ชื่อ..... รหัสนักศึกษา.....

PRINCE OF SONGKLA UNIVERSITY

FACULTY OF ENGINEERING

Midterm Examination: 1st Semester | Academic Year: 2010

Date: 3 August 2010 | Time: 13:30-16:30

Subject: 240-306 / 241-304 Computer Operating Systems | Room: S203, หัวหุ่นยนต์, S817

Instruction:

- Closed books, Closed notes, No calculator, No computer or laptop
- Please write your name and student id on all pages. There are 14 pages.
- This examination has 11 questions. Please answer all questions. Your answer can be in Thai.
- The total score is 300.
- Definitions of some terms are given on the last page

NOTE that I can only grade what I can read. If I cannot read your name or your id, you will not get the score.

คำสั่ง

- ห้ามนำหนังสือ หรือ เอกสารใดๆ หรือเครื่องคิดเลข หรือเครื่องคอมพิวเตอร์ เข้าห้องสอบ
- กรุณาเขียนชื่อและรหัสนักศึกษาบนข้อสอบทุกหน้า ข้อสอบมีทั้งหมด 14 หน้า
- ข้อสอบมี 11 ข้อ กรุณาตอบทุกข้อ คุณสามารถตอบเป็นภาษาไทยได้
- คะแนนเต็ม 300
- คำนิยามส่วนหนึ่งได้ให้ไว้ในข้อสอบหน้าสุดท้าย

อาจารย์จะสามารถตรวจได้เฉพาะคำตอบที่อาจารย์อ่านออกเท่านั้น หากอาจารย์อ่านคำตอบคุณไม่ออก หรือ อ่านชื่อและรหัสนักศึกษาของคุณไม่ออก คุณจะไม่ได้คะแนน

ทุจริตในการสอบ โทษขั้นต่ำคือ ปรับตกในรายวิชาที่ทุจริต และพักการเรียน 1 ภาคการศึกษา

ชื่อ..... รหัสนักศึกษา.....

Question 1: (30 points) [20 minutes] Under what circumstances would a user be better off using a time-sharing system rather than a PC or a single-user workstation?

ภายใต้เงื่อนไขใดที่ผู้ใช้ควรจะใช้ระบบไทม์แชริง ดีกว่า ระบบคอมพิวเตอร์ส่วนบุคคลหรือระบบที่มีผู้ใช้คนเดียว

ชื่อ..... รหัสนักศึกษา.....

Question 2: (20 points) [10 minutes] What would be the output from the following problem at LINE C and LINE P? Explain your answer

ผลลัพธ์ที่ LINE C และ LINE P คืออะไร จงอธิบายคำตอบของท่าน

```
#include <pthread.h>
#include <stdio.h>
int value = 30;
void *runner(void *param);

int main(int argc, char *argv[]) {
  pthread_t tid;
  pthread_attr_t attr;
  int pid;
  pid = fork();
  if (pid == 0) {
    pthread_attr_init(&attr);
    pthread_create(&tid,&attr,runner,NULL);
    pthread_join(tid,NULL);
    printf("CHILD: value = %d\n",value);  /* LINE C */
  }
  else if(pid > 0) {
    wait(NULL);
    printf("PARENT: value = %d\n",value);  /* LINE P */
  }
}

void *runner(void *param) {
  value = 15;
  pthread_exit(0);
}
```

ชื่อ..... รหัสนักศึกษา.....

Question 3: (30 points) [15 minutes] Select TRUE or FALSE and give your explanation if your answer is FALSE

จงเลือกว่าข้อความต่อไปนี้ ถูก หรือ ผิด หากคำตอบของคุณคือผิดให้อธิบายด้วย

TRUE FALSE	System calls enable device controllers to inform the CPU that it has finished its operation. ซิสเต็มคอลทำให้ตัวควบคุมอุปกรณ์บอกกับหน่วยประมวลผลว่าตนเองนั้นทำงานเสร็จแล้ว ....
TRUE FALSE	A thread library provides the programmer with an API (Application Programming Interface) for creating and managing threads. เธรดไลบรารีมีเอพีไอให้ผู้เขียนโปรแกรมใช้ในการสร้างและจัดการกับเธรด ....
TRUE FALSE	FCFS scheduling algorithm could result in starvation. อัลกอริทึมแบบ FCFS อาจจะทำให้เกิดปัญหาการอดอยากได้ ....
TRUE FALSE	Busy waiting means that a process is waiting for a condition to be satisfied in a tight loop without relinquishing the processor. บีซีเวททิงแปลว่าโพรเซสวนลูปรอเงื่อนไขหนึ่งโดยไม่ได้ปล่อยการถือครองหน่วยประมวลผล ....

