# Dalhousie University
## Faculty of Computer Science

**CSCI 3120** **Operating Systems** **Summer-2018**

**Time Allowed: 75 minutes** **Midterm Examination** **[40 marks]**

Name: ______________________________

Student Number: ______________________________

Student Signature: ______________________________

**Exam Instructions:**

1. This exam has 7 pages. Ensure that you have a complete paper.
2. Use the bubble sheet to answer Section 1 that has 25 Questions. Fill in the appropriate choice on the sheet completely using a pencil.
3. Section 2 has two questions. Write your answers in the space provided. If you need more space, use the reverse side of the page.
4. Scientific calculators may be used; however, all other electronic devices are prohibited.
5. Use of books, notes, memoranda etc. is strictly prohibited.
6. Place your book-bags, coats, and books either at the front of the room or underneath your desks.
7. You may not leave the room during the last 5 minutes of your exam. You may not re-enter the room once you leave.
8. You may not ask questions of invigilators. Understanding the question is part of the examination.
9. Write legibly and neatly. I can't mark it if I can't read it.

| Section – Question | Marks Obtained/Total |
|---|---|
| Sec I | /25 |
| Sec II – Q1 | /5 |
| Sec II – Q2 | /10 |
| | |
| Total | /40 |

Best of Luck!

**Section 1: To be completed on the Bubble sheet [25 marks]**
For each of the following questions, completely fill in the circle corresponding to the best answer on the bubble sheet provided

1. A process control block ____.
   - **A) includes information on the process's state**
   - B) stores the address of the next instruction to be processed by a different process
   - C) determines which process is to be executed next
   - D) is an example of a process queue

2. A process whose parent has terminated without calling wait(), is known as a(n) ________ process.
   - A) zombie
   - **B) orphan**
   - C) terminated
   - D) init

3. In Unix, the _______ process is assigned as the parent to orphan processes.
   - A) zombie
   - **B) init**
   - C) main
   - D) renderer

4. ______ is the number of processes that are completed per time unit.
   - A) CPU utilization
   - B) Response time
   - C) Turnaround time
   - **D) Throughput**

5. A race condition ____.
   - A) results when several threads try to access the same data concurrently
   - **B) results when several threads try to access and modify the same data concurrently**
   - C) will result only if the outcome of execution does not depend on the order in which instructions are executed
   - D) None of the above

6. The ____________ refers to the number of processes in memory.
   - A) process count
   - B) long-term scheduler
   - **C) degree of multiprogramming**
   - D) CPU scheduler

7. A process may transition to the Ready state by which of the following actions?
   - A) Completion of an I/O event
   - B) Awaiting its turn on the CPU
   - C) Newly-admitted process
   - **D) All of the above**

8. A significant problem with priority scheduling algorithms is _____.
   - A) complexity
   - **B) starvation**
   - C) determining the length of the next CPU burst
   - D) determining the length of the time quantum

9. The Windows CreateProcess() system call creates a new process. What is the equivalent system call in UNIX?
   - A) NTCreateProcess()
   - B) process()
   - **C) fork()**
   - D) getpid()

10. The ____ of a process contains temporary data such as function parameters, return addresses, and local variables.
    - A) text section
    - B) data section
    - C) program counter
    - **D) stack**

11. When a child process is created, which of the following is a possibility in terms of the execution or address space of the child process?
   A) The child process runs concurrently with the parent
   B) The child process has a new program loaded into it.
   C) The child is a duplicate of the parent
   **D) All of the above**

12. _____ allow operating system services to be loaded dynamically.
   A) Virtual machines
   **B) Modules**
   C) File systems
   D) Graphical user interfaces

13. A ____ can be used to prevent a user program from never returning control to the operating system.
   A) portal
   B) program counter
   C) trap
   **D) timer**

14. If a disk has 256 cylinders, how many tracks does it have on each platter's surface?
   A) 8
   B) 128
   **C) 256**
   D) 512

15. The list of processes waiting for a particular I/O device is called a(n) _______
   A) standby queue
   **B) device queue**
   C) ready queue
   D) interrupt queue

**Study the piece of code shown below and answer questions 16 – 18:**

```
int main()
{
   int tot_sum = 20;
   pid_t xid = fork();
   if (xid > 0)
   {
      int new_sum = 10;
      new_sum = new_sum + tot_sum;     /* line A */
      tot_sum = new_sum;
      wait(NULL);
      return 0;
      xid = fork();
   }
   else if (x == 0)
   {
      int new_sum = 30;
      new_sum = new_sum - tot_sum;     /* line B */
      tot_sum = new_sum;
      return 0;
      fork();
   }
}
```

16. What is the value of new_sum after line A is executed?

A) 0
B) 10
C) 20
**D) 30**
E) Cannot be determined from this code

17. What is the value of new_sum after line B is executed?

A) 0
**B) 10**
C) 20
D) 30
E) Cannot be determined from this code

18. How many processes are created in total (including the parent process) when this code is run?

**A) 2**
B) 4
C) 8
D) 16
E) Cannot be determined from this code

**Study the piece of code shown below and answer questions 19 – 21:**

```
#include …  /* assume all appropriate files are included and code compiles */
static int sum = 0;
void *myfunc()
{
  sum++;     /* line A */
}
int main()
{
   sum = 10;
   pid_t pid = fork();
   if (pid == 0)
   {
      pthread_t *tid;  /* line B */
      pthread_create(tid, NULL, myfunc, NULL);
      pthread_join(*tid, NULL);
      sum = sum * 2;   /* line C */
   }
   return 0;
}
```

