| 课程名称 | 操作系统原理及应用 | 考试学期 | 13-14-3 | 得分 |
| --- | --- | --- | --- | --- |

| 适用专业 | 软件工程 | 考试形式 | 闭卷 | 考试时间长度 | 90分钟 |
| --- | --- | --- | --- | --- | --- |

## 1.Short Answer Questions(Please write the answers on the answer sheet.)

（1）Describe the advantages of multiprogramming（5pts）.

（2）What is the dual-mode of operation? What is the function of the mode bit (7pts)?

（3）Describe the advantages of Two-level Multithreading Mode（5pts）。

（4）Describe the differences between process and program (7pts).

（5）Describe the differences between Non-preemptive and preemptive（5pts）.

（6）Why is it important for the scheduler to distinguish I/O-bound programs from CPU-bound programs (8pts)?

（7）What is the main advantage of the microkernel approach to operating system design（5pts）?

（8）Describe the five state process model, describe what transitions are valid between the five states, and describe an event that might cause such a transition. (8pts).

（9）Describe the differences among short-term, medium-term, and long-term scheduling（6pts）..

（10）Describe the differences between shared memory and message passing in interprocess communication. (6pts).

2. (6pts) Please give the probable screen output of the following program.

```
int main(){
    pid_t   pid;
    int a=8,b=5;
    pid = fork();        /* fork another process */
    if (pid < 0) {       /* error occurred */
        fprintf(stderr, "Fork Failed");exit(-1);
    }
    else if (pid > 0) {
        printf(" case11  pid > 0, a=%d, b=%d\n",a,b);
        a=a*2;
        wait (NULL);
        printf(" case12  pid > 0, a=%d, b=%d\n",a,b);
    }
    else {
        b=a-1;
        printf("case 2, a=%d, b=%d\n",a,b);
        exit(0);
    }
}
```

*(handwritten notes:)* case11 pid>0 a=8,b=5　case2 a=8,b=7　case12 pid>0 a=16,b=5

3.(12pts) Consider the following set of processes, assumed to have arrived at time 0, in the order P1, P2, P3, P4, P5, with the length of the CPU-burst time given in milliseconds:

| Process | P1 | P2 | P3 | P4 | P5 |
| --- | --- | --- | --- | --- | --- |
| Burst Time | 9 | 3 | 2 | 1 | 5 |
| Priority | 3 | 1 | 5 | 4 | 2 |

*(handwritten notes:)* P1: 4+3+2+2=11　P2: 1+4+3=　P3: 2+4 =6　P5: 4+3+2+2=11

(a) Draw four Gantt charts that illustrate the execution of these processes using FCFS, non-preemptive SJF, non-preemptive priority (a smaller priority number implies a higher priority), and RR (quantum = 1) scheduling.

(b) What are the average turnaround time and the average waiting time of all process for each of the scheduling algorithms in part (a) respectively?

4.(8 pts) Do you think the following algorithm can solute the critical section problem of P0 and P1? Give your reason.

Share variables int turn=0;

Share variables boolean flag[0]= flag[1]=false;

```
P0:  do {                          P1:  do {
        flag[0] = true;                   flag[1] = true;
        while (turn != 0){                while (turn != 1){
            while (flag [1]);                 while (flag [0]);
            turn =0;                          turn =1;
        }                                 }
        critical section                  critical section
        reminder;                         reminder;
    } while (true);                    } while (true);
```

5.(12 pts) Give a pseudo-code solution to the readers/writers problem in which writers are favored (specifically, if a writer is waiting to start reading, then no new readers are allowed to start reading). Use semaphores as your synchronization primitive.

*(handwritten notes:)* if( ==1) wait(wre) signal(mutex) //read. wait(mutex) read-- if(==0) signal(wre). signal(mutex)

```
    wait(NULL);
    printf("PARENT: value = %d", value);    /*Print 2*/
    }
}
```

(a) What is the screen output of the above program? Please give your reasons.

