘
0             15            30                                                    90
```

  - Average Turnaround = (15 + 30 + 90) / 3 = 45

Process	Arrival time	Duration
A	0	60
B	5	15
C	10	15

- *What is the average turnaround time for the above processes using SJF?*

```
┌─────────────────────────────────────────────────┬───────────────┬───────────────┐
│ A                                               │ B             │ C             │
└─────────────────────────────────────────────────┴───────────────┴───────────────┘
0                                                 60              75            90
```

  - Average Turnaround = (60 + 70 + 80) / 3 = 70
  - We're back to FIFO—*What happened!? How can we fix this?*

Preemption

- OS only regains control when a system call occurs—e.g., read/write file, yield
  - Syscalls may occur infrequently, or never, due to program design, bugs, or malicious behavior
- *How does an OS forcibly regain control?*—set a timer that raises an interrupt
  - Interrupt causes a trap instruction to be executed
  - Interrupts can also be raised by devices—e.g., Network Interface Card (NIC)
- *What must the OS do if it decides to run another process?*—perform a context switch
  - Save the machine state associated with the process that was running—in particular, the contents of all registers are saved in the process's control structure
  - Restore the machine state associated with the process that should run—again, the contents of all registers are loaded from the process's control structure

Shortest Time-to-Completion First (STCF) scheduling

- Allow preemption
- If a process arrives that has less computation remaining than the currently running process, then preempt the current process and run the new process
- Also known as Preemptive Shortest Job First (PSJF)

Process	Arrival time	Duration
A	0	60
B	5	15
C	10	15

- *What is the average turnaround time for the above processes using STCF scheduling?*

```
┌───┬─────────┬─────────┬─────────────────────────────────┐
│ A │ B       │ C       │ A                               │
└───┴─────────┴─────────┴─────────────────────────────────┘
0   5        20        35                                90
```

  - Average Turnaround = (90 + 15 + 25) / 3 = 43.3
- Now consider response time: $T_{response} = T_{first\_run} - T_{arrive}$
- *What is the average response time for the same processes using STCF scheduling?*
  - Average Response = (0 + 0 + 10) / 3 = 3.3
- Now consider wait time: $T_{wait} = \sum (T_{start\_run} - T_{become\_ready})$
  - A is waiting from time 5 to 35, so T_{wait} = 30
  - B does not wait, so T_{wait} = 0
  - C is waiting from time 10 to 20, so T_{wait} = 10
  - Average Wait = (30 + 0 + 10) / 3 = 13.3
- *What major assumption have we made thus far that is impractical in a real system?*—we know a process's duration (i.e., how much work it has to do)

Round Robin (RR)

- Let each process run for a small amount of time, then switch to the next process; when you get to the last process, then start again with the first process and repeat

Process	Arrival time	Duration
A	Just before 0	60
B	Just before 5	15
C	Just before 10	15

- *What is the average turnaround time and response time for the above processes assuming we let a process run for 5 seconds before switching processes?*

```
┌────┬────┬────┬────┬────┬────┬────┬────┬────┬────────────────────────────────┐
│ A  │ B  │ C  │ A  │ B  │ C  │ A  │ B  │ C  │ A                              │
└────┴────┴────┴────┴────┴────┴────┴────┴────┴────────────────────────────────┘
0    5    10   15   20   25   30   35   40   45                             90
```

  - Average Turnaround = (90 + 35 + 35) / 3 = 53.3
  - Average Response = (0 + 0 + 0) / 3 = 0
  - Average Wait = (30 + 20 + 20) / 3 = 23.3
- In practice, there is a queue of processes that are in the ready state, resulting in the following schedule:

```
┌────┬────┬────┬────┬────┬────┬────┬────┬────┬────┬───────────────────────────┐
│ A  │ B  │ A  │ C  │ B  │ A  │ C  │ B  │ A  │ C  │                           │
└────┴────┴────┴────┴────┴────┴────┴────┴────┴────┴───────────────────────────┘
0    5    10   15   20   25   30   35   40   45   50                         90
```

- *Determine the schedule for the above process with a time quantum of 10.*

```
┌────────┬────────┬────────┬────────┬────┬────┬───────────────────────────────┐
│ A      │ B      │ C      │ A      │ B  │ C  │ A                             │
└────────┴────────┴────────┴────────┴────┴────┴───────────────────────────────┘
0        10       20       30       40   45   50                            90
```

  - Average Turnaround = (90 + 40 + 40) / 3 = 56.6
  - Average Response = (0 + 5 + 10) / 3 = 5
  - Average Wait = (30 + 25 + 25) / 3 = 26.6
- *What happens to average response time as we increase the time quantum?*