19. What is the value of sum after line A is executed?

A) 0
B) 1
C) 10
**D) 11**
E) 22

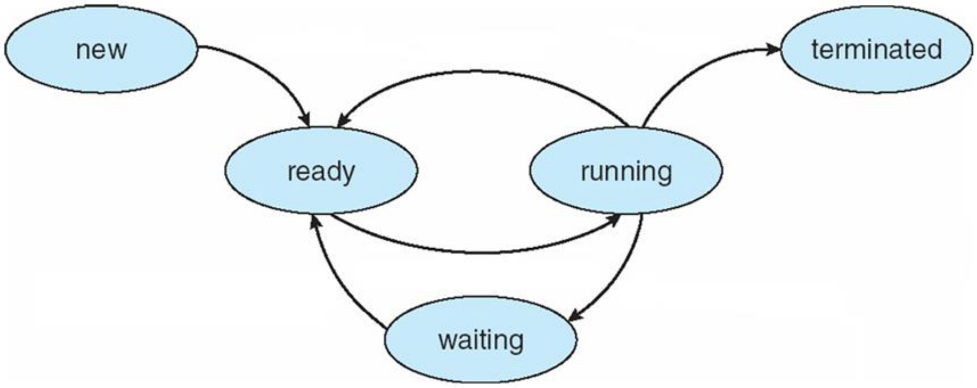
20. What is the value of sum after line B is executed?

A) 0
**B) 10**
C) 11
D) 20
E) 22

21. What is the value of sum after line C is executed?

A) 0
B) 10
C) 11
D) 20
**E) 22**

**Study the figure below that shows the various states a process can be in, and answer questions 22 – 25:**




22. Which of the event below moves the process from the running state to the ready state?
A) Process is admitted into the scheduler
**B) Process is interrupted by a timer**
C) Process is dispatched by the scheduler
D) I/O for the process completes
E) Process issues a trap or system call

23. Which of the event below moves the process from the waiting state to the ready state?
A) Process is admitted into the scheduler
B) Process is interrupted by a timer
C) Process is dispatched by the scheduler
**D) I/O for the process completes**
E) Process issues a trap or system call

24. Which of the event below moves the process from the running state to the waiting state?
A) Process is admitted into the scheduler
B) Process is interrupted by a timer
C) Process is dispatched by the scheduler
D) I/O for the process completes
**E) Process issues a trap or system call**

25. Which of the event below moves the process from the ready state to the running state?
A) Process is admitted into the scheduler
B) Process is interrupted by a timer
**C) Process is dispatched by the scheduler**
D) I/O for the process completes
E) Process issues a trap or system call

**Section II:**

**Question 1:** [5 marks]

a) Cache and RAM may be made of the same hardware; however, cache is usually much faster than RAM. Briefly explain why? [2 marks]

**Answer:**

**Cache is located closer to the CPU compared to RAM and this reduces the delay (also known as the signal's propagation delay)**

b) Consider a computer system that has cache memory, main memory (RAM) and disk. It takes 1 ns to access a word from the cache, 10 ns to access a word from the RAM and 10 ms to access a word from the disk. If the cache miss rate is 10%, and the main memory miss rate (after a cache miss) is 0%, what is the average time to access a word? Show your calculations. [3 marks]

**Solution:**

***Cache hit = 90% * 1 ns = 0.9ns***
***Cache miss = 10% * 1ns = 0.1ns -> Note that cache is accessed anyways***
***Memory hit = 10% * (100%*10 ns) = 1ns***

***Hence, average total time to access a word = 0.9 + 0.1 + 1 = 2.0 ns***

**Question 2: Scheduling algorithms [10 marks]**

Given that the following processes arrive for execution at the times indicated in the Table shown below. Each process will run for the time listed. The CPU will be idle if there is no process to run. In answering the questions, base all decisions on the information you have at the time the decision must be made. Consider all times are in milliseconds.

a) Draw the **Gantt chart** and state the **turnaround time** and **waiting time** for each of these processes for the **non-preemptive SJF scheduling** algorithm in Table 1 below. [3 marks]

b) Draw the **Gantt chart** and state the **turnaround time** and **waiting time** for each of these processes for the **preemptive SJF scheduling** (shortest remaining time first) algorithm in Table 2 below. [5 marks]

c) Find the **percentage CPU utilization** (defined as the percentage of time the CPU was busy executing processes compared to the total time). [2 marks]

| Process | Arrival Time | Burst Time |
|---|---|---|
| P1 | 0.0 | 5 |
| P2 | 10.0 | 10 |
| P3 | 11.0 | 5 |
| P4 | 13.0 | 2 |
| P5 | 32.0 | 8 |

**Solution:**

**a) Table 1: Non-preemptive SJF**

| Process | Waiting time | Turnaround time |
|---|---|---|
| P1 | **0** | 5 – 0 = **5** |
| P2 | **0** | 20 – 10 = **10** |
| P3 | 22 – 11 = **11** | 27 – 11 = **16** |
| P4 | 20 – 13 = **7** | 22 – 13 = **9** |
| P5 | **0** | 40 – 32 = **8** |

| P1 | Idle | P2 | P4 | P3 | Idle | P5 |
|---|---|---|---|---|---|---|

0 5 10 20 22 27 32 40

**b) Table 2: Preemptive SJF**

| Process | Waiting time | Turnaround time |
|---|---|---|
| P1 | **0** | 5 – 0 = **5** |
| P2 | 0 + (18-11) = **7** | 27 – 10 = **17** |
| P3 | 0 + (15-13) = **2** | 18 – 11 = **7** |
| P4 | **0** | 15 – 13 = **2** |
| P5 | **0** | 40 – 32 = **8** |

| P1 | Idle | P2 | P3 | P4 | P3 | P2 | Idle | P5 |
|---|---|---|---|---|---|---|---|---|

0 5 10 11 13 15 18 27 32 40

**c) % utilization for both = (40-10)/40 = 30/40 = 0.75 = 75%**