School of Computing

Final Assessment for AY2021/2022, Semester 2

CS2106 - Introduction to Operating Systems

Data Races vs. Scheduling

Each question in this section describes a different system. The same multi-threaded program *P* with **four (4)** threads is running on all the systems in this section. All four threads are executing the same *C* code shown below, which increments a globally shared variable `x`:

```
void threadRun(){
    x++;
}
```

Your task is to identify whether data races can happen (race condition) when program *P* runs on a given system.

Variable `x` is has value **0**, and the value of `x` after the correct execution of the program is supposed to be **4**.

Assume that every system has a proper compiler that takes advantage of the underlying instructions.

1. System H

   a) One CPU core

   b) First-Come First-Served Scheduling non-preemptive process scheduling

   c) There is **no** instruction INC, which increments an integer value in memory in a single instruction. Instead, the compiler would use three instructions (one to load `x` from memory into a register, another to increment the register, and, the third to store `x` back into memory).

   Question: Is the following statement true or false?

   Data races could happen in program P when executing on System H.

   (2 marks)

   - True
   - False

2. System G

   a) One CPU core

   b) Round-Robin process scheduling

   c) There is an instruction INC, which increments an integer value in memory in a single instruction

   Question: Is the following statement true or false?

   Data races can happen in program P when executing on System G.

(2 marks)

- True
- False

3. System R

a) Two CPU cores

b) A **preemptive** scheduling algorithm.

c) There is an instruction **`atomic_INC`**, which increments an integer value in memory in a single **atomic** instruction (similarly to test-and-set).

Question: Is the following statement true or false?

Data races could happen in program P when executing on System R.

(2 marks)

- True
- False

4. System T

a) Two CPU cores

b) A **preemptive** scheduling algorithm.

c) There is **no** instruction INC, which increments an integer value in memory in a single instruction. Instead, the compiler would use three instructions (one to load **`x`** from memory into a register, another to increment the register, and, the third to store **`x`** back into memory).

Question: Is the following statement true or false?

Data races can happen in program P when executing on System T.

(2 marks)

- True
- False

False

5. System M

a) One CPU core

b) A **preemptive** scheduling algorithm.

c) There is **no** instruction INC, which increments an integer value in memory in a single instruction. Instead, the compiler would use three instructions (one to load `x` from memory into a register, another to increment the register, and, the third to store `x` back into memory).

Question: Is the following statement true or false?

Data races can happen in program P when executing on System M.

(2 marks)

- True
- False

6. System J

a) Two CPU cores

b) A **preemptive** scheduling algorithm.

c) There is an instruction INC, which increments an integer value in memory in a single instruction.

Question: Is the following statement true or false?

Data races could happen in program P when executing on System J.

(2 marks)

- True
- False

Buddies

A **1MiB** region of memory is managed by a **buddy allocator**. The minimum unit of allocation is **1KiB**, and the maximum possible allocation is **1MiB**.

Currently, the entire memory region is free.

7. Fill in the blanks

(3 marks)

First comes process A and requests **130KiB**. The allocator will have to break exactly ___1___ partitions to serve A's request.

Enter the correct answer below.

1 [ ] Please enter a number for this text box.

8. Fill in the blanks

(3 marks)

Then comes process B and requests **257 KiB**. The allocator will have to break exactly ___1___ partitions to serve B's request.

Enter the correct answer below.

1 [ ] Please enter a number for this text box.

9. Fill in the blanks

(3 marks)

Finally, Process C comes and requests **2KiB**. The allocator will break exactly ___1___ partitions to serve this request.

Enter the correct answer below.

1 [ ] Please enter a number for this text box.

10. Fill in the blanks

(4 marks)

After these three requests are served, the total amount of memory lost to **internal** fragmentation is 1 KiB.

Enter the correct answer below.

1 Please enter a number for this text box.

File Systems

Use the following information to answer the four questions that follow:

We have a file system with the following inode structure:

File metadata
Direct pointer 1
Direct pointer 2
…
Direct pointer 6
Single indirect pointer 1
Single indirect pointer 2
Double indirect pointer 1
Triple indirect pointer 1

Assuming that each block number is 32 bits long and each data block is 256 bytes long.

11. Fill in the blanks

(3 marks)

What is the maximum possible size of a file in this file system in MiB (1 MiB = 1024 x 1024 bytes). Express your answer in 2 decimal places.

Answer: 1 MiB (2 dp)

Enter the correct answer below.

1 Please enter a number for this text box.

12. Fill in the blanks

(3 marks)

What is the maximum number of data blocks that can be indexed in this file system? Express your answer in powers of 2. (The ^ means "to the power of")

Maximum number of data blocks that can be indexed = 2^__1__

Enter the correct answer below.

1 Please enter a number for this text box.

13. Fill in the blanks

(3 marks)

Assuming that our file system has a maximum of 256 inodes, in the worst case how many data blocks are used as index blocks? Assume that inodes themselves are not stored in data blocks.

In the worst case, __1__ data blocks would be used as index blocks.

Enter the correct answer below.

