DUBLIN INSTITUTE OF TECHNOLOGY

DT228 BSc. (Honours) Degree in Computer Science

Year 2

DT282 BSc. (Honours) Degree in Computer Science (International)

Year 2

WINTER EXAMINATIONS 2018/2019

OPERATING SYSTEMS 2 [CMPU2017]

MR. DENIS MANLEY

FRIDAY 14TH JANUARY   9.30 A.M. – 11.30 A.M.

TWO HOURS

ANSWER QUESTION 1 AND ANY TWO OTHERS.

QUESTION 1 IS WORTH 40 MARKS, ALL THE REST ARE WORTH 30.

1

a) Given the following arrival times and CPU time for 4 processes determine the average turnaround time for:

     a round robin schedule algorithm with a time slice of 8 *ms*
     the shortest remaining time.   **(8 marks)**

Arrival Time	0	1	2	3
Job	A	B	C	D
CUP cycle time	8	4	9	5

b) Distinguish, using a suitable example, between a single integer pointer and a double integer pointer.   **(4 marks)**

c) Explain, using a simple example, the purpose of the linux *wait()* command and how it achieves its purpose.   **(8 marks)**

d) Explain, using suitable examples, the steps required to add a node to an ordered link list.   **(10 marks)**

e) Will the following code add a node from the tail of a queue? Clearly explain your reasoning.   **(10 marks)**

```c
char dequeue(QueueNode** headPtr, QueueNode** tailPtr)
{
  QueueNode* newPtr;
  newPtr= malloc(sizeof(QueueNode));

  if (newPtr != NULL) {
    newPtr->data = value;
    newPtr->nextPtr = NULL;

    if (*headPtr == NULL ) {
      *headPtr = newPtr;
    }
    else {
      (*tailPtr)->nextPtr = newPtr;
    }

    *tailPtr = newPtr;
  }
  else {
    printf("%c not inserted. No memory available.\n", value);
  }
}
```

a) Identify the *four* conditions necessary for *Deadlock* to occur.   **(4 Marks)**

b) Deadlock detection / recovery is another way of handling deadlock; explain, using an example the steps involved to reduce a deadlock detection directed resource graphs.
(8 Marks)

c) A consequence of deadlock prevention is starvation. What is starvation and how operating systems prevent starvation?   **(4 marks)**

d) A second method of dealing with deadlock is deadlock prevention. Using deadlock prevention Bankers algorithm answer the following:

	Allocation				Max				Need				Available			
													3	2	2	1
	A	B	C	D	A	B	C	D	A	B	C	D	A	B	C	D
P0	4	0	0	1	7	0	2	1								
P1	1	1	0	0	1	6	5	0								
P2	1	0	4	5	3	3	4	6								
P3	0	4	2	1	1	5	6	2								
P4	0	3	1	2	2	4	3	2								

(a.)   How many resources of type A, B, C and D are there?   **(3 marks)**
(b.)   What are the contents of the *Need* column?   **(3 marks)**
(c.)   Is the system in a safe state? Provide reasoning for your answer.

(4 marks)

(d.)   If a request from process P2 arrives for additional resources of {0, 2, 0, 0}, can the Bankers algorithm grant the request immediately? Provide reasoning for your answer.   **(4 marks)**

3

a) Distinguish between single and multi-threading processes. **(4 marks)**

b) In C a thread is created using the following code:

*int pthread_create(pthread_t *tidp, pthread_attr_t *attr, *start_rtn, void * arg)*

Clearly explain what each of the arguments in the thread create function mean.
(8 marks)

c) Explain, in your own words, the following code: **(10 marks)**

```c
#include<pthread.h>
#include <stdio.h>
int value;
void *my_thread(void *param);

int main (int argc, char *argv[])
{
pthread_t tid; int retcode;

if (argc != 3) {
    fprintf (stderr, "usage: a.out <integer value>\n");
    exit(0);
}


retcode = pthread_create(&tid,NULL,my_thread,argv[1]);


if (retcode != 0) {
            fprintf (stderr, "Unable to create thread\n");
            exit (1);
    }


    pthread_join(tid,NULL);
    printf ("I am the parent: the cube of value passed = %d\n", value);
} //main


void *my_thread(void *param)
    •   {
```

```
        int i = atoi (param);
        printf ("I am the child, I am passed value %d\n", i);
        value = i * i*i;
        pthread_exit(0);
    •   }
```

d) What would be the output of the executable code of the above program, explaining your answer where the following is input at the command prompt: **(6 marks)**

    ../a.out
    ./a.out 5 6
    ./a.out 6 5

e) What would be the two outcomes in the above program if the *pthread_join* command was removed and the command line input was *./a.out 6 5*? Explain the reason for your answer.   **(2 marks)**

4:

a) Explain, using an example, how you would map the logical address of a process to its physical address. **(8 marks)**

b) Describe the purpose of each of the following fields in the page map table of a virtual; memory management system:

      the status field,

      the modified field. **(4 marks)**

c) Page swapping is an essential element of virtual memory: two page swapping algorithms are the *First In First Out (FIFO)* and *Least Recently Used (LRU)* algorithms. Using a demand page system with 3 frames how many page faults will be generated by the following sequence? Clearly show how you arrived at the answer

Reference Sequence = [A, B, A, B, F, D, F, C, G, F, G, B, D, C].

    i.   FIFO

    ii.   LRU

(12 marks)

d) A *cache* improves the speed of processor access to instructions and data. Explain how the cache; main memory and secondary storage combine to achieve virtual memory page swapping. **(6 marks)**

COLLEGE EXAMINATIONS

AMENDMENTS TO EXAMINATION QUESTION PAPER

COURSE REF   VENUE:

SUBJECT: Operating Systems 2

DATE: 14/1/19

TIME: 10:05

SIGNED: Denis

INSTRUCTIONS:

Q1 part 5 (it is ~~delete~~ add a node ...)
and not "add a node ...")
function name is "enqueue"

Question 1 part e:
may I assume the variable "value" is created
as global variable outside the function

COLLEGE EXAMINATIONS

AMENDMENTS TO EXAMINATION QUESTION PAPER

COURSE REF   VENUE:

SUBJECT: Operating Systems 2

DATE: 14/1/19

TIME: 10:05 - 10:20

SIGNED:

INSTRUCTIONS:

Q1 E - stays as is add node.

function name is "enqueue"   should be 3rd parameter char value in function heading

Q1 E -
function ~~and~~ name is QueueNode   QueueNode ~~char~~ void enqueue (Node ** headPtr, Node ** tailPtr, char value)