學號:_____ 姓名:_____ 系級:_____

國立中正大學資訊工程學系大三 B 班
九十二學年度第一學期作業系統期中考試

(1) Consider a system of processes P_1, P_2, …, P_n, each of which has a unique priority number. Write a monitor that allocates five identical CDROMs to these processes, using the priority numbers for deciding order of allocation. (20 %)

Ans:

```
class Monitor{

private:

    bool CDRom[5]; //true 代表該 CDRom 正在使用中，false 代表閒置

    condition x;

public:

    Monitor(){

        CDRom[0] = CDRom[1] = CDRom[2] = CDRom[3] = CDRom[4] = false;

    }


    void acquire(int processID,int CDRomID){

                    if(CDRom[0] && CDRom[1] && CDRom[2] && CDRom[3] && CDRom[4]){

                            x.wait(processID);

                    }

                    if(!CDRom[0]){

                            CDRomID = 0;

                    }else if(!CDRom[1]){

                            CDRomID = 1;
```

```
                }else if(!CDRom[2]){
                        CDRomID = 2;
                }else if(!CDRom[3]){
                        CDRomID = 3;
                }else if(!CDRom[4]){
                        CDRomID = 4;
                }CDRom[CDRomID] = true;
    }

    void release(int CDRomID){
                CDRom[CDRomID] = false;
                x.signal();
    }
};
```

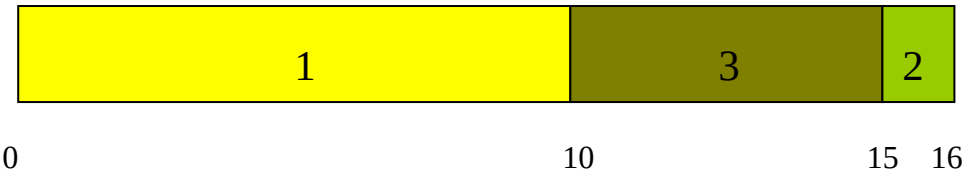
(2) Jobs 1, 2, 3 arrive at times shown below, and require the burst times indicated.

Job	Arrival Time	Burst Time	Priority
1	0	10	33
2	3	1	11
3	2	5	22

a) Draw Gantt charts and determine the average waiting time for FCFS (First Come First Served) scheduling algorithm. (5 %)

(Gantt charts 3%, average waiting time 2%)

Ans:

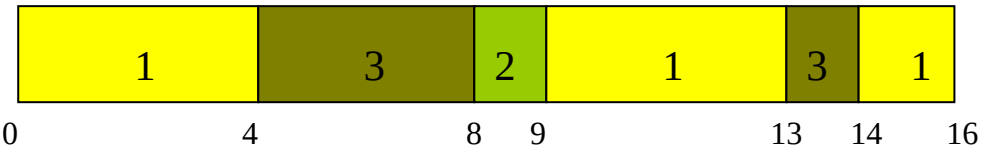


waiting time = finishing time – arrival time – burst time ;

Average waiting time: [(10-10-0)+(16-3-1)+(15-2-5)]/3=20/3

b) Draw Gantt charts and determine the average waiting time for RR (Round Robin) scheduling algorithm with a quantum time of 4 units. (5 %)

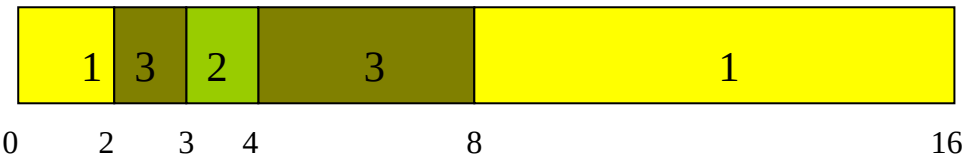
Ans:



Average waiting time: [(16-10-0)+(9-3-1)+(14-2-5)]/3= 6

c) Draw Gantt charts and determine the average waiting time for PP (Preemptive Priority) scheduling, where smaller numbers are of higher priorities. (5 %)

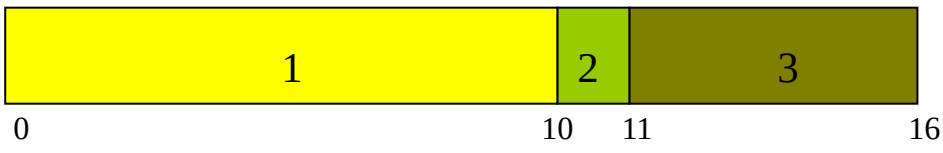
Ans:



