



Department of Computer Science and Engineering
Midterm Examination Spring 2025
CSE 321: Operating Systems

Duration: 1 Hour 30 Minutes

Total Marks: 30

Answer all the following questions.
 Figures in the right margin indicate marks.

Name:	ID:	Sec:
-------	-----	------

1. a) Write the answers in the question paper.

CO1

i) Let's consider a scenario where a total of **4** processes are available for CPU allocation. Now, they have burst times like **4, 5, 6, 8**. The OS is configured to use Round-Robin scheduling with a time quantum of **10**. At this point, **figure out** the problem with this amount of time quantum.

[1*2
=2]

Answer:

ii) Observe the following code. **Calculate** how many processes and how many threads will be created.

```
pid_t pid;
pthread_create( . . . );
pid = fork();
if (pid == 0) { /* child process */
    fork();
}
fork();
```

Answer:

b) Find outputs of the given code:

[10*

0.5=

5]

```
int main(){
    pid_t a,b;
    int c[]={5,7,3};
    a=fork();
    if(a<0){
        printf("error\n");
    }
    else if(a==0){
        c[0]=c[1]-c[2];
        c[1]=c[0]-c[2];
        c[2]=c[0]-c[1];
        b=fork();
        if(b<0){

        }
        else if(b>0){
            wait();
            c[0]=c[1]*c[2];
            c[1]=c[0]*c[2];
            c[2]=c[0]*c[1];
        }
        else{
            c[0]=c[1]+c[2];
            c[1]=c[0]+c[2];
            c[2]=c[0]+c[1];
        }
    }
    else{
        printf("hello\n");
        wait();
        c[0]=c[1]+c[2];
        c[1]=c[0]+c[2];
        c[2]=c[0]+c[1];
    }
    for(int i=0;i<3;i++){
        printf("c[%d]= %d\n",i,c[i]);
    }
    return 0;
}
```

Write the outputs in the question paper. Output sequence should be exactly matched.

Outputs

1	
2	
3	
4	
5	
6	
7	
8	
9	
10	

c) Write the answers in the answer script.

Dipu is implementing his own OS. For his OS he has decided to develop **Multilevel Feedback Queue** as a scheduler. To design the scheduler he decided to divide the ready queue into **three** levels. According to his preferences, in every level round-robin algorithm will be used and some constraints should be maintained in each level which are given below.

- **Q1:** time quantum = **3**, priority = **1**
- **Q2:** time quantum = **5**, priority = **5**
- **Q3:** time quantum = **10**, priority = **7**

Here, the lower priority value indicates the higher priority and in case of an I/O request, the scheduler will promote a process by one level upon I/O operation completion. At a certain moment, Dipu has collected information on burst times, arrival times, and I/O times of four processes

Process	Burst Time	Arrival Time	I/O Time
P1	19	0	6 [After 15s of total CPU allocation]
P2	8	2	N/A
P3	9	12	4 [After 7s of total CPU allocation]
P4	17	7	N/A

- I. **Draw** a Gantt chart using multilevel feedback queue scheduling algorithm showing the states of the ready queues of different levels. **[12]**
- II. **Calculate** the average waiting time and average turnaround time from the Gantt chart. **[1.5+
1.5=
3]**

2. **a) Write the answers in the question paper.** **[1*3
=3]**

CO2

i) You are given an array of **100** integers. You are required to calculate the sum of these integers using a **10-core** CPU. **Identify** what kind of parallelism you need to implement to ensure that the operation is performed most efficiently.

Answer:

ii) The BRACU CSE Department has unveiled its newly upgraded research lab featuring **55** high-performance PCs. These PCs are allocated to thesis groups based on their area of research. The allocation is as follows:

- **20** PCs are dedicated to AI/ML-related thesis groups.
- **18** PCs are dedicated to NLP-related thesis groups.
- The remaining PCs are assigned to Image Processing-related thesis groups.

Each thesis group is allocated some PC within their respective topic area for their allocated time. However, since the total number of thesis groups exceeds the number of available PCs, groups must wait their turn if all PCs are currently in use. Here, a strict constraint is maintained that each group can only use the PCs allocated to their area of research. During the thesis defense week, **23** NLP groups may try to access PCs, but there may not be enough PCs in the NLP section, and some of them may need to wait for their turn. **State** which synchronization method can be used here to ensure the synchronization.

Answer:

iii) You are booking a train ticket for your journey back home during Eid. When you visit the booking website, you find that there is only one seat left. You proceed to book it, and the booking is confirmed. The code used to book a seat is as follows:

```
lock = false
book_seat(){
    while(test_and_set(&lock));
        if (available_seats > 0) {    // Check if seats are
available
            available_seats = available_seats - 1 // Deduct
seat
            print("Booking confirmed!") // Confirm booking
        }
        else {
            print("No seats available.")
        }
        lock = true;
    }
```

Figure out the error from the above code.

Answer:

b) Find outputs of the given code:

[10*

0.5=

5]

```
int t_id[]={1,2,3,4};
void *func(int *id);
int a=123;
int b=50;
sem_t s1,s2,s3,s4;

int main(){
    pthread_t t[4];
    sem_init(&s1,0,0);
    sem_init(&s2,0,1);
    sem_init(&s3,0,0);
    sem_init(&s4,0,0);
    for(int i=0;i<4;i++){
        pthread_create(&t[i],NULL,(void *)func,&t_id[i]);
    }
    for(int i=0;i<4;i++){
        pthread_join(t[i],NULL);
    }
    sem_destroy(&s1);
    sem_destroy(&s2);
    sem_destroy(&s3);
    sem_destroy(&s4);
    printf("Final a: %d\nFinal b: %d\n",a,b);
    return 0;
}

void *func(int *id){
    if(*id==1){
        sem_wait(&s3);
        printf("a: %d\nb: %d\n",a,b);
        a=a+b;
        b=b-a;
        sem_post(&s3);
    }
    else if(*id==2){
        sem_wait(&s1);
        printf("a: %d\nb: %d\n",a,b);
        a=a*b;
        b=b*5;
        sem_post(&s1);
        sem_post(&s3);
    }
    else if(*id==3){
        sem_wait(&s2);
        printf("a: %d\nb: %d\n",a,b);
        a=a-b;
        b=b-a;
        sem_post(&s2);
        sem_post(&s4);
    }
    else{
        sem_wait(&s4);
        printf("a: %d\nb: %d\n",a,b);
        a=a-12;
        b=b-16;
        sem_post(&s4);
        sem_post(&s1);
    }
}
```

Write the outputs in the question paper. Output sequence should be exactly matched.

Outputs

1	
2	
3	
4	
5	
6	
7	
8	
9	
10	