THE UNIVERSITY
OF QUEENSLAND
AUSTRALIA

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 Two Final Examinations, 2017

CSSE2310/7231 Computer Systems Principles and Programming

This paper is for St Lucia Campus students.

Examination Duration: 180 minutes

Reading Time: 10 minutes

Exam Conditions:

This is a Central Examination

This is a Closed Book Examination - specified materials permitted

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)

Calculators - Casio FX82 series or UQ approved (labelled)

One A4 sheet of handwritten or typed notes double sided is permitted

Materials To Be Supplied To Students:

Instructions To Students:

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

Attempt all questions

For Examiner Use Only

Question	Mark

Total   _____

Question 1) Write shell commands to do the following: [5 marks (1 each)]

A) Show the names and permission information for all files in the current directory.

> ls -la

B) Count the number of `.c` files in the current directory.

> ls -l *.c | wc -l

C) There is a file called `bill`. Show the first line (only) of the file which contains the word "chip".

> grep chip bill | head -1

D) Show all the lines of `bill` which contain the both words "chip" and "fish".

> cat bill | grep chip | grep fish

E) A file `dict` contains one word per line. Output the last four words (in alphabetical order).

> cat dict | tail -4 | sort

Question 2) Write shell commands to do the following: [10 marks (2 each)]

A) Output the name of the first directory which will be searched for executables. (Hint : is a separator.)

```
echo $PATH | cut -d ':' -f 1
```

B) Copy all files in the current directory ending in `.pdf` to `/dest` and add `old_` to the front of their name. Eg: `A.pdf` would be copied to `/dest/old_A.pdf`.

```
for file in *.pdf
do cp $file ./dest/old_$file
dome;
```

C)

```
UID        PID  PPID  C STIME TTY          TIME CMD
uqjfenw1 58653 58652  0 23:09 pts/92   00:00:00 -bash
uqjfenw1 58654 58653  0 23:10 pts/92   00:00:00 ./do_things
uqjfenw1 58897 58653  0 23:10 pts/92   00:00:00 vim
uqjfenw1 58899 58654  0 23:10 pts/92   00:00:00 mark <defunct>
uqjfenw1 58702 58653  0 23:10 pts/92   00:00:00 sleep
```

Remove `mark` from the process list above.

```
kill -9 58654
```

D) Display the changes which have been made to `thing.c` and `fred.c` since the last commit.

```
svn diff thing.c fred.c
```

E) Delete `old.h` (in the current directory) from the svn repository.

```
svn rm old.h
svn commit
```

Question 3) Give the output for the following code fragments.   [6 marks (1 each)]

A)
```
int x=4, y=9;
printf("%d", x|y);
```


B)
```
int x=4, y=9;
printf("%d", x&y);
```


C)
```
int a=13, b=5;
printf("%d", a/b);
```


D)
```
int x=5, y=17;
for (int i=0;i<5;++i) {
    x=x^y;
}
printf("%d %d %d", i, x, y);
```

E)

```c
int a=0, b=0, c=5;
do {
    a++;
    switch (a%3) {
    case -1: b+=5; break;
    case 0: b++; break;
    case 1: b+=3; continue;
    case 2: b+=4; break;
    }
    a++;
} while (a<4);
printf("%d %d", a, b);
```

F)

```c
int a=1, b=3;
if ((a--) && (b++)) {
    b++;
}
if (b<a) {
    b+=100;
}
if (!a) {
    b+=10;
}
printf("%d %d", a, b);
```

Question 4) Suppose a system uses 32bit virtual addresses, 34bit physical addresses and a two level page table. Pages are 16KiB pages (16384Bytes) each. (All addresses are given in base 10).
A process has the following (complete) page table: [5 marks]

Page	Frame
102	204
103	203
204	206
205	205
206	11001
408	34
409	1901
64998	654
64999	102
65000	103

page = 1687755/16384 = 103.012390137 = 103
offset = 1687755%16384 = 203
phy = 203*16384 + 203 = 3326155

A) What physical address correspond to the following virtual addresses: [2 marks]

3358719 | 3391487 |

page = 3358719/16384 = 204.999938965 = 204
offset = 3358719%16384 = 16383
phy = 206*16384 + 16383 = 3391487

3375102 | 3375102 |

page = 3375102/16384 = 205.99987793 = 205
offset = 3375102%16384 = 16382
phy = 205*16384 + 16382 = 3375102

