

Week 12: Module Review and Exam Preview

CS211: Programming and Operating Systems

Niall Madden (Niall.Madden@NUIGalway.ie)

Wednesday and Thursday, 5+6 May 2021



**I will teach you in a room.
I will teach you now on Zoom.
I will teach you in your house.
I will teach you with a mouse.
I will teach you here and there.
I will teach you because I care.
So just do your very best.
And do not worry about the rest.**

Source: [sbagley](#) on Twitter

1 Part 1: Module review

2 Part 2: The Exam

- Honesty
- Terms and conditions

3 Part 3: Sample Exam Questions

CS211

Week 12: Module Review and Exam Preview

Start of ...

PART 1: Module Review

Part 1: Module review

The topics we have covered (not necessarily in order) are:

(a) What is an OS?

(b) Computer History: from batch systems to distributed systems.

(5-1) What is a process?

(c) Programming with processes: fork, getpid, getppid.

(d) Interprocess communication with pipe + read + write + signals.

(e) [Threads, including pthread] (not covered in detail)

(f) Scheduling Algorithms. "things happening at once!"

(g) Concurrency; race conditions; Critical Sections; locks; Semaphores;

Part 1: Module review

(h) The dining Philosophers problem

+ Banker's Algorithm for Resource Sharing.

(i) Memory management.

+ Page + Frame.

Part 1: Module review

In C Programming, we had

- (i) Basic ~~structure~~ *structure* : *main()*, *{};*, *#includes,*
 - (ii) if else, etc (*flow of control*)
 - (iii) Loops: for, while, do... while;
 - (iv) Input (printf) and output (scanf) Also *getchar, getc.*
 - (v) Functions, including argument lists, return values, void, call-by-value; call-by-reference, *function prototypes.*
 - (vi) Pointers + Arrays.
- (0): Variables - *int, float, char, char*, -*

Part 1: Module review

- (vii) Strings (ie arrays of chars).
- (viii) Files: `fopen`, `fclose`, reading and writing
is open, `fseek`, `rewind`, ... checking if a file
- (ix) User-defined types `enum`, `struct`, `typedef`
- (x) Dynamic memory allocation: `calloc`, `malloc`, `free`

THE END!!

I hope you have enjoyed CS211, and found it interesting and/or useful.
Thank you for your commitment, collaboration, interest and insights.

CS211
Week 12: Module Review and Exam Preview

END OF PART 1

CS211
Week 12: Module Review and Exam Preview

Start of ...

PART 2: The Exam

Part 2: The Exam

The final assessment for CS211 is a two hour exam during the “summer” sitting. The structure is **different** from previous years.

→ There are **17** questions: some are short, some are very short. You should try to answer **all** of them.

- Questions 1–10 multiple choice questions, each worth **5** marks. They mainly cover C programming, but there are also a few on OS theory. For these questions, it is enough to indicate the correct options. No need to explain your answers.
- Questions 11–16 are on OS theory. The marks vary, but total to 50 marks.
- Question 17 carries zero marks, but asks you to list resources you used in preparing your solution.

The first piece of text on the exam paper reads:

Declaration

*In submitting this work I confirm that it is entirely my own. I acknowledge that I may be invited to online interview if there is any concern in relation to the integrity of my exam submission, and I am aware that any breach will be subject to the **University's Procedures for dealing with breaches of Exam Regulations**.*

Implies in this is that it is **strictly** forbidden to upload the exam paper to anywhere. This will be monitored. ✓

Open Book

This is an OPEN BOOK exam: you may use any online resource you wish, including the lecture notes, text-book, C compiler, etc, providing you list all such resources. You may **not** consult with any person, or share work with anyone else. During the period of the examination you are not permitted to communicate with anyone else, other than the invigilator, about the examination.

- The exam will be posted to Blackboard at 16.30, Monday 31 May 2021.
- During the exam, I can be contacted at Niall.Madden@NUIGalway.ie. Since the exam mostly takes place outside of office hours, don't contact anyone else about it.
- **Duration:** 2 Hours + 30 minutes for uploading So the deadline is 19.00, **Monday 31 May 2021.**
- **LENS:** If you need extra time, let me know in advance. The extra time to complete the exam will be added onto the end (sorry!). Indicate the amount of extra time in the uploaded PDF.
- You can hand-write, or type your answers. For Questions 1-10, you can also annotated the PDF. Convert your scan to a single PDF file (or two) and upload the PDF to EXAM section of the 2021-CS211 module on Blackboard.
- If you encounter difficulties uploading your PDF, email a copy to Niall (Niall.Madden@NUIGalway.ie) by the deadline before continuing to upload it to Blackboard.

