

UNIVERSITI MALAYA  
UNIVERSITY OF MALAYA

PEPERIKSAAN IJAZAH SARJANA MUDA TEKNOLOGI MAKLUMAT  
EXAMINATION FOR THE DEGREE OF BACHELOR OF INFORMATION TECHNOLOGY

SESI AKADEMIK 2016/2017 : SEMESTER I  
ACADEMIC SESSION 2016/2017 : SEMESTER I

WXES2113 : Sistem Pengendalian  
Operating Systems

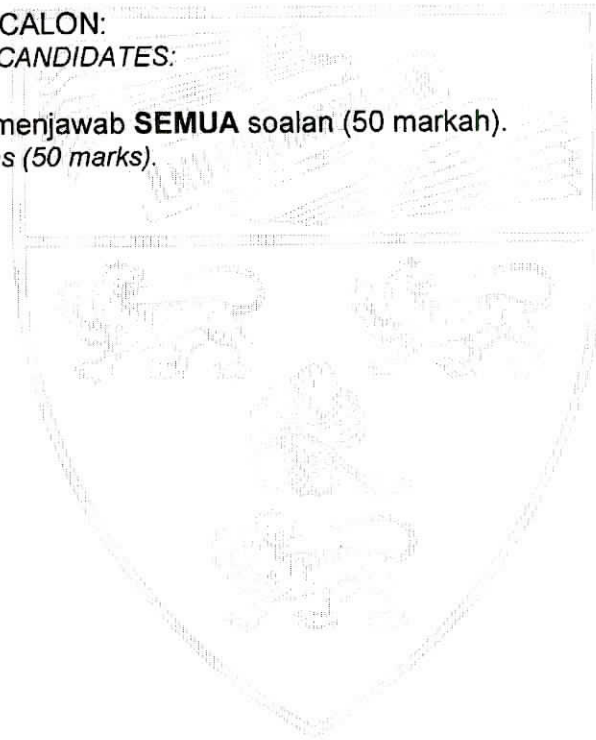
Dis 2016/Jan 2017  
Dec 2016/Jan 2017

Masa: 3 jam  
Time: 3 hours

---

ARAHAN KEPADA CALON:  
INSTRUCTIONS TO CANDIDATES:

Calon dikehendaki menjawab **SEMUA** soalan (50 markah).  
Answer **ALL** questions (50 marks).



(Kertas soalan ini mengandungi 4 soalan dalam 5 halaman yang dicetak)  
(This question paper consists of 4 questions on 5 printed pages)

1. Proses berikut sepatutnya tiba untuk dilaksanakan mengikut masa yang telah dinyatakan. Setiap proses akan berjalan mengikut masa yang telah disenaraikan. Sila gunakan penjadualan tak pintas dalam menjawab soalan dan rangkumkan semua keputusan maklumat yang anda dapati di masa keputusan harus dilakukan.

*Suppose that the following processes arrive for execution at the times indicated. Each process will run for the amount of time listed. In answering the questions, use non-preemptive scheduling, and base all decisions on the information you have at the time the decision must be made.*

<i>Proses</i>	<i>Masa Tiba</i>	<i>Masa Tamat</i>
<i>Process</i>	<i>Arrival Time</i>	<i>Burst Time</i>
<i>P<sub>1</sub></i>	0.0	8
<i>P<sub>2</sub></i>	0.4	4
<i>P<sub>3</sub></i>	1.0	1

- a) Apakah masa purata pusing balik untuk proses ini dengan menggunakan algoritma penjadualan Tiba Dahulu Layan Dahulu (FCFS)?

*What is the average turnaround time for these processes with the First Come First Serve (FCFS) scheduling algorithm?*

(4 markah/marks)

- b) Apakah masa Purata pusing balik untuk proses ini dengan menggunakan algoritma penjadualan Kerja Terpendek Dahulu (SJF)?

*What is the average turnaround time for these processes with the Shortest Job First (SJF) scheduling algorithm?*

(4 markah/marks)

- c) Tugas Terpendek Dahulu (SJF) sepatutnya meningkatkan prestasi, akan tetapi didapati proses  $P_1$  dijalankan pada masa 0 kerana pada masa tersebut tidak diketahui ada dua proses pendek yang akan tiba. Kira purata masa pusing balik jika CPU dibiarkan melahu untuk 1 unit pertama dan seterusnya Kerja Terpendek Dahulu (SJF) digunakan. Adalah diingatkan proses  $P_1$  dan  $P_2$  dalam keadaan menunggu semasa waktu melahu tersebut, oleh itu masa tunggu mungkin akan meningkat. Algoritma ini juga dikenali penjadualan pengetahuan masa depan.