3375103 | 3375103 |

page = 3375103/16384 = 205.999938965 = 205
offset = 3375103%16384 = 16383
phy = 205*16384 + 16383 = 3375103

1687755 | 3326155 |

B) Give the smallest address above 6701265 which will cause a segmentation fault if accessed by this process?   [1 marks]

page = 6701265/16384 = 409.012756348 = 409
if page is 410, it will cause a seg fault
410*16384 = 6717440

C) Assuming that the TLB starts out empty, how many pages in the page table must be accessed to read from each of the virtual addressses given above? Assume the reads happen immediately after each other.

[2 marks]

3358719

2

page = 3358719/16384 = 204.999938965 = 204

3375102

2

page = 3375102/16384 = 205.99987793 = 205

3375103

0

page = 3375103/16384 = 205.999938965 = 205

1687755

2

page = 1687755/16384 = 103.012390137 = 103

Question 5)   [7 marks]

    Consider the following directory listing:

```
7602238 drwxr-xr-x 2 me sw     4.0K Sep  7 18:19 .
7602237 drwxr-xr-x 6 me sw     4.0K Sep  7 18:17 ..
7602242 -rw-r--r-- 2 me staff  91K Sep  7 18:13 crash
7602245 -r--r--r-- 1 me staff  19K Sep  7 18:16 gg
7602242 -rw-r--r-- 2 me staff  91K Sep  7 18:13 hamster
7602243 -rw-r-xr-- 2 me staff 931K Sep  7 18:13 nyancat
7602247 -rw-r--r-- 1 me staff 5.0K Sep  7 18:19 papers
7602244 lrwxrwxrwx 1 me staff    5 Sep  7 18:15 thing -> crash
```

A) How many other subdirectories does our parent directory have?   [1 mark]

       6 - 2 (the . and ..) - 1 (current dir) = 3

B) Who can run the following: [2 marks]

`./thing`

      no one

`./nyancat`

   group staff, not me, not everyone

C) If the current directory were removed. How much data (file contents) would be lost?   [2 marks]

    the files gg (7602245), hamster (same as crash, 7602242) as well as papers (7602247) will be
    lost. That's because nyancat has 2 references but the other one is not in the same directory,
    and thing is just a soft link to crash so doesn't count.
    That makes 19K + 91K + 5K = 115 KB

D)

```
drw---xr-x 2 me    staff   4096 Sep  7 18:54 .
drwxr-xr-x 6 root  staff   4096 Sep  7 18:17 ..
-rw-r--r-- 1 me    staff   4266 Sep  7 18:54 document.py
-rwxr-xr-x 1 me    staff 953112 Sep  7 18:52 prog
-rwxr--r-- 1 me    staff 798728 Sep  7 18:53 prog3
```

Attempting to execute `./prog3` fails (as me). Why?   [2 marks]

    You can't access anything in a directory without x, even if you know the directory's name.

