Indian Institute of Technology Ropar

Quiz-1 (September 2023)

Course Name	Course Code	Date of Exam	MM	Time
Operating Systems	CS-303	Sept. 06, 2023	10	40 mins

Important Instructions*: All questions are compulsory.*

1. **(1 marks)** How many times does this code print ”hello”? Explain your answer.

```
void main(int argc, char **argv) {
    int i;
    for (i=0; i < 3; i++) {
        execl("/bin/echo", "echo", "hello", 0);
    }
}
```

Note: Answers without explanation will be given zero marks.

Solution: Ans : execl overwrites the current process by loading the program /bin/echo. The for loop is gone! Therefore, the answer is 1 *Note for marking: no partial marking*

2. **(2 marks)** Assume there are 4 processes in the system. The processes are scheduled using Shortest Job First algorithm. Prove that this algorithm is optimal for mean turnaround time objective.
Hint: Calculate mean turnaround time for 4 jobs and then see when it is minimized.

Solution: Assume the burst times of the processes are t_1, t_2, t_3, t_4 and they come in the same order then the average turnaround time is as follows:

$$\frac{(4*t_1+3*t_2+2*t_3+t_4)}{4}$$

This formula is minimized when $t_1 \leq t_2 \leq t_3 \leq t_4$, hence shortest job first is optimal.

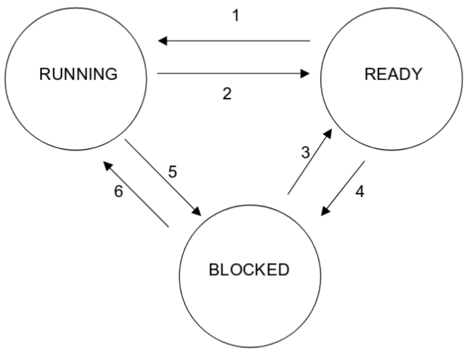
3. **(2.5 marks)** Consider a system running ten 1/O-bounds tasks and one CPU-bound task. Assume that the I/O bound tasks issue an I/O operation once for every millisecond of CPU computing and that each I/O operation takes 10 milliseconds to complete. Also, assume that the context-switching overhead is 0.1 millisecond and all processes are long-running tasks. What is the CPU utilization for a round-robin scheduler when the time quantum is 10 millisecond.
Hint: Try to find the CPU used for one quanta of every process.

Solution:

The time quantum is 10 milliseconds: The I/O-bound tasks incur a context switch after using up only 1 millisecond of the time quantum. The time required to cycle through all the processes is therefore 10*1.1 + 10.1 (as each I/O-bound task executes for 1 millisecond and then incur the context switch task, whereas the CPU-bound task executes for 10 milliseconds before incurring a context switch). The CPU utilization is therefore 20/21.1 * 100 = 94.78%.

Note for marking: no partial marking

4. **(1.5 marks)** Briefly explaining what conditions cause a process to move between each of the 3 states, and what causes each arrow. Label it N/A if it doesn’t happen.



Solution:

```
Arrow 1: Process is scheduled and run by the scheduler.
Arrow 2: Time slice runs out, but process is still wanting to run. Yield().
Arrow 3: I/O completes, or lock is acquired. Woken up by a semaphore.
Arrow 4: N/A
Arrow 5: Any blocking action. I/O request, lock blocks.
Arrow 6: N/A
```

(***Note for marking: Give 0.25 marks for each of 6 arrows.***)

5. **(0.5+0.5+1+1 marks)** Assume the following command is run on the terminal window which is running `/bin/bash` shell.

```
$ ./a.out > out.txt
```

Answer the following questions:

(a) How many processes are there in this discussion, give names of processes?

(b) Fill in the blanks in the following code.

```
pid_t p = fork();
if ( p < 0 ) {
     exit(-1);
} else if ( p == 0 ) {
   ___________________;
   ___________________;
   ___________________;
} else {
    wait(NULL);
    exit(0);
}
```

Solution:

Ans 1: There are 2 processes in the system, one is `bash` and the other one is `a.out` which will be created by bash. **(0.5 marks)**

Ans2:

```
blank-1: close(1);            (0.5 marks)
blank-2: open("out.txt",O_WRONLY | O_CREAT,0777);       (1 marks)
blank-3: exec("./a.out","a.out",NULL);      (1 marks)
```

Note for marking: Students may provide answers where they pass out.txt as an argument to a.out, this answer is not acceptable since, a.out is a binary and should not be altered. The partial code provided is a part of the /bin/bash not a.out. For blank-2, we can give partial marking of 0.5 if they only used open but not O_CREAT, however, opening file in write mode is compulsory.