*The SJF algorithm is supposed to improve performance, but notice that we chose to run process  $P_1$  at time 0 because we did not know that two shorter processes would arrive soon. Compute what the average turnaround time will be if the CPU is left idle for the first 1 unit and then SJF scheduling is used. Remember that processes  $P_1$  and  $P_2$  are waiting during this idle time, so their waiting time may increase. This algorithm could be future-knowledge scheduling.*

(4 markah/marks)

## 2. Sila jawab soalan berikut:

*Please answer the following questions. Be brief and to the point:*

- a) Mengapa saiz-saiz halaman seringkali berkuasa 2?

*Why are the page sizes always powers of 2?*

(4 markah/marks)

- b) Namakan **DUA (2)** perbezaan di antara alamat logikal dan alamat fizikal.

*Name **TWO (2)** differences between logical and physical addresses.*

(4 markah/marks)

- c) Andaikan halaman bersaiz 1KB, apakah nombor halaman dan ofset untuk alamat rujukan berikut (dinyatakan dalam nombor perpuluhan):

*Assuming a 1-KB page size, what are the page numbers and offsets for the following address references (provided as decimal numbers):*

- i) 3085
- ii) 42095
- iii) 215201
- iv) 650000
- v) 2000001

(5 markah/marks)

## 3. a) Terangkan perbezaan di antara fragmentasi dalaman dan luaran.

*Explain the difference between internal and external fragmentation.*

(4 markah/marks)

- b) Apakah *thrashing*? Bagaimana kah sistem mengesan *thrashing*? Apabila *thrashing* dikesan, apakah yang boleh dilakukan oleh sistem untuk menghapuskan masalah *thrashing*?

*What is thrashing? How does the system detect thrashing? Once the thrashing is detected, what can the system do to eliminate the problem of thrashing?*

(4 markah/marks)

- c) Pertimbangkan Jadual halaman yang ditunjukkan oleh rajah di bawah adalah merupakan sebuah sistem dengan 12-bit alamat maya dan fizikal berserta 256 bait halaman. Senarai tiga bingkai halaman yang kosong adalah D, E, F (dimana D berada di depan, E di tempat kedua, F berada di akhir senarai).

*Consider the page table shown in the figure below for a system with 12-bit virtual and physical addresses and with 256-byte pages. The list of free page frames is D, E, F (that is, D is at the head of the list, E is second, and F is last).*

Page	Page Frame
0	–
1	2
2	C
3	A
4	–
5	4
6	3
7	–
8	B
9	0

Tukarkan alamat maya berikut kepada alamat-alamat fizikal di dalam bentuk heksadesimal. Semua nombor mesti diberikan dalam bentuk heksadesimal. (sengkan “–” untuk bingkai halaman menunjukkan halaman tiada di dalam memori).

*Convert the following virtual addresses to their equivalent physical addresses in hexadecimal. All numbers are given in hexadecimal. (A dash for a page frame indicates that the page is not in memory).*

- i) 9EF
- ii) 111
- iii) 700
- iv) 0FF

(4 markah/marks)



4. a) Bilangan selinder di dalam cakera = 200 (bernomor 0 hingga 199)  
 Posisi kepala semasa = selinder 23  
 (Nota: Kepala selinder bergerak dari selinder 0 ke silinder 199)  
 Permintaan semasa mengikut tertib ketibaan = 20, 17, 55, 35, 25, 78, 99  
 Tunjukkan penjadualan untuk algoritma penjadualan cakera LOOK bagi permintaan di atas. Gunakan rajah untuk jawapan lebih jelas.

*Total number of cylinders in the disk = 200 (numbered 0 to 199)*

*Current head position = cylinder 23*

*(Note: The head has been moving from cylinder 0 towards cylinder 199)*

*Current requests in order of arrival = 20, 17, 55, 35, 25, 78, 99*

*Show the schedule for LOOK disk scheduling algorithms for the above set of requests.*

*Use a figure to make your answer clearer.*

(5 markah/marks)

- b) Jelaskan **TIGA (3)** keadaan di mana *blocking I/O* boleh digunakan. Jelaskan tiga keadaan dimana *non-blocking I/O* boleh digunakan. Kenapa tidak melaksanakan *non-blocking I/O* dan adakan proses *busy-wait* sehingga.

*Describe **THREE (3)** circumstances under which blocking I/O should be used. Describe three circumstances under which non-blocking I/O should be used. Why not just implement non-blocking I/O and have processes busy-wait until their devices are ready?*

(8 markah/marks)

**TAMAT**  
**END**