Xi’an Jiaotong-Liverpool University

PAPER CODE	EXAMINER	DEPARTMENT	TEL
CPT 104	Gabriela Mogos Dongyao Jia	Department of Computing	1515

2022/23 SEMESTER 2 – Open Book Final Exam

BACHELOR DEGREE – Year 2

Operating Systems Concepts

TIME ALLOWED: 2 hours

INSTRUCTIONS TO CANDIDATES

1. Total marks available are 100, accounting for 80% of the overall module marks.
2. Answer all FOUR questions.
3. The number in the column on the right indicates the marks for each question.
4. Relevant and clear steps should be included in the answers.
5. The university approved calculator - Casio FS82ES/83ES can be used.
6. All the answers must be in English in the answer script provided.

QUESTION I. Fundamentals (44 marks)

1. Can the **Priority-based scheduling** schemes result in starvation? If so, how might you fix this? Explain. (4 marks)

2. Give **two** positive and **two** negative effects of increasing the **page size**. Explain. (12 marks)

3. How do **base register** and **limit register** help in protecting against illegal memory access? (4 marks)

4. If you have to design and program a **web browser** (which is a complex application), what would you choose to use multiple processes, multiple threads, or both? Explain your answer. (6 marks)
Discuss **two** advantages of your choice. (6 marks)

5. **Redundant Array of Independent Disks** or **RAID** is a set of physical disk drives viewed by the operating system as a single logical drive. Give **two** advantages and **two** disadvantages of RAID. Explain. (12 marks)

QUESTION II. CPU scheduling, Memory management, Disk scheduling

(32 marks)

1. Consider a *Real-Time System* in which there are three processes. Their arrival time, period and execution time are as follows:

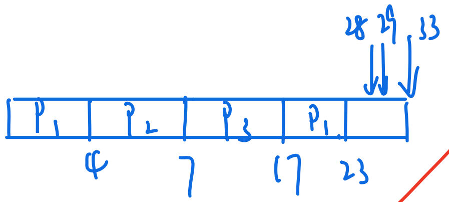
Process	Arrival time	Execution time	Deadline
P1	0	10	33
P2	4	3	28
P3	5	10	29

Calculate total utilization of CPU. $\frac{10}{33} + \frac{3}{28} + \frac{10}{29} \approx 75.5\%$ (2 marks)

We assume that all three processes are released at time 0. Explain the **Earliest Deadline First Scheduling Algorithm** of the processes. (4 marks)

Show the processes on timing diagram. (2 marks)

Earliest-deadline-first (EDF) scheduling dynamically assigns priorities according to deadline. The earlier the deadline, the higher the priority; the later the deadline, the lower the priority. Under the EDF policy, when a process becomes runnable, it must announce its deadline requirements to the system. Priorities may have to be adjusted to reflect the deadline of the newly runnable process. Note how this differs from rate-monotonic scheduling, where priorities are fixed.



2. Calculate the number of page faults for the following sequence of page references (each element in the sequence represents a page number) using the **First-In, First-Out (FIFO) algorithm** with frame size of **3**.

1 2 3 2 1 5 2 1 6 2 5 6 3 1 3 6 1 2 4 3

(6 marks)

Page references	1	2	3	2	1	5	2	1	6	2	5	6	3	1	3	6	1	2	4	3
	1	1	1			2		3	5	1	6		2	5		3		1	6	2
		2	2			3		5	1	6	2		5	3		1		6	2	4
			3			5		1	6	2	5		3	1		6		2	4	3

the total page faults are 14

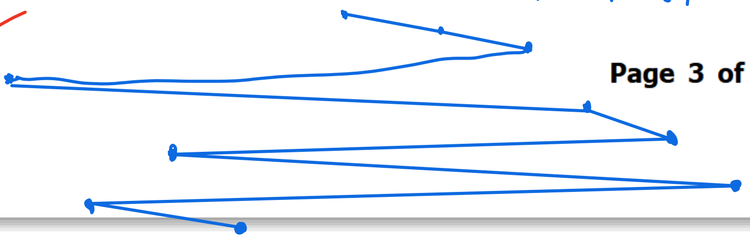
3. Consider a disk queue with I/O requests on the following cylinders in their arriving order:

44, 20, 95, 4, 50, 52, 47, 61, 87, 25

We assume a disk with 100 tracks and the disk head is initially located at track 50.