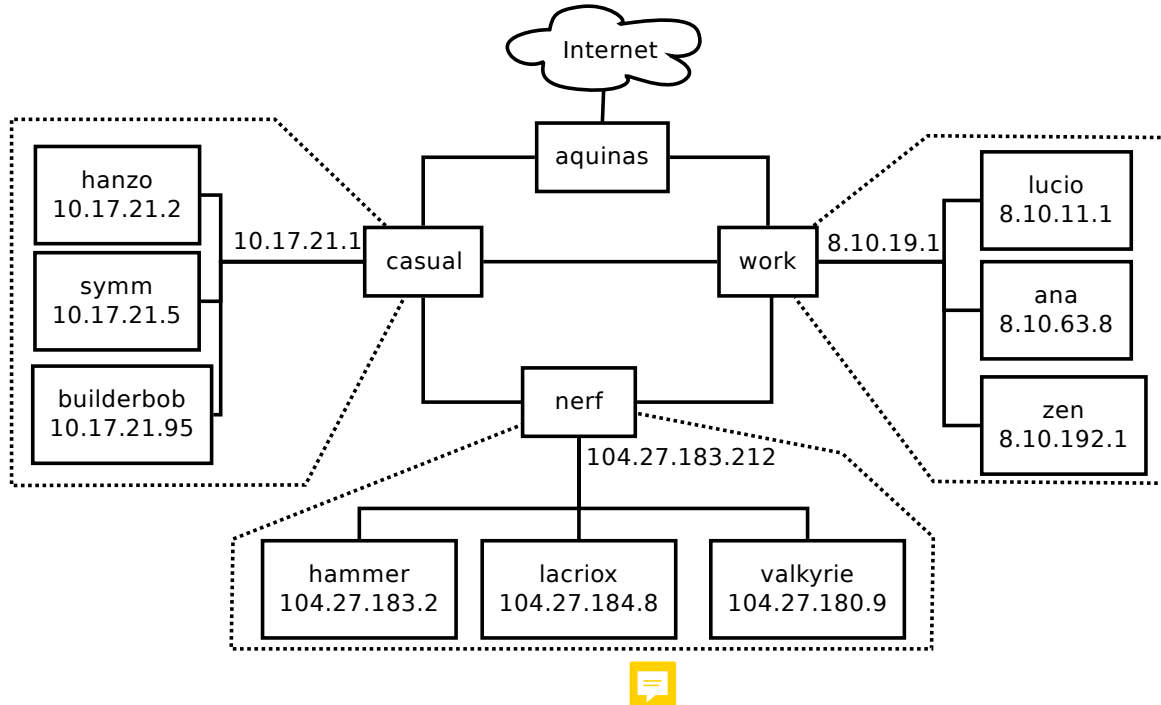Question 6)   [17 marks]

A) Consider the following network: [10 marks]

i) Complete the table for each part of the network (use the smallest possible networks).

Network	Broadcast address	Netmask	CIDR network notation
casual	10.17.21.127	255.255.255.128	10.17.21.0/25
work	8.10.255.255	255.255.0.0	8.10.0.0/16
nerf	104.27.251.15	255.255.240.0	104.27.176.0/20

ii) Fill in the minimum number of addresses of each type used to send each message type from A to B:

Type	A	B	Destination MACs	Source IPs	Destination IPs
TCP segment	aquinas	hammer	3	1	1
UDP datagram	aquinas	hammer	3	1	1
Ethernet frame	zen	work	1	0	0
TCP segment	symm	uq.edu.au	**leave blank**	1	1

B) Network monitoring shows that connections to `lucio` are preceeded by UDP traffic to `zen`. Give a likely explanation for this. [2 marks]

> I think it's because zen is a DNS server containing the information about lucio's IP, so incoming messages ask zen using UDP what lucio's IP is before they talk to lucio.

C) An organisation owns the 4.51.19.0/17 block and they wish to break this range into networks each containing 64 machines. What is the maximum number of such networks they could have? [2 marks]

Number of network addresses = 2^(32-17) = 32768
Number of host addresses = 32768 - 2 = 32766
Each subnetwork should hold 64 machines, including broadcast and network addresses this is 66.
Requires 7 bits per subnetwork (does not fit 2^6).
Number of subnetworks =floor(32766 / (2^7)) = 255 networks

D) Which layer of the network stack are each of the following terms primarily associated with?[3marks]

ssh	application
port	transport
MAC	link & network
routing	network
putty	application
ethernet	link

Question 7) Consider a "unix" filesystem where: [6 marks]

- All i-nodes are cached in RAM

- blocks are 2KiB.   2*2^10 = 2048

                                              2048/8 = 256

- block pointers are 8bytes.

- i-nodes have 5 direct pointers, 1 indirect pointer and 2 double indirect pointers.

        5*2048

A) How many blocks would need to be accessed to (independently) read the following bytes from a file: [2 marks]

10200	1
10205	1
51025	2
102050	2

Space that direct pointers take => 5 * 2048 = 10240 bytes

+ (1 * 256 * 2048) = 534528 bytes so the 1 indirect pointers have a range from 10240 (incl) to 534527 bytes

+ (2 * 256 ^2 * 2048) = 268969984 bytes. Range = 534528 (incl) to 268969984 bytes (counting bytes start at 0).