Always give ID number on every page!!

If writing in long-hand: number the pages.

If typing, also number pages.

CS211

Week 12: Module Review and Exam Preview

END OF PART 2

CS211

Week 12: Module Review and Exam Preview

Start of ...

PART 3: Sample Exam Questions

(Revised Version)

Solutions to Q 1 - 10 in previous session
Q 11 - 16 in this session

Part 3: Sample Exam Questions



CS211

SAMPLE PAPER (version 2), ^{MAY} APRIL 2021

- Q1. [5 MARKS] In Operating Systems theory, which of the following best describes **a process**?
- (a) ~~A unit of memory.~~
 - (b) ~~A person who is in favour of Cess.~~
 - (c) A program that is being executed.

Answer : (c)

Part 3: Sample Exam Questions

Q2. [5 MARKS] Match these computing devices with the era in which they were first produced.

Computer systems:

- (i) Multitasking (ii) Multiprogramming
(iii) Batch (iv) Micro (personal) computers

Eras:

- (a) 1950's (b) 1960's (c) 1970's (d) 1980's

Batch \leftrightarrow 1950 | Multiprogramming \leftrightarrow 1960s
Multitasking \leftrightarrow 1970s | PCs \leftrightarrow 1980s.

Q3. [5 MARKS] Which of the following common C functions are defined in the `stdio.h` header (select all that apply).

- (a) `scanf()` (b) `rand()` (c) `fclose()` (d) `pipe()`

`stdio.h` = "Standard input/output".

Answer: (a) `scanf`, (c) `fclose`.

Part 3: Sample Exam Questions

Q4. [5 MARKS] Which of the following are data types in C? (Select all that apply)?

(a) ~~function~~

(b) double

(c) int

(d) ~~string~~

(e) char

Q4 Answer: b, c & e

Q5. [5 MARKS] Which of the following gives the correct syntax for a for-loop in C: (Note: the answer may be case-sensitive).

(a) `for(int i=0; i++; i<=10)`

(b) `for(int i=0; i<=10; i++)`

(c) `for(int i=0, i++, i<=10)`

(d) `FOR(int i=0, i<=10, i++)`

(e) `for(int i=0; i=10; i++)`

(f) `for(int i=0; i==10; i++)`

(a): continuation & increment are swapped. → wrong

(c) = (a) and has commas. → wrong

(d) Has FOR instead of for, so wrong

(e) Not illegal, but won't do anything

(f) Same as (e). wrong

WRONG
WRONG

(b)
is
correct.

Part 3: Sample Exam Questions

- Q6. [5 MARKS] Suppose a program defined `float x=0.001234`. Determine which one of the lines of code below would produce the output:

`x=0.00123`

- (a) `printf("x=%f\n", x);`
- (b) `printf("x=%e\n", x);`
- (c) `printf("x=%.3f\n", x);`
- (d) `printf("x=%0.3e\n", x);`
- (e) `printf("x=%.5f\n", x);`
- (f) `printf("x=%0.5e\n", x);`

(e) is correct because it is in floating point format, with 5 digits after decimal place.

0.001230
1.23e-3
0.001
1.230e-3
1.23000e-3

- Q7. [5 MARKS] Which of the following functions returns the PID of the calling process's parent?

- (a) `getpid()`
- (b) `getppid()`
- (c) `getpartentpid()`
- (d) `getpartentppid()`

(a) - gets my pid (so wrong)
(b) is correct
(c) & (d) are not functions.

Part 3: Sample Exam Questions

in read mode.

Q8. [5 MARKS] A C program contains the line
`fileptr = fopen("Results.csv", "r");`
Which of the following statements is true?

- (a) A new file named `Results.csv` is created. If it already existed, its contents are overwritten. *wrong*
- (b) A file called `Results.csv` is opened in read mode. If it does not exist, a `NULL` pointer is returned. *correct*
- (c) A file called `Results.csv` is opened in write mode. If it already exists, a `NULL` pointer is returned. *X*

Ans: (b)

Q9. [5 MARKS] Which of the following are conditions that need to hold in order for **deadlock** to occur. (Select all that apply).