Average waiting time: [(16-10-0)+(4-3-1)+(8-2-5)]/3= 7/3

d) Draw Gantt charts and determine the average waiting time for non-preemptive SJF (Shortest Job First) scheduling algorithm. (5 %)

Ans:



Average waiting time: [(10-10-0)+(11-3-1)+(16-2-5)]/3= 16/3

(3) Give one disadvantage found in each of the following scheduling policies: FCFS, SJF, RR, priority-based, and multilevel feedback queue. Explain how the disadvantages may be solved. (10 %)

Solution:

FCFS:

1、會產生護送現象(convoy effect)，使得 cpu 和裝置使用率降低。

2、讓較短的 process 先做。

SJF:

1、在 cpu 排程上，無法預估下一個 cpu 分割的長度。

2、使用前幾次 cpu 分割所測的值之指數平均值，爲其預估值。

RR:

1、取時間量(time quantum) ，若是取太大會有 FCFS 的問題，若是取太小會 context switch 太頻繁。

2、使用 80%的 cpu burst time 小於時間量(time quantum)的標準來訂定時間量的長短。

Priority-based :

1、有無限期的阻塞(indefinite blocking)或是飢餓(starvation)的問題。

2、採用老化(aging)的方法，逐漸提高停留在系統中已經經過一段長時間的 process 之優先權。

Multilevel feedback queue :

1、難以決定 queue 的個數，每一個 queue 分配的時間，queue 跟 queue 之間的時間，queue 之間的排程，該進入那一個 queue 中，決定降底高優先 queue 的 process 到下一 level 的時機的方法，決定何時把 process 提升到 queue 的方法。

2、寫出上述一個改善方法。

(4) When a signal is sent to a process with multiple threads, which thread(s) receive(s) the signal? How does Solaris 2 solve this problem? Does Windows 2000 have this problem? If yes, how does it solve it? If no, why not? (10 %)

Solution :

1、三種不同情形: 全部能接收到 signal 的 threads 都會接受到，再由 thread 本身決定要不要處

理。只會某一個特定的 thread 接受到。user 指定某一個 thread 來接收。

2、solaris2: 指定一個專門來處理 thread signal 的接收。

3、no

4、windows 2000 使用 APCs(asynchronous procedure calls)方式去模擬 signal，當使用者的 thread 接收到一個特殊事件的通知時，APC 允許一個使用者 thread 去說明一個被呼叫的 function。一個 APC 模擬 signal 被傳送到一個特定的 thread。

(5) Explain why short-term process scheduling must be faster than long-term process scheduling. What is medium-term process scheduling? In what different situations is CPU scheduling performed? (10 %)

Solution:

1、 Long-term process scheduler 是從 process pool 中選出 process 將它們載入記憶體內以便執行，而其執行次數很少，因爲只有 process 離開系統的時候才需要被呼叫，進入系統的 new process 可能相隔幾分鐘之久；而 short-term process scheduler 是從記憶體中選出一個已經 ready 的 process，並且將 cpu 使用權分配給它，而且必須經常爲 cpu 選擇新的 process，因爲一個 process 可能只執行幾毫秒然後就等待一項 I/O 要求，所以要馬上更換另一個 process 執行，相隔時間非常短(幾十毫秒)，short-term process scheduler 必須要非常快速。由上述可知 short-term process scheduler 比 long-term process scheduler 執行上快。

2、Medium-term process scheduling : 例如 time-sharing systems。可以將行程從記憶體中有效地移開並且從 cpu 競爭中移開，由此減少在記憶體中的 processes 個數。稍後，透過 medium-term process scheduling 程式再把該行程放回記憶體中並且放在它移開之前的位置上繼續執行。

3、四種可能會發生 CPU 排程時機:

