



**THE UNIVERSITY
OF QUEENSLAND**
A U S T R A L I A

This exam paper must not be removed from the venue

Venue _____
 Seat Number _____
 Student Number

--	--	--	--	--	--	--	--

 Family Name _____
 First Name _____

School of Information Technology and Electrical Engineering EXAMINATION

Semester One Final Examinations, 2021

CSSE2310/CSSE7231 Computer Systems Principles and Programming

This paper is for St Lucia Campus students.

Examination Duration: 120 minutes

Reading Time: 10 minutes

Exam Conditions:

This is an Open Book examination
 During reading time - write only on the rough paper provided
 This examination paper will be released to the Library

Materials Permitted In The Exam Venue:
(No electronic aids are permitted e.g. laptops, phones)

Calculator - Casio FX82 series or UQ approved (labelled)

Materials To Be Supplied To Students:

None

Instructions To Students:

There are 75 marks total. Questions carry the number of marks indicated. Answer all questions on this exam paper.

Additional exam materials (e.g. answer booklets, rough paper) will be provided upon request.

For Examiner Use Only

Question	Mark
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	

Total _____/75_____

QUESTION 1**(11 marks – 1 mark each)**

(a) Write a shell command to show the names (only) of all files and directories in the "sample" subdirectory of the user's home directory (which is NOT the current directory)

(b) Write a shell command to copy all .c files in the /tmp directory to the "data" subdirectory of the current directory

(c) Write a shell command to make a new subdirectory of the current directory called "breakfast"

(d) Write a shell command to compile and link an executable program called `doit` that uses pthreads and the maths library from a C file called `threadcalc.c`. All input and output files are in the current working directory.

(e) Write a shell command to bring any changes from the subversion repository into the working directory (the current directory).

(f) Write a shell command to show all lines in a file called "colours" (in the current directory) that contain the string of characters "green".

QUESTION 1 continued

(g) Write a shell command to show all lines in a file called "services" in the /etc directory that do not contain the string of characters "tcp".

(h) Write a shell command that finds all lines in the file "shells" (in the /etc directory) that contain the string "bin" (without the quotes) but not string "usr" (without the quotes) and appends those lines to a file called "foo" in the "tmp" subdirectory of the current directory.

(i) Write a shell command that counts the number of lines in a file called "friends" (in the current directory) that contain the word "Shaheen".

(j) Write a shell command that counts all lines in the file "fstab" (in the /etc directory) that contain the string "ext4" (without the quotes) and writes that count value to a file called "ext4.count" in the "tmp" subdirectory of the current directory.

(k) Write a shell command that, in the current directory, creates a symbolic link called "sh" that points to "/usr/bin/sh"

Q1

/11

QUESTION 2**(6 marks – 1 mark each)**Write C declarations to declare a variable “`bar`” as ...

(a) An array of five non-negative whole numbers

(b) A pointer to a function which takes two character pointers as parameters and returns an integer

(c) An integer which is potentially modified by multiple threads

(d) A high precision floating point number

(e) An array of three character pointers (strings)

(f) A pointer to a function that is suitable for use as a thread function

Q2
/6

QUESTION 3**(4 marks – 1 mark for each address)**

Consider a system with 4KiB pages and a process with the following page table. All numbers are in base 10.

Page Number	Frame Number
0	-
1	41
2	42
3	-
4	44
5	22
...	
40	23
41	24
42	43
43	112
...	
16,380	132
16,381	99

For each of the following virtual addresses, what is the corresponding physical address? Write your answer in base 10. If accessing the virtual address would result in a segmentation fault, then write "SEGFault". If there is insufficient information in the page table (i.e. the page number is not listed) then write "UNKNOWN".

Virtual Address	Physical Address
8,192	
16,380	
167,936	
25,000	

Q3
/4

QUESTION 4**(6 marks – 1 mark each)**

`moss.labs.eait.uq.edu.au` supports 43-bit physical addresses and 48-bit virtual addresses and uses a four level page table. Pages are 4KiB in size. Page table entries are 8 bytes each.

A process uses the following virtual address range (all numbers are in base 10):

1200 MiB starting at address 0

(a) What is the maximum memory space for a process on this system (in GiB)?

(b) How much memory (in KiB) would be needed to store the page table for this process?

(c) If the process doubled its memory usage (to 2400 MiB starting at address 0), how much memory (in KiB) would now be needed to store the page table?

QUESTION 4 continued

(d) If the original process doubled its memory usage (to 2400MiB total - 1200MiB at address 0, and 1200MiB at the highest possible addresses), how much memory (in KiB) would now be needed to store the page table?

(e) For the original memory usage (1200MiB starting at address 0), if the system used a two level page table, how much memory (in KiB) would be needed to store the page table?

(f) For the original memory usage (1200MiB starting at address 0), if the system used a single level page table, how much memory (in MiB) would be needed to store the page table?