- (a) Hold and wait
- (b) ~~Hide and seek~~
- (c) ~~Mutual Delusion~~
- (d) No preemption

*Also: circular wait
mutual exclusion (ie can't share)*

Part 3: Sample Exam Questions

Q10. [5 MARKS] In Operating Systems theory, which of the following best describes a **Page Fault**?

- (a) ~~A torn page in a your computer's manual.~~
- (b) An exception raised when a process attempts to accesses a memory page that is not currently loaded into memory. *correct*
- (c) ~~An error generated when a process attempts to read file that is corrupted (due to hardware failure).~~

Correct answer (b) :

Part 3: Sample Exam Questions

Q11. [15 MARKS] The table below shows the CPU burst times (in seconds) of four processes submitted in the given order, all at time $t = 0$. Calculate the average wait and response times for each of ~~the~~ the following CPU scheduling algorithms.

- First-Come-First-Served (FCFS),
- Shortest-Job-First (SJF),
- Round-Robin (RR) with a time quantum of $q = 4$ seconds.

Process	P_1	P_2	P_3	P_4
Burst Time	20	4	10	2

	$t=0$	$t=20$	$t=24$	$t=34$	$t=36$	Total Wait time
P_1	Running	Done				0
P_2	Wait	Running	Done			20
P_3	Wait	Wait	Runs	Done		24
P_4	Wait	Wait	Waits	Run	Done	34

So Ave wait time is $\frac{1}{4}(0 + 20 + 24 + 34) = \frac{1}{4}(78) = 19.5$ s.

Ave Response time is $\frac{1}{4}(0 + 20 + 24 + 34) = 19.5$ s.

Part 3: Sample Exam Questions

Q11. [15 MARKS] The table below shows the CPU burst times (in seconds) of four processes submitted in the given order, all at time $t = 0$. Calculate the average wait and response times for each of Consider the following CPU scheduling algorithms.

- (i) ~~First Come First Served (FCFS)~~,
- (ii) Shortest-Job-First (SJF),
- (iii) ~~Round Robin (RR) with a time quantum of $q = 4$ seconds.~~

Process	P_1	P_2	P_3	P_4
Burst Time	20	4	10	2

SJF : order is P_4, P_2, P_3, P_1

	$t=0$	$t=2$	$t=6$	$t=16$	$t=36$	Total wait time
P_1	Wait	Wait	Wait	Run	Done	16
P_2	Wait	Run	Done			2
P_3	Wait	Wait	Run	Done		6
P_4	Runs	Done				0

Ave Wait Time is $\frac{1}{4}(16+2+6+0) = 6$ | Ave Resp Time is also 6.

Part 3: Sample Exam Questions

R = Run, 0 = Done
W = Wait

Q11. [15 MARKS] The table below shows the CPU burst times (in seconds) of four processes submitted in the given order, all at time $t = 0$. Calculate the average wait and response times for each of Consider the following CPU scheduling algorithms.

- (i) First-Come-First-Served (FCFS),
- (ii) Shortest-Job-First (SJF),
- (iii) Round-Robin (RR) with a time quantum of $q = 4$ seconds.

Process	P_1	P_2	P_3	P_4
Burst Time	20	4	10	2

t	0	4	8	12	14	18	22	26	28	32	36	Total Wait Times	Response time
P_1	R	W	W	W	R	W	R	W	R	R	0	18	0
P_2	W	R	D									4	4
P_3	W	W	R	W	W	R	W	R	D			18	8
P_4	W	W	W	R	D							12	12

Ave Wait time is $\frac{1}{4}(18 + 4 + 18 + 12) = 13$ s.

Ave Resp time is $\frac{1}{4}(0 + 4 + 8 + 12) = 6$ s.

Part 3: Sample Exam Questions

Q12. [4 MARKS] Give an example of a scenario, similar to Q11, where four processes are submitted at the same time, with no two having the same burst times, and yet each of FCFS, SJF, and RR give average wait and response times.

FCFS and SJF will always give the same results if the procs arrive in order of size (shortest first).

Also, if the q is longer than any of the burst times, then RR is the same as FCFS.