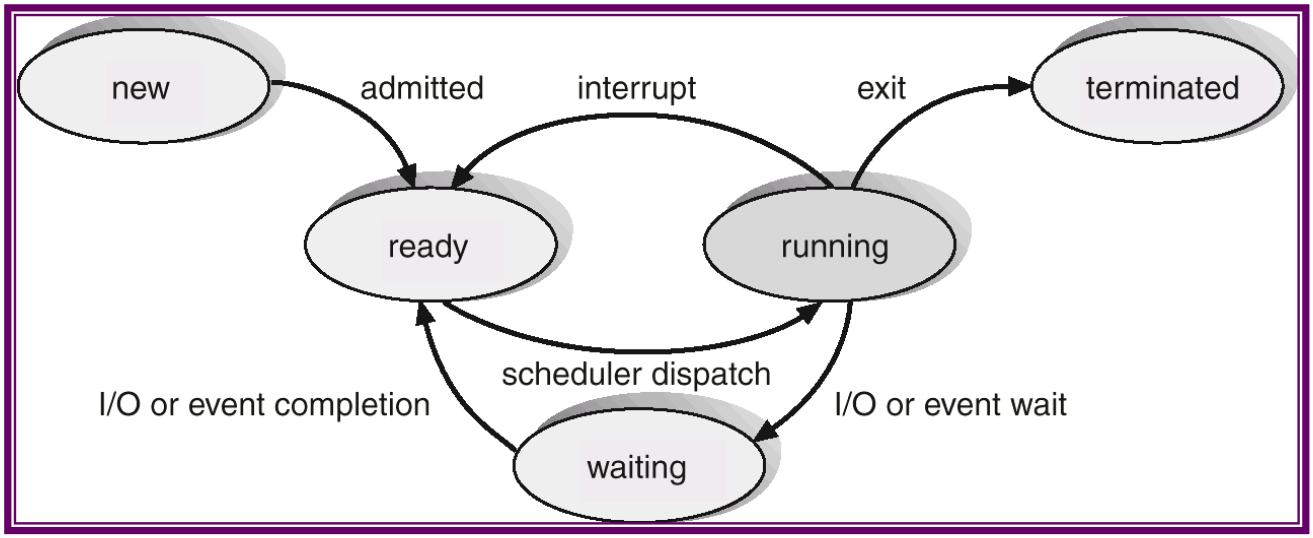
A、Switches from running to waiting state.

B、Switches from running to ready state.

C、 Switches from waiting to ready.

D、.Terminates.

學號:_____ 姓名:_____ 系級:_____



(6) Explain the following terms in one sentence each. **(10 %)**

(a) deferred thread cancellation

(b) memory-mapped I/O

(c) monitor (for synchronization)

(d) virtual machine

(e) soft real-time systems

(f) symmetric multiprocessing

(g) deterministic modeling

(h) turnstiles

(i) condition variable

(j) semaphore

Ans:

(a) The target thread can periodically check if it should terminate, allowing the target thread an opportunity terminate itself in an orderly fashion.

(b) The I/O device registers are mapped to certain reserved address ranges in the main memory.

(c) A high level synchronization structure which ensures only one process at a time can be active within itself .

(d) An interface is identical to the underlying bare machine, each process is provided a virtual copy of the underlying computer.

(e) Soft real-time systems allow some deadline can be violated.

(f) Each processor runs an identical copy of the operating system, these copies communicate with one another as needed.

(g) Taking a particular predetermined workload, it defines the performance of each algorithm for that workload.

(h) A queue structure containing threads blocked on a lock.

(i) A programmer who needs to write her own tailor-made synchronization scheme can define one or more variables of type condition in monitors.

(j) A semaphore is an integer variable that accessed only through two standard atomic operations: WAIT and SIGNAL.

(7) A production system, consisting of 3 processes, is to be automated by a program which contains the following 4 tasks: (20 %)

- **SUPERVISION1**, **SUPERVISION2**, **SUPERVISION3**:
  to read in data from respective processes,
- **PROTOCOL**:
  to output the measured values in tables and diagrams on a printer.

The following sequence of the tasks must be created by a suitable synchronization with semaphores:

- The protocol-task should be executed after each supervision-task.
- The sequence of the supervision-tasks is given by the number in their name. SUPERVISION1 is therefore the first task.
- The sequence given above should be repeated cyclically.
- The program of each task should be executed completely, before another task can be started.

Questions:

(a) Write down the names of the tasks in the desired sequence. (5%)

Ans:**SUPERVISION1->PROTOCOL->SUPERVISION2->PROTOCOL-> SUPERVISION3 -> PROTOCOL**

(b) Insert into each task the semaphore-operations required to guarantee the desired task sequence. (10 %)

Ans:

```
SUPERVISION1( ){
P(S1);
P(mutex);
….
….
….
V(S2);
V(SP);
}
```

```
SUPERVISION2( ){
P(S2);
P(mutex);
….
….
….
V(S3);
V(SP);
}
```

```
SUPERVISION3( ){
P(S3);
P(mutex);
….
….
….
V(S1);
V(SP);
}
```

```
PROTOCOL (){
P(SP);
….
….
….
V(mutex);
}
```

(c) With which values do the semaphore variables have to be initialized? (5 %)

Ans: mutex =1, S1 = 1, S2=0, S3=0, SP=0