

淡江大學 97 學年度碩士班招生考試試題

84-1
84-1

系別：資訊工程學系
資訊工程學系資訊網路與通訊碩士班

科目：作業系統

本試題共 二 頁，十 大題

1.(10%)Make the comparisons between *hard real time* system and *soft real time* system.

2. (10%)Describe the five states when a process execute.

3. (12%) What are two differences between user-level threads and kernel-level threads? Under what circumstances is one type better than the other?

4.(10%) Consider the following preemptive priority-scheduling algorithm based on dynamically changing priorities. Larger priority numbers imply higher priority. When a process is waiting for the CPU (in the ready queue but not running), its priority changes at a rate α ; when it is running, its priority changes at a rate β . All processes are given a priority of 0 when they enter the ready queue. The parameters α and β can be set to give many different scheduling algorithms.

a. What is the algorithm that results from $\beta > \alpha > 0$; $\alpha > \beta > 0$?

b. What is the algorithm that results from $\alpha < \beta < 0$?

5.(12%)Explain the differences in the degree to which the following scheduling algorithms discriminate in favor of short processes:

a. FCFS

b. RR

c. Multilevel feedback queues

6. (14%) Use the following shared data structure and semaphore operations (*wait()* and *signal()*) to design the structure of READER and WRITER processes regarding the synchronization problem. Make sure that your answer keeps READER processes waiting only when a WRITER process has already obtained permission to use the shared object. (Hint: WRITER process may starve)

```
Semaphore mutex, wrt;  
int readcount;
```

本
試
題
雙
面
印
製

系列：資訊工程學系
資訊工程學系資訊網路與通訊碩士班

科目：作業系統

本試題共 二 頁，十 大題

7.(10%) A distributed system using mailboxes has two interprocess communication primitives SEND and RECEIVE. The latter primitive specifies a process to receive from, and blocks if no message from that is available, even though message may be waiting from other processes. The processes need to communicate frequently about other matters. Is deadlock possible? Why.

8. (8%) Consider the following page-replacement algorithms. Rank these algorithms from "bad" to "perfect" according to their page-fault rate. Separate those algorithms that suffer from Belady's anomaly from those that do not.

- a. LRU replacement
- b. FIFO replacement
- c. Optimal replacement
- d. Second-chance replacement

9.(8%) A timesharing machine has a memory of 4K 18-bit words. It held one process at a time in memory. When the scheduler decided to run another process, the process in memory was written to a paging drum, with 4K-18-bit words around the circumference of the drum. The drum could start writing/reading at any word, rather than only at word 0. Why do you suppose this drum was chosen?

10.(6%) Which of the following programming techniques and structures are "good" for a demand-paged environment? Which are "not good"?

- a. Stack
- b. Hashed symbol table
- c. Sequential search
- d. Binary search
- e. Vector operations
- f. Indirection