ชื่อ..... รหัสนักศึกษา.....

Question 3 (continue)

TRUE FALSE	A CPU scheduling algorithm determines an order for the execution of its scheduled processes. Given n processes to be scheduled on one processor, there are n possible different schedules. อัลกอริทึมซีพียูสเก็ตดูลลิงกำหนดลำดับการประมวลผลของโพรเซส หากมี n โพรเซสที่จะต้องจัดลำดับการทำงานบนหน่วยประมวลผลหนึ่งตัวแล้ว จะมีลำดับการทำงานที่แตกต่างกันได้ทั้งหมด n แบบ ....
TRUE FALSE	Round-Robin treats all jobs equally so short jobs will be able to leave the system faster since they will finish first. เราน์ดโรบินมองทุกงานเท่าเทียมกันดังนั้นงานสั้นๆจะสามารถออกจากระบบได้อย่างรวดเร็วเพราะมันทำงานเสร็จก่อน ....

Question 4: (15 points) [5 minutes] Fill in the blank จงเติมคำในช่องว่าง

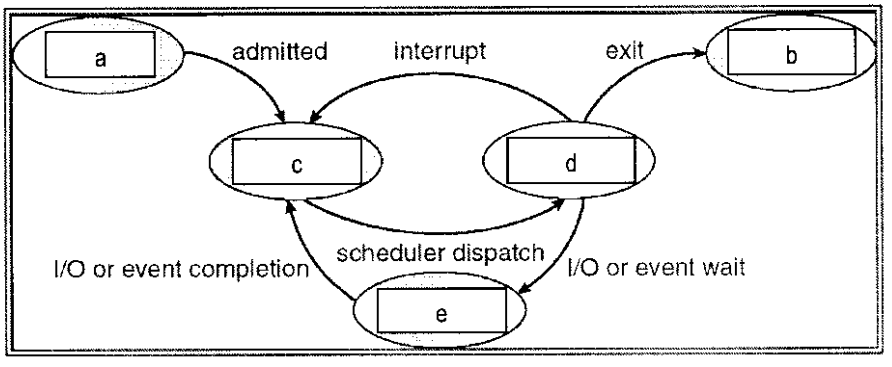

Diagram of process state

a. _____ b. _____ c. _____

d. _____ e. _____

ชื่อ..... รหัสนักศึกษา.....

Question 5: (20 points) [15 minutes] When a process creates a new process using the fork() operation, which of the following state is shared between the parent process and the child process? Explain your answer

เมื่อโพรเซสถูกสร้างขึ้นมาโดยใช้คำสั่ง fork() สิ่งใดต่อไปนี้จะมีการใช้งานร่วมกันระหว่างโพรเซสแม่และโพรเซสลูก อธิบายคำตอบของท่านด้วย

a) stack

b) shared memory segments

c) local variables on the process memory segments

d) global variables on the process memory segments

ชื่อ..... รหัสนักศึกษา.....

Question 6: [30 minutes] Consider the following set of processes, with the length of the CPU burst given in milliseconds. The processes are assumed to have arrived in the order P1, P2, P3, P4, and P5.