QUESTION 5**(5 marks – 1 mark each)**

Consider the following program. Assume that all system calls succeed.

```
#include <stdio.h>
#include <unistd.h>
#include <sys/wait.h>

int main(int argc, char** argv) {
    pid_t pid;
    printf("1\n");
    if(!fork()) {
        printf("2\n");
    } else {
        printf("3\n");
        if(pid=fork()) {
            printf("4\n");
            waitpid(pid, NULL, 0);
        } else {
            printf("5\n");
        }
        printf("6\n");
    }
    printf("7\n");
    fork();
    printf("8\n");
    return 0;
}
```

(a) How many processes are created by the execution of this program (including the initial process)?

(b) What is the maximum number of processes that could be running or runnable at any point?

(c) If standard output buffers are never flushed until a process exits, how many times will the character “3” be output?

(d) What is the minimum number of lines of text that this program will output when run?

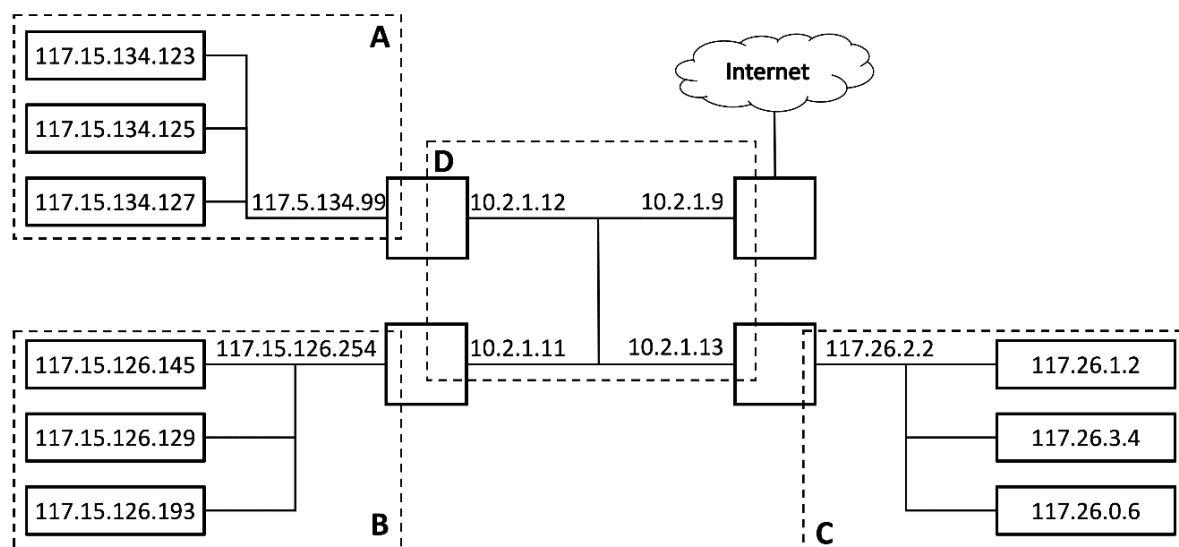
(e) What is the maximum number of lines of text that this program will output when run?

Q5

/5

QUESTION 6**(9 marks)**

Consider the following network (assume that all networks are as small as possible):



- (a) Fill in the netmask, broadcast address and CIDR for each of the networks (A to D):
(6 marks)

Network	Netmask	Broadcast Address	CIDR
A			
B			
C			
D			

- (b) Fill in the detail for the whole network shown above (as it would appear to the rest of the internet). (2 marks)

Netmask	Broadcast Address	CIDR

- (c) How many unused addresses are there in **network B**? (i.e. how many additional machines could be added to that network) (1 mark)

QUESTION 7**(8 marks)**

Consider a "unix" file system where

- blocks are 16 KiB
- block pointers are 8 bytes
- inodes have
 - 7 direct pointers
 - 3 single indirect pointers
 - 3 double indirect pointers

(a) What is the maximum file size on this file system? Express your answer in KiB. (2 marks)

(b) What is the maximum file size that can be stored without using a double indirect pointer? Express your answer in KiB. (2 marks)

QUESTION 7 continued

(c) Assuming only the inode is cached in memory, how many blocks must be accessed to read bytes 200,000 to 220,000 (inclusive) from a file into memory? (2 marks)

(d) If one of the double indirect pointers was replaced by a triple indirect pointer, what is now the maximum file size on the system? Express your answer in KiB. (2 marks)

Q7
/8

QUESTION 8**(5 marks – 1 mark each)**

Consider the following directory listing (generated by “ls -ali”):

```
$ ls -ali
total 2164
5067385 drwxrwxr-x   3 bob      admin    4096 Oct 17 11:08 .
   75112 drwxr-xr-x 100 root      root     4096 Oct 17 10:43 ..
5061551 -rwxr-xr-x    1 bob      admin    8192 Oct 18 10:23 admin
5067391 lrwxrwxrwx    1 bob      users      4 Oct 17 10:59 backup -> data
5066833 drwxr-x---    5 bob      admin    4096 Oct 17 10:59 data
5067386 -rwxr-xr-x    1 alice     admin 2190232 Oct 17 11:00 generate
5068104 lrwxrwxrwx    1 alice     admin      8 Oct 17 11:09 install -> generate
5067407 -rw-r--r--    1 bob      admin    279 Oct 17 11:05 doc2
5067392 -rw-r--r--    1 alice     admin    279 Oct 18 11:05 docs
```

Users "alice" and "bob" are the only members of the "admin" group. All users on the system are members of the "users" group.