B) What is the maximum file size for this filesystem?   [2 marks]

        2 * (5 + 256 + 2*256^2) = 262666 KiB

C) If the structure was changed to remove a double indirect and replace it with a triple indirect, what would be the new maximum file size?   [2 marks]
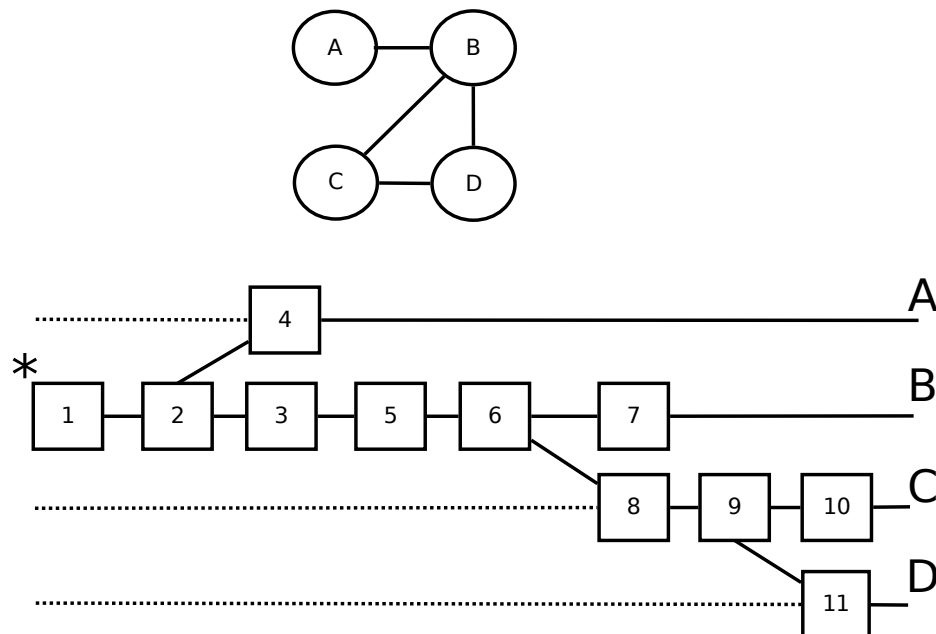
    New modified quantities (5 direct, 1 single indirect, 1 double indirect, 1 triple indirect)
    Max size = 2 KiB * [(5) + (1)*256 + (1)*256^2 + (1)*256^3] = 33686026 KiB
    Difference = 33686026 - 262666 = 33423360 KiB

Question 8)   [7 marks]

A) The top diagram shows 4 processes (running programs A, B, C, D) and pipes between them. The lower diagram (starting with process marked *****) will create this arrangement (each row in the diagram represents a single process). Fill in the boxes to indicate which operations need to be performed when. Use operations from the following list (some may be needed more than once): `kill(A)`, `exec(A)`, `exec(B)`, `exec(C)`, `exec(D)`, `fork`, `pipeAB`, `pipeBC`, `pipeCD`, `pipeBD`, `exit`, `socket`.   [4 marks]

1. pipeAB

2. fork()

3. pipeBC

4. exec(A)

5. pipeBD

6. fork()

7. exec(B)

8. pipeCD

9. fork()

10. exec(C)

11. exec(D)

B) The current process has a child with pid=451.   Write a code fragment to print out its exit status.   [3 marks]

```
int status;
waitpid(451, &status, 0);
if (WIFEXITED(status)) {
        printf("%d\n", WEXITSTATUS(status));
}
```

Question 9) In the following, you may omit `#include`s. Where you are asked to write functions, you can also write helper functions if you need them. You may assume that any functions you write have been prototyped at the top of the file (so you don't need to worry about which order you write them in). You may assume that system calls succeed.

In later parts of this question, you may make use of functions described in earlier parts (and you may assume they are implemented correctly). [22 marks]

A) Write a function called `split_string`, which takes a string and a separator character and returns an array of strings consisting of parts of the original string which were separated by the given character (the final element in the array will be a null pointer). **For this part, you must not call any standard string functions.**

For example: `split_string("http://example.com/fred",'/')`
would return an array of `"http:"`, `""`, `"example.com"`, `"fred"`, `(char*)0`. [5 marks]

```
char** split_string(const char* line, char sep) {
```

B) Write a function `run_part` which takes an array of strings representing `argv` and two file descriptors (`a` and `b`). When the function is finished, a new process will be running with `a` as `stdin` and `b` as `stdout`. You are not permitted to call `system()` or `popen()`.   [3 marks]

```
void run_part(const char** argv, int a, int b) {
```

C) Write a function called `run_cmd` which takes in a string and executes it (arguments are separated by *single* spaces). As well as the string to run, it should also take pointers to two integers. These are the file descriptors for sending to the command and reading from the command respectively. So `run_cmd("ls -l", &i1, &i2)` would run `ls` with the argument `"-l"`. If the string contains any `'|'` characters, then each part should be executed separately and the standard output of the left program fed into the standard input of the right program.