So, eg with $q=4$, suppose we have

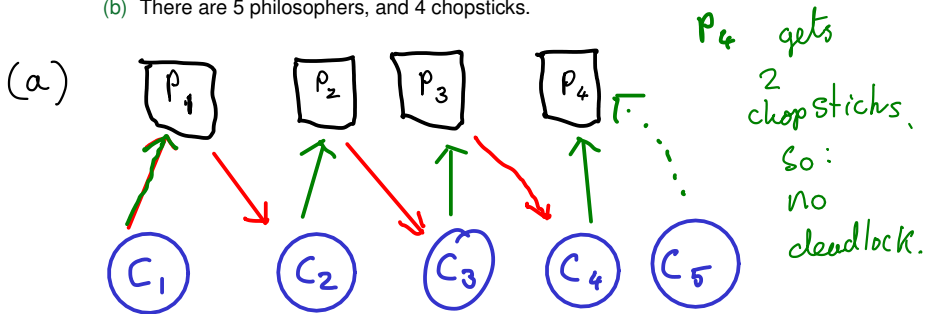
Procs	P_1	P_2	P_3	P_4
Burst time	1	2	3	4.

Part 3: Sample Exam Questions

Q13. [8 MARKS] Recall the “dining philosophers problem” as a model for process synchronisation.

Give a **resource allocation graph** for each of the following scenarios, and determine if **deadlock** is possible.

- (a) There are 4 philosophers, and 5 chopsticks.
- (b) There are 5 philosophers, and 4 chopsticks.



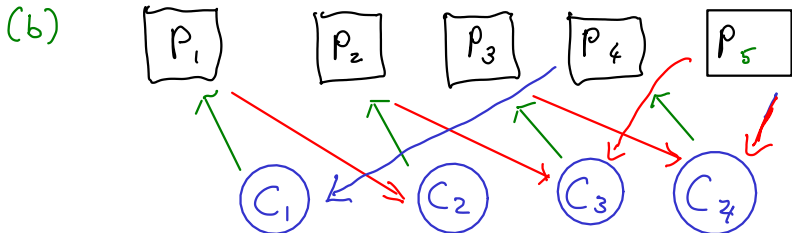
Note: we have broken the circular wait condition.

Part 3: Sample Exam Questions

Q13. [8 MARKS] Recall the “dining philosophers problem” as a model for process synchronisation.

Give a **resource allocation graph** for each of the following scenarios, and determine if **deadlock** is possible.

- (a) There are 4 philosophers, and 5 chopsticks.
- (b) There are 5 philosophers, and 4 chopsticks.



There is a cycle: $P_1 \rightarrow C_2 \rightarrow P_2 \rightarrow C_3 \rightarrow P_3 \rightarrow C_4 \rightarrow P_4 \rightarrow C_1 \rightarrow P_1$

So **Deadlock!**

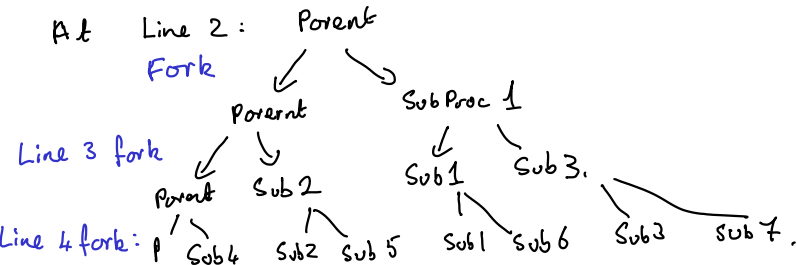
Part 3: Sample Exam Questions

- Q14. [8 MARKS] Consider the following two snippets of C code. For each, determine the number of lines of output generated, and provide an explanation.

Example 1

```
1 int main(void){
  fork();
3  fork();
  fork();
5  printf("Here is a line of output.\n");
  return(0);
7 }
```

← 8 procs
run this



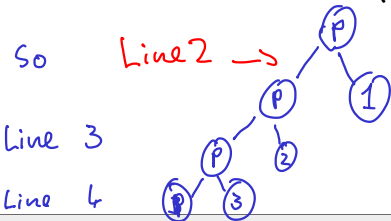
Part 3: Sample Exam Questions

Example 2

```
1 int main(void){  
2     if (fork() > 0)  
3         if (fork() > 0)  
4             fork();  
5     printf("Here is a line of output.\n");  
6     return(0);  
7 }
```

Note: fork returns 0 to the subprocess, and child's PID to parent.

So $(\text{fork}() > 0)$ creates a subprocess and evals as true for parent false for subprocess



At the end, there are 2 procs running, so 4 lines of output.

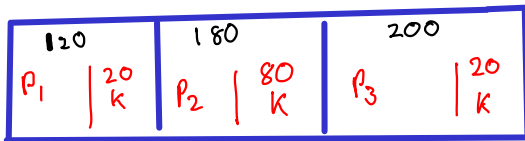
Part 3: Sample Exam Questions

Q15. [8 MARKS] Suppose a system has three free memory partitions, of size **120k**, **180k** and **200k**, in that order.