1 Please enter a number for this text box.

14. Fill in the blanks

(3 marks)

Assuming that each file uses exactly one inode, what is the maximum amount of file data that can be stored, in this file system? Express your answer to 2 decimal places in GiB (1 GiB = 1024 x 1024 x 1024 bytes).

Maximum amount of file data: __1__ GiB (2 dp)

Enter the correct answer below.

1 Please enter a number for this text box.

Drive Head Scheduling

The operating system needs to access the following track numbers at the times shown. If more than one track is to be accessed at the same time unit, then the requests arrived in the order shown for that time unit (e.g. in time unit 1, the request for track 20 came before track 15).

Time	Track to be accessed
1	20, 15
2	3, 7, 4
3	8
4	5

The operating system uses a scheduling window of two time units. That is, it batches and schedules track accesses across 2 time units. For example, it takes the track accesses for time units 1 and 2 and schedules them together. It then takes the accesses for times 3 and 4 and schedules them together, etc.

15. Fill in the blanks

(7 marks)

Fill the blanks with the schedule in terms of track numbers for SCAN scheduling. The head is currently at track 0.

Put one track number per blank. Do not put in track 0.

1 ____

2 ____

3 ____

4 ____

5 ____

6 ____

7 ____

Enter the correct answer below.

1 Please enter a number for this text box.

2 Please enter a number for this text box.

3 Please enter a number for this text box.

4 Please enter a number for this text box.

5 Please enter a number for this text box.

6 Please enter a number for this text box.

7 Please enter a number for this text box.

16. Fill in the blanks

(2 marks)

How many tracks does the head travel in total for SCAN scheduling, assuming that the head was originally at track 0?

Number of tracks traveled: 1 tracks.

Enter the correct answer below.

1 Please enter a number for this text box.

17. Fill in the blanks

(2 marks)

If we had used FCFS scheduling instead, how many tracks would the head have traveled, assuming that the head was originally at track 0?

Number of tracks traveled: 1 tracks.

Enter the correct answer below.

1 Please enter a number for this text box.

Fill in the blanks

18. (2 marks)

Suppose we used a scheduling window of 4 instead of 2. How many tracks in total would the head have to travel, assuming that the head was originally at track 0?

Number of tracks traveled: 1 tracks.

Enter the correct answer below.

1 Please enter a number for this text box.

Finish Quiz Save For Later

In the Hair Salon

We now explore the ***The Hair Salon Problem.***

Somewhere in Clementi there's a hair salon with **10 waiting chairs** for customers who wait for a haircut, and **one barber chair** (similar to the one shown below), where the customer is seated while their hair is being cut.

Every customer visits the salon periodically. If they enter the salon when all the waiting chairs are occupied, then they queue outside the salon (must be blocked) until a waiting chair is free. If a waiting chair is available, but the barber is busy doing a haircut for another customer, then the customer will sit in one of the available waiting chairs (in the blocked state). A customer is seated in the barber chair (in the blocked state) during the haircut to allow more customers to wait seated in the waiting chairs. When there are no customers, the barber sleeps (in the blocked state) until a new customer comes in.

Your task in the next four questions:

Complete the following code skeleton (the missing parts colored in red) to correctly synchronize the barber and the customer processes.

```
#include <cs2106_semaphore.h>
#define WAITING_CHAIRS 10
INITIALIZATION
```

```
void barber(){
  while(1)
   {
      BARBER_CODE1
      cutHair();
      BARBER_CODE2
     }

}
```

```
void customer() {
  while(1){
     CUSTOMER_CODE
     //visit the barber shop
     delay(rand());
     // wait for a random amount of
     // time between 2 shop visits
  }
  }
```

You are allowed to use shared variables, as well as ***named semaphores*** defined in **cs2106_semaphore.h**, which are semaphores identified by name:

```
// creates a named semaphore with an initial value
void create_sem(char *sem_name, int val);
// waits on the semaphore of a given name
void sem_wait(char *sem_name);
// signals to the semaphore of a given name
void sem_signal(char *sem_name);
```

1. INITIALIZATION:

   Define and initialize any shared variables in this section.

   (4 marks)

   Enter your answer here

2. BARBER_CODE1:

   (2 marks)

   Enter your answer here

3. BARBER_CODE2:

   (3 marks)

   Enter your answer here

4. CUSTOMER_CODE:

   (5 marks)

Enter your answer here

The adventures of Elon Musk

A hardware startup company funded by Elon Musk decided to build a processor (codenamed *Konstrakta*) that is very similar to the Intel processors found in the xcne machines that we used in the labs.

Just like the processor in the xcne machines, the *Konstrakta* processor uses a hierarchical direct page-table structure, with each table/directory entry occupying **64 bits** (8 bytes) regardless of the level in the hierarchy. The *Konstrakta* processor has a **4-level** page-table structure and the physical frame size is **4KB**, just like the xcne processors. However, the key difference is the size of the virtual address in *Konstrakta* chips is **50 bits** instead of 48.

