UNIVERSITY OF LONDON
IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 2000

BEng Honours Degree in Computing Part I
MEng Honours Degrees in Computing Part I
for Internal Students of the Imperial College of Science, Technology and Medicine

This paper is also taken for the relevant examinations for the Associateship of the City and Guilds of London Institute

PAPER C111

OPERATING SYSTEMS I

Monday 15 May 2000, 14:00
Duration: 90 minutes
(Reading time 5 minutes)

Answer THREE questions

Paper contains 4 questions

1 a Describe briefly the effect of the following system calls

i) `fork`

ii) `execve`

and explain how they can be used together.

[5]

b Consider the following Turing program:

```
if not fork() = 0 then
  if not fork() = 0 then
    % ... this is process A ...
  else
    % ... this is process B ...
  end if
else
  if not fork() = 0 then
    % ... this is process C ...
  else
    % ... this is process D ...
  end if
end if
```

Write down the process tree showing the hierarchical relationship between the processes. Describe in words the parent/child/sibling relationships.

[5]

c Explain in a few lines the function of a *process control block (PCB)*. Identify the main parts in a PCB and their purpose.

[5]

d What are the key differences between *processes* and *threads*, and what role does the operating system play in their management?

[5]

2 Consider the following set of processes whose execution time is given in seconds:

Process ID	Execution time (sec)
1	10
2	1
3	2
4	1
5	5

You can assume that all processes arrive at $t = 0$ in the order P1, P2, P3, P4, P5. The operating system supports the following two scheduling algorithms: First-come, first-serve (FCFS) scheduling and round robin (RR) scheduling with time slice $\Delta t = 1$ sec.

a Describe briefly how the two scheduling algorithms operate. What is the key difference between the two scheduling algorithms?

[5]

b For each of the two scheduling algorithms, draw a diagram showing the execution of the processes and calculate the average turnaround and waiting time.

[10]

c Another operating system uses a round robin scheduling algorithm with a single ready queue. Assume that a process appears twice in the ready queue.

i) How will this affect the round robin scheduling?

ii) What are the different consequences of this on single- and multi-processor machines?

[5]

3a Explain briefly the following terms:

i) Critical section
ii) Mutual exclusion
iii) Starvation
iv) Deadlock

[6]

b Show with examples how locks can be used to achieve mutual exclusion and how their use can lead to a deadlock.

[4]

c A small remote valley is divided into a west and east part by a river running through it. A single bridge connects the two sides of the valley. The bridge is strong enough to carry more than one car but it is too narrow to allow cars simultaneously to travel east and west. As a result, cars crossing the bridge in one direction must make sure that no other car is crossing the bridge in the opposite direction.

i) Using semaphores, specify a procedure `car_west` for cars wishing to travel westwards. You can assume that cars wishing to travel eastwards use a similar existing procedure `car_east`. Make sure that you specify how you initialise your data structures.

ii) Explain whether in your algorithm it is possible for starvation to occur, assuming that there are a fixed number of cars in the valley and cars travel regularly from one side of the valley to the other side.

You can use Turing code or similar pseudo code as well as the following data structures:

```
% Counter for cars travelling east
var east_counter : int
% Counter for cars travelling west
var west_counter : int
% Semaphore to protect east_counter
var east_sema     : semaphore
% Semaphore to protect west_counter
var west_sema     : semaphore
% Semaphore to protect bridge
var bridge_sema   : semaphore
```

[10]

4a Describe the following two memory partitioning schemes:

i) fixed memory partitioning

ii) dynamic memory partitioning

and describe how they can lead to

iii) internal memory fragmentation

iv) external memory fragmentation

[6]

b Assume that your operating system and hardware support memory management by paging. The page size is 4 bytes and your system has 32 bytes of memory. The page table looks like this:

page	frame
0	5
1	6
2	1
3	2

Calculate to which physical memory addresses the following virtual memory addresses translate to: 0, 1, 9, 14.

[4]

c Assume another operating system that runs on a computer with 64 MB of memory. The operating system uses the buddy system to handle memory requests. Given the following sequences of memory requests and memory releases, draw a diagram of the memory layout after each request or release and calculate the amount of internal memory fragmentation after each request or release.

- A = Request 10 MB
- B = Request 24 MB
- C = Request 6 MB
- Release B
- Release A
- D = Request 8 MB
- Release C
- Release D

[10]