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

- ▶ P_1 , which requires 100k,
- ▶ P_2 , which requires 100k,
- ▶ P_3 , which requires 180k,
- ▶ P_4 which requires 120k.

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



P_4 is unallocated.

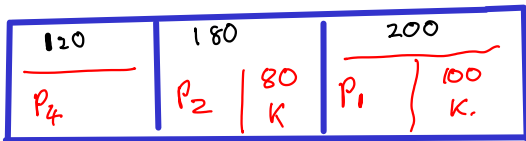
Part 3: Sample Exam Questions

Q15. [8 MARKS] Suppose a system has three free memory partitions, of size **120k**, **180k** and **200k**, in that order.

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

- ▶ P_1 , which requires 100k,
- ▶ P_2 , which requires 100k,
- ▶ P_3 , which requires 180k,
- ▶ P_4 which requires 120k.

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



P_3
not
allocated

Part 3: Sample Exam Questions

Q16. [10 MARKS] Suppose a system has $F = 4$ available frames, and executes a process that has the following "page reference string"

{1, 2, 3, 4, 1, 2, 5, 1, 2, 3, 4, 5}.

Example
in week 11.

Calculate the **Frame Hit rate** using the following algorithms.

- Least recently Used (LRU).
- Optimal.

	1	2	3	4	1	2	5	1	2	3	4	5
F ₁	<u>1</u>	0	1	1	<u>1</u>	1	1	<u>1</u>	1	1	①	5
F ₂		<u>2</u>	2	2	2	<u>2</u>	2	2	<u>2</u>	2	2	2
F ₃			<u>3</u>	3	3	③	<u>5</u>	5	5	⑤	<u>4</u>	4
F ₄				<u>4</u>	4	4	4	4	④	<u>3</u>	3	3
					HIT	HIT		H	H			

There are 4 hits. so Hit Rate is $\frac{4}{12} = \frac{1}{3}$
(or 33.33%)

Part 3: Sample Exam Questions [Added after video]

Q16. [10 MARKS] Suppose a system has $F = 4$ available frames, and executes a process that has the following "page reference string"

{1, 2, 3, 4, 1, 2, 5, 1, 2, 3, 4, 5}.

Optimal:
remove the page that will be referenced least soon

Calculate the **Frame Hit rate** using the following algorithms.

- (a) Least recently Used (LRU).
- (b) Optimal.

	1	2	3	4	1	2	5	1	2	3	4	5
F ₁	<u>1</u>	1	1	1	<u>1</u>	1	1	<u>1</u>	1	①	<u>4</u>	4
F ₂		<u>2</u>	2	2	2	<u>2</u>	2	2	<u>2</u>	2	2	2
F ₃			<u>3</u>	3	3	3	3	3	3	<u>3</u>	3	3
F ₄				<u>4</u>	4	④	<u>5</u>	5	5	5	5	<u>5</u>

Need a free frame. #4 will be used least soon, so replace it.

Need to replace a frame with 4. Any of 1, 2 or 3 will do

Part 3: Sample Exam Questions

Q16. [10 MARKS] Suppose a system has $F = 4$ available frames, and executes a process that has the following "page reference string"

{1, 2, 3, 4, 1, 2, 5, 1, 2, 3, 4, 5}.

Calculate the **Frame Hit rate** using the following algorithms.

- (a) Least recently Used (LRU).
 (b) Optimal.

(b) Continuing with "Optimal". Now need to count Hits

	1	2	3	4	1	2	5	1	2	3	4	5
F ₁	<u>1</u>	0	1	1	<u>1</u>	1	1	<u>1</u>	1	<u>1</u>	<u>4</u>	4
F ₂		<u>2</u>	2	2	2	<u>2</u>	2	2	<u>2</u>	2	2	2
F ₃			<u>3</u>	3	3	3	3	3	3	<u>3</u>	3	3
F ₄				<u>4</u>	4	<u>4</u>	<u>5</u>	5	5	5	5	<u>5</u>
					Hit	Hit		Hit	Hit	Hit		Hit.

Hit Rate: $\frac{6}{12}$ (ie 50%).

Part 3: Sample Exam Questions

Q17. [0 MARKS] What resources, other than lectures notes, did you use when answering the questions on this exam?