- (a) Within this directory, what command can bob run to prevent alice from being able to read the “doc2” file but still allow other users of the system to read that file?

- (b) Within this directory, what command can bob run to allow all users other than alice to list the contents of the data subdirectory?

- (c) How many subdirectories does the data subdirectory have?

(Question continues on next page)

QUESTION 8 continued

At some later point in time, the same command (`ls -ali`) includes two additional lines:

```
5067407 -rw-r--r--  2 bob    admin   279 Oct 17 11:05 docx
5067408 lrwxrwxrwx  1 alice  users    4 Oct 19 11:08 docz -> docs
```

The contents of `doc2`, `docs` and `docx` are identical.

(d) What command could have been used to create `docx`?

(e) What command could have been used to create `docz`?

QUESTION 9**(4 marks – 1 mark each)**

A system has the following ordinary users and groups (and no others):

User	Groups
alice	staff, users, project
bob	staff, users
carol	admin, users
dave	admin, users, project
eve	project, users
fred	admin, staff, users

Consider the following directory listing:

```
-r--r-x--- 1 alice project 138856 Oct 17 10:46 file1
--w-r-xrwx 1 dave  admin  2190232 Oct 17 11:49 file2
---xr--r-x 1 eve   users   24000 Oct 17 09:14 file3
-rwx---r-x 1 fred  staff   123456 Oct 17 11:50 file4
```

List all of the users who are allowed to do the following.

(a) Read from `file1`(b) Write to `file2`(c) Run `file3`(d) Read from `file4`

Q9
/4

QUESTION 10**(17 marks)**

Consider a file `listen.c` that has the following contents:

```
#include <sys/types.h>
#include <sys/socket.h>
#include <netdb.h>
#include <string.h>

int listen_on_port(char* port) {
    struct addrinfo* ai = 0;
    struct addrinfo hints;
    memset(&hints, 0, sizeof(struct addrinfo));
    hints.ai_family = AF_INET;
    hints.ai_socktype = SOCK_STREAM;
    getaddrinfo("localhost", port, &hints, &ai);
    int server_fd = socket(AF_INET, SOCK_STREAM, 0);
    bind(server_fd, (struct sockaddr*)ai->ai_addr,
        sizeof(struct sockaddr));
    listen(server_fd, 1);
    return server_fd;
}
```

and the file `listen.h` that contains the following function prototype:

```
int listen_on_port(char* port);
```

Write a file called `netshell.c` (12 marks) and an associated `Makefile` (5 marks) that will create an executable `netshell` that has the following behaviour.

```
./netshell port
```

The program will listen on the specified port for connections. When a connection is received, the program will create a new process and execute a shell ("`/bin/bash`"). The `stdin` of this shell must come from the network connection, and similarly the `stdout` and `stderr` of the shell must be sent to the same network connection. The effect of this program is to create a network shell.

- Your program should keep listening for connections, and will not terminate unless killed with `CTRL-C`, `SIGKILL/SIGINT` or similar.
- Your program must make use of the `listen_on_port()` function from `listen.c`.
- Your program must check that sufficient command line arguments are supplied, and if not, then print a message to standard error and exit with a non-zero exit status.
- No other errors need to be checked for. You may assume that all system calls succeed.
- You must not use the `system()` or `popen()` library functions.

(Question continues on next page)

QUESTION 10 (continued)

An example execution of `netshell` may be as follows:

```
$ ./netshell 45000
```

This will listen on localhost TCP port 45000.

If a connecting program (such as `netcat`) sends the string `"ls"` followed by a newline, it will receive the output of the `"ls"` program back over that connection. Similarly, a connecting program sending the string `"echo hello >&2"` would receive the string `"hello"` back (`netshell` sends the shell `stderr` back over the connection).

Your program and Makefile must be written to build and run on `moss.labs.eait.uq.edu.au` for marking purposes.

Your Makefile must have the following characteristics:

- Compilation and linking must be separate steps, i.e. `netshell` is built from object files.
- Compilation must include the C compiler flags `-std=gnu99` and `-pedantic`. (Others can be included if you wish.)
- (Re)compilation of a C file will only happen if it or a local header file it includes has been modified since the last time it was compiled

Running `make` without any arguments will build `netshell` (if required)

- Running `make clean` will remove `netshell` and all object files

Security note – Never do this in real life, as it presents a major security risk granting any connecting process the same privileges as the user running 'netshell'. This task is for educational purposes only.

Please write your `netshell.c` and Makefile into the spaces provided on the following pages.

(Question continues on next page)

QUESTION 10 (continued)

(a) netshell.c

[illegible]

QUESTION 10 (continued)

(a) netshell.c (continued)

[illegible]

QUESTION 10 (continued)

(a) netshell.c (continued)

[illegible]

QUESTION 10 (continued)

(b) Makefile

This image shows a single page of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

(b) Makefile (continued)

[illegible]

Q10

This page is provided for rough or additional working and will not be marked unless an earlier answer explicitly refers to a continuation of an answer on this page.