

CS211: Programming and Operating Systems
Semester 2 Sample Exam, 2019/2020

Answers to Questions

- Q1. Put these operating systems in order in which they were first developed. **Answer:** first UNIX (1970 approx), then MS-DOS (1981), and then Linux (1991)
- Q2. Which of the following devices run a Linux-based operating system (select all that apply)? **Answer:** all of Raspberry Pi; All of the world's 500 most powerful supercomputers; and Most (over 80%) of smartphones.
- Q3. Which of the following C header files defines the `scanf()` function? **Answer:** `stdio.h`
- Q4. Which of the following are data types in C? (Select all that apply)? **Answer:** `float`, `double`, and `char`. But not `triple` or `string`.
- Q5. What is the name of the function first called when a compiled C program is run? (Note: answer is case-sensitive). **Answer:** `a. main()`
- Q6. In C, which of the following is a comment (select all that apply)? **Answer:**
- a) `# this is *not* a comment in C`
 - b) `//this is a comment in C`
 - c) `/* this too is a comment in C?`
- Q7. Match of the following data types with their corresponding conversion character used by `printf()`
- Answer:** `%f` converts `float`,
`%c` converts `char` `%d` converts `int`
- Q8. The `fork()` function creates a child process. What does it return? **Answer:** c) The child's PID to the parent, and 0 (zero) to the child.
- Q9. Which of the following functions creates a unidirectional data channel that can be used for interprocess communication? **Answer:** d) `pipe()`
- Q10. In C, which of the following is the correct syntax for an `if`-statement for checking if the variable `x` stores the value 211? Note:
- the answer is case-sensitive
 - there could be more than one correct answer
- Answer:** this is the only correct one:
- a) `if (211 == x)`
- Q11. Which of the following is the *string termination character* for a C-string? **Answer:** a) `\0`
- Q12. Which of the following CPU scheduling algorithms is preemptive? Select all that apply. **Answer:**
- a) First-Come-First-Served (FCFS) is *not* preemptive.
 - b) Shortest-Job-First (SJF) is *not* preemptive.
 - c) Round-Robin (RR) *is* preemptive.

Q13. Four processes are submitted in the following order, all at time $t = 0$,

- a. P_1 with a burst time of 16 seconds
- b. P_2 with a burst time of 12 seconds
- c. P_3 with a burst time of 8 seconds
- d. P_4 with a burst time of 4 seconds

Calculate the average wait time for each of:

- i. First-Come-First-Served (FCFS). **Answer:** P_1 waits 0s, P_2 waits 16s, P_3 waits 18s, and P_4 waits 36s. So the average is $(0 + 16 + 18 + 36)/4 = 20s$.
- ii. Shortest-Job-First (SFJ), **Answer:** P_1 waits 24s, P_2 waits 12s, P_3 waits 4s, and P_4 waits 0s. So the average is $(24 + 12 + 4 + 0)/4 = 10s$.
- iii. Round Robin with a time quantum of $q = 4$ seconds. **Answer:** P_1 waits 24s, P_2 waits 24s, P_3 waits 20s, and P_4 waits 12s. So the average is $(24 + 24 + 20 + 12)/4 = 20s$.

Q14. True or false? There are situations where each of

- i. first-Come-First-Served (FCFS),
- ii. shortest-Job-First (SFJ), and
- iii. round robin (RR) with a time quantum of $q = 4$ seconds

give exactly the same average wait and turn-around times. Give an example to support your answer.

Answer: Yes. For example if every process has the same burst time, which is no more than the time quantum, the all three give the same result.

Q15. Recall the *Dining Philosophers Problem* for as a model for process synchronisation.

Determine if the following statements are **true** or **false**.

- 1. If there are five diners, and five chopsticks, and all diners pick up a chopstick at the same time, the system will be in deadlock. **Answer:** True.
- 2. If there are only four diners, and five chopsticks, the system **cannot** reach a deadlocked state. **Answer:** True. At least one diner will have access to a chopstick on their left and right. So they will eat. Then they will put them down, so others can eat. Etc.
- 3. If there are six diners, and five chopsticks, the system **cannot** reach a deadlocked state. **Answer:** False.

Q16. Suppose a system with two memory partitions, of size 100k and 150k, in that order.

Four jobs requiring (contiguous) memory space of various sizes are submitted at the same in the following order:

- i. P_1 , which requires 70k
- ii. P_2 , which requires 80k
- iii. P_3 , which requires 50k
- iv. P_4 , which requires 50k

Would all these process be allocated memory if the **First-Fit** (FF) scheme is employed? If not, which one(s) would be omitted?

Answer: For FF, first P_1 would be allocated to the 100k partition (leaving 30k of space), then P_2 would be allocated to the 150k partition (leaving 70k of space), and finally P_3 would be allocated to part of what is left of the 150k partition (leaving 20k of space in it). P_4 would not be allocated.

Would all these process be allocated memory if the **Worst-Fit** (WF) scheme is employed? If not, which one(s) would be omitted? **Answer:** For WF, first P_1 would be allocated to the 150k partition (leaving 80k of space), then P_2 would be allocated to the 100k partition (leaving 20k of space), and finally P_3 would be allocated to part of what is left of the 150k partition (leaving 30k of space in it). P_4 would not be allocated.

Q17. Consider the following piece of C code.

```
int x=0;
fork();
x++;
fork();
x++;
printf("The value of x is: %d.\n",x);
return(0);
}
```

What output should be generated? **Answer:**

The value of x is: 2.
The value of x is: 2.
The value of x is: 2.
The value of x is: 2.