Process	Burst Time	Arrival time
P1	7	0
P2	1	1
P3	2	2
P4	1	2
P5	5	6

จากข้อมูลของโพรเซสทั้งห้าตัวที่กำหนดให้ข้างต้น สมมุติให้โพรเซสทั้งหมดเข้ามาในระบบตามลำดับ P1, P2, P3, P4, และ P5

a) (40 points) draw four Gantt charts that illustrate the execution of these processes using the following scheduling algorithms: First-Come-First-Served, Shortest-Job-First (both preemptive and non-preemptive), and Round-Robin (time quantum = 3). Use FCFS to break tie.
จงวาด Gantt charts เพื่อแสดงการทำงานของโพรเซสแต่ละตัว เมื่อเราใช้ First-Come-First-Served, Shortest-Job-First (ทั้งแบบ preemptive และ non-preemptive) และ Round-Robin (time quantum = 3) หากมีตัวเลือกสองตัวให้เลือกทำงานโพรเซสที่เข้ามาก่อนเสมอ

First-Come-First-Served

Shortest-Job-First (preemptive)

Shortest-Job-First (non-preemptive)

Round-Robin (time quantum = 3)

b) (25 points) What is the waiting time of each process for each of these scheduling algorithms? จงกรอกข้อมูลค่า waiting time ของโพรเซสแต่ละตัว เมื่อเราใช้สเก็ตดูลเลอร์แต่ละวิธี

algorithm	Waiting time (milliseconds)					Average waiting time (milliseconds)
	P1	P2	P3	P4	P5	
First-come-First-Served						
Shortest-Job-First (preemptive)						
Shortest-Job-First (non-preemptive)						
Round-Robin						

Question 7: (20 points) [15 minutes] Let S and Q be two semaphores initialized to 1. What is wrong with the following pseudo code? How would you fix it?

กำหนดให้ S และ Q เป็นเซมมะฟอร์ทั้งคู่มีค่าเริ่มต้นที่ 1 จงอธิบายว่าโครงร่างโปรแกรมข้างล่างนี้มีปัญหาอะไร และคุณจะแก้ไขอย่างไร

```
     P0              P1
S.acquire()     Q.acquire()     /* semaphore          /* semaphore
Q.acquire()     S.acquire()     acquire method */    release method */
   ......          ......       acquire() {         release(){
S.release()     S.release()       while value <= 0      value++;
Q.release()     Q.release()     ;                     }
                                   value--;
                                }
```

ชื่อ..... รหัสนักศึกษา.....

Question 8: (30 points) [20 minutes] Following is the structure of process Pi in a two process synchronization problem, please explain whether the solution preserves the mutual exclusion and satisfies the progress and the bounded waiting requirements. Variable turn and flag are shared between the two processes

ข้างล่างคือโครงร่างโปรแกรมของโพรเซส Pi ในปัญหาการจัดการทำงานให้สอดคล้องกันของสองโพรเซส โปรดอธิบายว่าวิธีการดังกล่าวทำให้ mutual execution คงอยู่และสามารถรองรับ progress และ bounded waiting หรือไม่ ทั้งสองโพรเซสจะมีการใช้ตัวแปร turn และตัวแปร flag ร่วมกันอยู่

```
int turn;          /* initially 0 */
boolean flag[2];   /* initially false */

do {
  flag[i] = true;
  while(flag[j]) {
     if(turn == j) {
       flag[i] = false;
       while(turn == j)
          ; // do nothing
       flag[i] = ture;
     }
  }
  // critical section
  turn = j;
  flag[i] = false;
  // remainder section
} while(true);
```

Question 9: (30 points) [20 minutes] Show that, if the acquire() and release() semaphore operations are not executed atomically, then mutual exclusion may be violated.

จงแสดงให้เห็นว่าหากคำสั่ง acquire() และคำสั่ง release() ของเซมมะฟอร์ไม่ทำงานแบบอะตอมมิกแล้วอาจทำให้เกิดปัญหากับ mutual exclusion ได้

ชื่อ..... รหัสนักศึกษา.....

Question 10: (20 points) [15 minutes] Explain your solution to the homework 2 part 3 (Reader-Writer problems using 2 threads as readers and 1 thread as the writer). Explain the algorithm and any concept you use to solve this question. อธิบายวิธีการที่คุณใช้ในการตอบคำถามในการบ้านชิ้นที่ 2 ส่วนที่ 3 (ผู้อ่านและผู้เขียน ซึ่งกำหนดให้ 2 เธรดเป็นผู้อ่าน และ 1 เธรดเป็นผู้เขียน) อธิบายอัลกอริทึม แนวคิดและความรู้ต่างๆที่คุณใช้ในการแก้ปัญหาโจทย์ข้อนี้

ชื่อ..... รหัสนักศึกษา.....