For example: `run_cmd("ls -l|sort|cat", &i1, &i2)`

Note: there should be no spaces around `'|'`.   [9 marks]

```
void run_cmd(const char* cmd, int* to, int* from) {
```

D) Write a **program** which takes two commandline arguments (a command to execute and a string of input to pass to the executing command). The program should execute the command, send the string to it (followed by `'\n'`), then print any output from the command along with a count of how many characters the command output. For example: `./a.out 'cat|tr t T' attic` would output:

```
aTTic
(6 chars)
```

If an incorrect number of arguments is passed, print nothing and exit with status 1.   [5 marks]

Question 10) Write a function `total` which takes a file descriptor for a connected socket. It should read ten space separated integers from the socket and then send the total to the socket and then disconnect. You may omit `#include`s and you may assume all system calls succeed.   [5 marks]

```
void total(int fd) {
```

Question 11) In all of the following you may omit `#include`s. You may assume that all system calls succeed. You do not need to consider integer overflow. You may write additional functions to call.

Write a function called `integrate` which takes the following parameters:

`f`: a pointer to the function to be integrated (which takes a `double` and returns a `double`)

`a`: the lower bound

`b`: the upper bound

`N`: the number of slices

`T`: how many threads to use

The result will be:

$$(b - a)/N * \sum_{k=0}^{N-1} f(a + k * (b - a)/N)$$

That is, add up

$$(b - a)/N * f(a + k * (b - a)/N)$$

for each $0 <= k <= N - 1$.

Note that your function must make non-trivial use of threads. You may assume that N is a multiple of T. [10 marks]

END OF EXAMINATION

Working space

This page left blank

Working space

This page left blank

You may detach this sheet.
Do not record answers on this sheet.

Example addresses:

broadcast	130.102.17.255
netmask	255.255.255.0
CIDR	130.102.17.0/24

%c	character
%d	integer
%u	unsigned integer
%lf	double (scanf)
%f	double (printf)
%p	void pointer
%ld	long integer

```
grep [-v] [ $ ^ . * ]
ls [-ladi]
ps [-ef]
sort [-r -k]
uniq [-c]
cat
head [-]
tail [-]
cut [-f -d]
wc [-l]
diff
svn
chmod
ln [-s]
rm [-rf]
mkdir
rmdir
cp [-r]
mv
vim/nano
less
```

Layers
link
application
onion
physical
network
gooey caramel
transport

KiB =	2^{10} Bytes
MiB =	2^{20} Bytes
GiB =	2^{30} Bytes

[ ]	array access
.	member selection
$->$	follow and select
$++--$	
sizeof	
$\sim$	bitwise not
!	logical not
+ -	unary forms
&	address of
*	follow
( )	cast
* / %	
+ -	binary forms
$> < <= >=$	
== !=	
&	bitwise AND
^	bitwise XOR
|	bitwise OR
&&	logical AND
||	logical OR
? :	ternary operator
= += -= ...	
,	comma operator
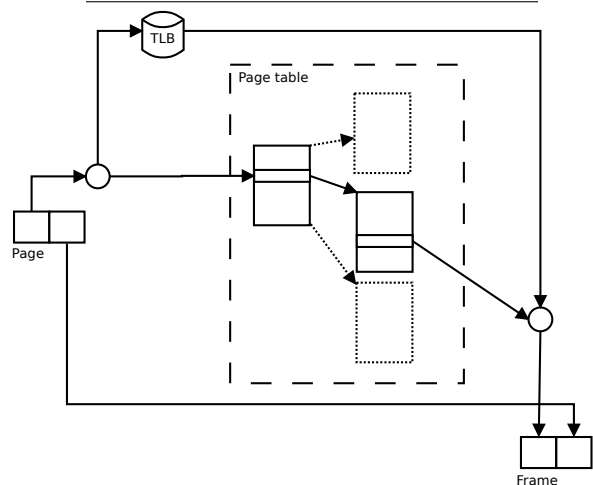
This page intentionally blank