



# University of Victoria

## Midterm 1 PRACTICE QUESTIONS

### Fall 2025

Course Name & No.	CSC 360: Operating Systems
Section (CRN)	
Instructor	Wenjun Yang
Duration	50 Minutes

- This series of practice questions reflects the grade distribution and structure of our actual exam. You may expect the exam to have the same structure and distribution of concepts, albeit with different questions.
- The marks included may serve to help guide your understanding of work distribution within the exam, as you may assume the actual midterm to have similarly weighted questions.

## Part 1: Single Choice Questions (30 Marks)

No practice is provided here, however you may assume that these questions will be constructed from the concepts covered in the study guide.

Distribution: 10x3 pts.

- This part omitted from practice exam.

## Part 2: Written Answer Questions (70 Marks)

The following questions require written responses. Point-form should be used where possible, providing they accurately convey the idea behind your response.

Distribution: 30+40 pts.

**Make sure to state any assumptions made.**

### Question 11 (20+10 Marks)

Consider the syntactically correct C code below:

```
1  #include <sys/types.h>
2  #include <stdio.h>
3  #include <unistd.h>
4  #include <sys/wait.h>
5
6  int counter = 10;
7
8  int main(){
9      pid_t pid;
10     pid = fork();
11     if (pid == 0) {
12         counter *= 3;
13         printf("CHILD: counter = %d\n", counter);
14         return 0;
15     }
16     else if (pid > 0) {
17         wait(NULL);
18         printf("PARENT: counter = %d\n", counter);
19         return 0;
20     }
21 }
```

**11.a (20 Marks)**

After running the program, what will be the complete output from both lines 13 and 18?

**11.b (10 Marks)**

In no more than 1-2 sentences, explain what would happen if we moved the `wait(NULL)` call from line 17 to after the `printf()` statement on line 18?

## Question 12 (40 Marks)

Consider the syntactically correct C code below:

```
1 #include <unistd.h>
2
3 int main(){
4     fork();
5     fork();
6     fork();
7     return 0;
8 }
```

Including the initial parent process, how many process are created by the program shown above? You may use a tree-like structure to help visualize this.

Partially correct answers with a clear visualization may receive partial marks.