IMPERIAL COLLEGE LONDON

EE1-09

DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING
EXAMINATIONS 2016

EIE PART I: MEng, BEng and ACGI

Introduction to Computer Architecture and Systems

Friday 3rd June 2016, 10:00AM

Time allowed: 1:00 hour

Corrected Copy

There is ONE Question on this paper

Answer ALL questions.

Any special instructions for invigilators and information for candidates are on page 1.

Examiners responsible

First marker: T.J.W. Clarke
Second marker: C. Bouganis

The Questions

The question is on the next page

1. a) Consider the processes shown in Figure 1.1 with arrival time, duration as shown. In the questions below the sequence X:Y:Z indicates the priorities of X,Y,Z respectively to be 1, 2 and 3 where X,Y,Z are some permutation of A,B,C. Lower numbers represent higher priorities.

i) Calculate the waiting time for process A under priority scheduling with preemption for each of the six cases given by priorities : A:B:C, A:C:B, B:A:C, B:C:A, C:A:B, C:B:A. (This is all possible permutations).

[2]

ii) Calculate the *minimum* waiting time for process A, given it may start at any time in the range 0 ms - 4 ms, with all other system parameters as shown in Figure 1.1, for the same 6 cases, under priority scheduling with preemption.

[2]

iii) An operating system has 10 users, each of which run both interactive and batch processes. It is desired that interactive processes should have the fastest possible response, and that batch processes should run fairly. Propose a scheduling strategy that will implement this.

[2]

b) In the context of memory paging systems, consider the following scenario:

- You have three available frames
- The reference string is: 9-1-3-1-3-2-4-3-6-3

i) Starting with empty frame contents, show the sequence of frame contents after each request and count the number of page faults for the LRU algorithm.

[2]

ii) Contrast the advantages and disadvantages of optimal and LRU page replacement.

[2]

c) i) Write pseudocode showing how a single global variable can be used to force mutually exclusive access to a resource from two processes A and B.

[1]

ii) Show that Peterson's Algorithm, shown in Figure 1.2, ensures that processes A and B can never both be in their critical region at the same time.

[2]

iii) Give one reason why semaphores provide a better solution to mutual exclusion than Peterson's Algorithm.

[1]

d) i) Draw a diagram showing all states and possible transitions of the 7 state process model. Which state(s) can contain a maximum number of processes at any one time, if the total number of processes is unknown. What is this number?

[2]

ii) A disk interrupt happens and an I/O request from process A is completed. What possible state transition(s) within the 7 state model can process A make as the result of this?

[1]

e) Show a block diagram of the typical hardware to implement page-based memory allocation. With reference to this diagram describe the sequence of operations required to implement a memory read by a user process of a location within a page which is swapped out of main memory before the start of the memory read.

[3]

Process	arrival time (ms)	duration (ms)
A	0	4
B	1	2
C	2	3

Figure 1.1: Three processes.

```
Variables:
Interested_A := FALSE
TURN := 'A'

Process A
Interested_A := TRUE;
Turn := 'B';
While (Interested_B and not Turn = 'A');
/* Critical Region */
Interested_A := FALSE;
```

```
Variables:
Interested_B := FALSE

Process B
Interested_B := TRUE;
Turn := 'A';
While (Interested_A and not Turn = 'B');
/* Critical Region */
Interested_B := FALSE;
```

Figure 1.2: Peterson's Algorithm.