Write the sequence in which requested tracks are serviced using the **Shortest Seek Time First (SSTF) algorithm** and calculate the **total head movement** (in number of cylinders) incurred while servicing these requests. (6 marks)

44 20 95 4 50 52 47 61 87 25



$2+5+3+17+26+8+70+5+16$

$=152$

4. A paging scheme uses a **Translation Lookaside Buffer** (TLB). A **TLB** access takes 20 ns, and a main memory access takes 50 ns. What is the effective access time (in ns) if the Translation Lookaside buffer TLB hit ratio is 70% and there is no page fault? (6 marks)

$(20+50)\times 70\% + (20+50+50)\times(1-70\%)$

$= 49+36$

$= 85$ ns

5. A program has been divided into five modules. Their lengths and base addresses are stored in the segment table, as depicted in the following space:

Segment number	Length	Base address
0	300	4000
1	600	1000
2	500	2700
3	900	1800
4	1000	2500

What will be the physical memory address for the following logical addresses?

Show the physical memory mapping for the segments. (6 marks)

Segment number s	Offset d
1	550
3	908
4	670

Segment 1: $550 < 600$ $\therefore$ $1000+550 = 1550$

Segment 3: $908 > 900$ request generated is Invalid, trap will be produced.

Segment 4: $670 < 1000$ $\therefore$ $2500+670 = 3170$

QUESTION III. Resource allocation (12 marks)

Consider a system with the following information.

Available

R1	R2	R3
1	5	2

Process	Max			Allocation		
	R1	R2	R3	R1	R2	R3
P1	0	0	1	0	0	1
P2	1	7	5	1	0	0
P3	2	3	5	1	3	5
P4	0	6	5	0	6	3

(1) Is this system currently in a safe or unsafe state? Why? Explain. (6 marks)

(2) If a request from P2 arrives for (0, 5, 0), can that request be safely granted immediately? Explain the answer. (6 marks)

(1)

Require

	R_1	R_2	R_3
P_1	0	0	0
P_2	0	7	5
P_3	1	0	0
P_4	0	0	2

Safe state. By safety Algorithm, all Finish[i] is true.

(2). Yes. Because ① (0,5,0) < (1,5,2) in Available table. ② 检查 Safe state.

(0,5,0) < (0,7,5)

QUESTION IV. Operating System in C Language (12 marks)

The following code implements a **simple scenario of a barbershop**, which includes a barber process and multiple customer processes:

1) If there is an empty chair, a customer can sit down and wait for the barber to cut their hair.
2) If there are no empty chairs, the customer leaves.
3) When the barber wakes up, if there are customers waiting, the barber prepares to cut their hair.

```
int waiting = 0;
int CHAIRS = N;
semaphore customers = 0;
semaphore barbers = 0;
semaphore mutex = 1;
```

barber process:	consumer process:

process barber(){
while(true){
P(customers);
P(mutex);
waiting--;
V(barbers);
V(mutex);
cut_hair(); //cutting hair
}
}
	process customer_i(){
	P(mutex);
	if(waiting < CHAIRS){
	waiting++;
	V(customers);
	V(mutex);
	P(barbers);
	get_haircut(); //receiving service
	}else{
	V(mutex);
	}
	}

(1) What semaphores are used in this code？ What are the implication of the initial value of each semaphore？ (4 marks)

Customers: there are no customers initially. So the barber process is block.

barbers: the barber doesn't wake up. so the customers are block waiting for barbers to wake up.

mutex: there isn't mutex lock initially.

(2) What would happen if "waiting--" and "waiting++" were placed after V(mutex)? (4 marks)

① The "waiting--" and "waiting++" will happen at the same time, which will cause a rare condition.

②. logical and synchronization errors.

(3) What would happen if the value of CHAIRS is set to 0 in this code? (4 marks)

There will be no chairs. So every customer come and leave, which will result in the barber process will be blocked in the P(customers).

END OF EXAM PAPER