(b) What are the functions of pthread_join() and wait() respectively?

## 4. CPU Scheduling [15 pts] (Answer this question on the *answer sheet*.)

Consider a multi-level feedback queue in a single-CPU system. The first level (queue 0) is given a quantum of 8 ms, the second one a quantum of 16 ms, the third is scheduled FCFS. Assume jobs arrive all at time zero with the following job times (in ms): 4, 7, 12, 20, 25 and 30, respectively.

(a) Draw a Complete Gantt Chart to illustrate the execution of the processes using the multi-level feedback queue scheduling described above.

(b) Show calculation of Average Waiting Time.

## 5. Process Synchronization [15 pts] (Answer this question on the *answer sheet*.)

(a) What is the critical-section problem? What are the three *correctness properties* to a solution of the critical-section problem. Give a short explanation for each property.

(b) Peterson's algorithm to solve the critical section problem for two processes was adopted by a student as below, messing up some of the variables. With respect to the three *correctness properties*, indicate which one is satisfied and which one is not satisfied, and give your reasons..

| Initially flag[1] = true, flag[2] = false, turn = 2; | |
| Process 1 | Process 2 |
| do { | do { |
| 1:  flag[1] = true; | 1:  flag[2] = true; |
| 2:  turn = 1; | 2:  turn = 2; |
| 3:  while (flag[2] and turn == 2); | 3:  while (flag[1] and turn == 1); |
| 4:   < critical section > | 4:   < critical section > |
| 5:  flag[1] = false; | 5:  flag[2] = false; |
| 6:   < remainder section > | 6:   < remainder section > |
| } while (1); | } while (1); |

## 6. Process Synchronization [10 pts] (Answer this question on the *answer sheet*.)

One teacher and fifty students participate in the midterm exam of Operating System Course together. Only one person can enter or leave the examination room at one

---

time. The teacher can not distribute the exam papers until fifty students all enter the examination room. The students can leave the examination room after they hand in their examination papers. The teacher can not leave the examination room until he collects all the papers. Please write C-like pseudo code that performs the appropriate initializations and implements the synchronization and the mutual exclusion between teacher and students by using semaphores.

(a) 进程 和程序的区别：

① 进程是执行中的程序, 进程有动态性.
   程序是一组代码的行为集合, 是静态的.

② 进程是暂时的, 是程序的一次执行
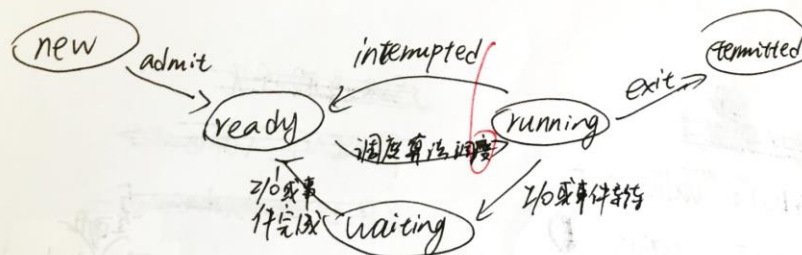   程序是永久的.

③ 进程不能脱离程序而存在. 程序指明了进程执行的顺序

(b) I/O bound 是指运行 I/O操作时间比 CPU计算时间长的进程
   CPU bound 是指 CPU 计算时间比 I/O 操作时间长的进程.
   application :  I/O bound process  打印机打印进程.
              CPU bound  process  科学计算, 大数据处理进程.

(c)



(d) 多线程程序的优点：

① 响应性好.

② 经济

③ 资源共享.

④ 多处理器体系结构的应用

例如一个 web 器, 如果它是一个单线程程序, 那么在加载图片时, 就不能够处理用户操作, 这使得响应性下降; 但如果是一个多线程程序, 那么在做加载图片应用时, 有另一个线程负责响应用户的操作, 这就使得应用的资源这个线程两个同时, 响应性更好, 用户体验更好, 有更好的性能.