Assume Process P runs the following simple program on a *Konstrakta* processor.

```
#include<stdio.h>
#include<stdlib.h>
#define HALF_GIGA 1<<29
#define NUM_PAGES 1<<17
void main(){
    char* array = malloc(HALF_GIGA);
    int i;
    for(i=0; i < NUM_PAGES; i++)
          array [i<<12]='a';
    printf("0x%lX\n", array);
    //point of interest
    free(array);
}
```

The program prints out the following line, which is the hexadecimal representation of the value stored in variable **array**:

0x1FFFFFFFF0000

(there are 8 Fs in the above number)

5. What is the key advantage of *Konstrakta* over the processors in xcne machines? How important would that advantage be in daily usage of the xcne cluster?

   (2 marks)

Enter your answer here

6. Explain the limitations of *Konstrakta* processors with respect to paging.

In other words, what limitations does *Konstrakta* have compared to the processors in the lab machines?

(4 marks)

Enter your answer here

7. Fill in the blanks

(2 marks)

What is the maximum number of entries that a process can occupy in the **root-level** page directory on the *Konstrakta* machine?

Your answer: 1

Enter the correct answer below.

1

8. Fill in the blanks

(2 marks)

When the program execution reaches the point of interest, in total 1 **valid entries in the root directory** are holding information related to process P's

dynamically allocated data.

Enter the correct answer below.

1 [ ] Please enter a number for this text box.

9. Fill in the blanks

(2 marks)

When the program execution reaches the point of interest, in total __1__ **valid entries** are holding information related to process P's dynamically allocated data in the **second level** of the hierarchical page-table structure (next after the root).

Enter the correct answer below.

1 [ ] Please enter a number for this text box.

10. Fill in the blanks

(2 marks)

When the program execution reaches the point of interest, there are in total __1__ **physical frames** holding information related to process *P*'s dynamically allocated data in the **last level** of the page-table structure.

Enter the correct answer below.

1 [ ] Please enter a number for this text box.

System X

100 very long compute-intensive processes are running on System X. All processes fit in memory entirely and do not perform any I/O operations. Assume that the processes do not communicate with each other and that there are no other processes in the system.

Some stats about System X:

1. It has 8 CPU cores

2. It uses the Round-Robin process scheduling

3. Upon a context switch, System X stores as much of the hardware context as possible on the stack of the corresponding process, and the remaining state is saved in the process control block (PCB).

11. Fill in the blanks

(2 marks)

There are in total __1__ stack pointer registers on System X.

Enter the correct answer below.

1 Please enter a number for this text box.

12. Fill in the blanks

(3 marks)

At any given time, at least __1__ processes will have the most up-to-date value of the stack pointer in their process control block (PCB).

Enter the correct answer below.

1 Please enter a number for this text box.

13. Fill in the blanks

(3 marks)

At any given time, at least __1__ processes will have the most up-to-date values of their general-purpose registers in their process control block (PCB).

Enter the correct answer below.

1 Please enter a number for this text box.

14. Fill in the blanks

(2 marks)

At any given time, the number of processes in the BLOCKED state is at least 1 .

Enter the correct answer below.

1 Please enter a number for this text box.

15. Fill in the blanks

(2 marks)

At any given time, the number of processes in the READY state is at least 1 .

Enter the correct answer below.

1 Please enter a number for this text box.

Disjoint Choices

16. Choose the correct statement regarding the management of the permission bits in case of two disjoint memory allocation techniques, **segmentation** and **paging**.

(2 marks)

- Permission bits are more efficiently managed in case of paging compared to segmentation.
- Permission bits are more efficiently managed in case of segmentation compared to paging.
- Permission bits are managed in the same manner in case of both segmentation and paging.
- None of the other answers is correct.

17. Choose the correct statement regarding **internal** fragmentation in case of **pure segmentation** and **segmentation with paging**.

(2 marks)

- Segmentation with paging is expected to have more internal fragmentation compared to pure segmentation.
- Segmentation with paging is expected to have less internal fragmentation compared to pure segmentation.
- Segmentation with paging and pure segmentation are expected to have the same amount of internal fragmentation.
- None of the other answers is correct.

18. Choose the correct statement regarding **external** fragmentation in case of **pure segmentation** and **segmentation with paging**.

(2 marks)

- Pure segmentation is expected to have more external fragmentation compared to segmentation with paging.
- Pure segmentation is expected to have less external fragmentation compared to segmentation with paging.
- Pure segmentation and segmentation with paging are expected to have the same amount of external fragmentation.
- None of the other answers is correct.

19. Choose the correct statement regarding **internal** fragmentation in **pure paging** and **segmentation with paging**.

(2 marks)

- Pure paging is expected to have more internal fragmentation compared to segmentation with paging.

Pure paging is expected to have less internal fragmentation compared to segmentation with paging.

Pure paging and segmentation with paging are expected to have the same amount of internal fragmentation.

None of the other answers is correct.

20. Choose the correct statement regarding **external** fragmentation in **pure paging** and **segmentation with paging**.

(2 marks)

In case of pure paging, the amount of external fragmentation is larger compared segmentation with paging.

In case of pure paging, the amount of external fragmentation is smaller compared segmentation with paging.

Pure paging and segmentation with paging are expected to have the same amount of external fragmentation.

None of the other answers is correct.

Finish Quiz Save For Later