Question 11: (20 points) [15 minutes] List questions that you would like to see in this exam and give the answers to your questions. If you have more than one question, please assign the score to each of your questions. However, the total point can not exceed 20 points.

เขียนโจทย์ที่ท่านคาดหวังว่าจะเจอในข้อสอบชุดนี้แต่ไม่เจอ กรุณาเขียนโจทย์และตอบคำถามของท่านเองด้วย หากท่านมีโจทย์มากกว่าหนึ่งข้อ กรุณากำหนดคะแนนให้โจทย์แต่ละข้อด้วย (แต่ต้องรวมกันแล้วไม่เกิน 20 คะแนน)

ชื่อ..... รหัสนักศึกษา.....

Definition (คำนิยาม)

- Multiprogramming ระบบมัลติโปรแกรมมิ่งจะจัดสิ่งแวดล้อมให้ทรัพยากรต่างๆของระบบ เช่น ซีพียู หน่วยความจำ และอุปกรณ์เชื่อมต่ออื่นๆ ให้ถูกใช้ประโยชน์อย่างเต็มที่
- Time-sharing ระบบไทม์แชร์ริ่ง เป็นระบบที่ใช้ระบบมัลติโปรแกรมมิ่งเพื่อสลับเปลี่ยนการทำงานระหว่างงานหลายๆงานโดยสลับการทำงานอย่างรวดเร็วมากจนกระทั่งผู้ใช้ไม่สามารถสังเกตุเห็นได้ว่ามีงานอื่นๆ หรือมีผู้ใช้อื่นๆ ร่วมใช้งานระบบอยู่ในขณะเดียวกันด้วย
- Dedicated machine คือ การที่ใช้งานเครื่องโดยไม่แบ่งกับใครเลย
- Virtual machines คือ การปฏิบัติกับระบบปฏิบัติการเสมือนเป็นฮาร์ดแวร์ตัวหนึ่ง virtual machine จะให้ความรู้สึกเหมือนมีการเชื่อมต่อแบบเดียวกับฮาร์ดแวร์หลายๆ ชนิด
- Starvation ปัญหาการอดอยาก ซึ่งคือการที่โพรเซสหรืองานใดๆ ต้องรอเป็นระยะเวลาอันยาวนานหรือในบางกรณีเป็นการรอแบบไม่มีที่สิ้นสุด
- Preemptive เมื่อโพรเซสได้เริ่มรันแล้ว ระบบจะสามารถหยุดการทำงานของโพรเซสและเรียกทรัพยากรคืนเพื่อไปแจกจ่ายให้กับโพรเซสอื่นได้ ซึ่งในระหว่างนั้น ตัวโพรเซสที่โดนพรีเอ็มก็จะไปรออยู่ใน ready queue
- Non-preemptive เมื่อโพรเซสได้เริ่มรันแล้ว โพรเซสจะใช้งานซีพียูไปจนกระทั่งเสร็จสิ้นการทำงาน
- Waiting time เวลาที่โพรเซสต้องรออยู่ใน ready queue
- Mutual Exclusion: หมายถึงในเวลาขณะใดขณะหนึ่ง หากมีโพรเซสหนึ่งกำลังรันอยู่ใน critical section โพรเซสอื่นๆ จะไม่สามารถรันใน critical section
- Progress: หมายถึง หากไม่มีโพรเซสใดกำลังรันอยู่ใน critical section และมีโพรเซสอื่นๆ ที่กำลังรอจะเข้าไปรันใน critical section แล้วนั้น การเลือกโพรเซสเพื่อเข้าไปรันใน critical section จะไม่สามารถเลื่อนออกไปได้
- Bounded waiting: หมายถึง มีการกำหนดจำนวนครั้งหรือจำนวนโพรเซสอื่นซึ่งเข้าไปรันใน critical section ในระหว่างที่โพรเซสหนึ่งรอขอเข้าไปรันใน critical section จนถึงเวลาที่โพรเซสนั้นได้รับอนุญาตให้รันได้
- Atomic: อะตอมมิก คือ การกระทำงานซึ่งจะต้องทำให้เสร็จสิ้น หรือไม่ทำเลย มีแค่สองกรณีเท่านั้น