National University of Computer & Emerging Sciences, Karachi
Fall-2018 CS-Department

Lab Final

Course Code: CL205	**Course Name: Operating Systems Lab**
Instructor Name: Sumaiyah Zahid	
Student Roll No:	**Section:**

"If there is something, you don't know today. You will surely learn afterwards. Life is not an exam hall."
BEST OF LUCK!

Instructions

- Rules are made to break them. So, invent yours and I'll break.

Time: 90 minutes **Max Marks:** 40 points

This program will create ____ child processes and ____ threads? (5 marks)

```
int main()
{
    printf("A\n");
    pthread_create(&tid, NULL, thread, NULL);
    fork();
    fork();
    pthread_create(&tid, NULL, thread, NULL);
    printf("B\n");
    fork();
    pthread_create(&tid, NULL, thread, NULL);
    return 0;
}
```

Output

```
int main() {
        printf("%d\n", getpid());
        a=fork();
        printf("%d\n", getpid());
        if (a>0){
        printf("%d\n", getpid());
        }
        fork();
        printf("%d\n", getpid());
        printf("      Done!\n");
return 0;
}
```

Output

Write appropriate system calls in the blanks **(5 marks)**

```
int main(void) {
        int shmid;
        key_t key;
        char *shm, *s;
        key = 2211;
        fflush(stdin);
        if((shmid = _____(key, MAXSIZE, IPC_CREAT | 0666)) < 0)
                die("error");
        if((shm = _____ ( _____, NULL, 0)) == (char*) -1)
                die("error");
        for(s = shm; *s != '\0'; s++)
                putchar(*s);
        *shm = '*';
        printf("\n");
        exit(0);
}
```

Advantage of FIFO over pipe is

a) related processes can communicate
b) unrelated processes can communicate
c) all of the mentioned
d) none of the mentioned

Which is Fastest IPC?

a) Message Queue
b) shared memory
c) Socket
d) All of the mentioned

What are the two basic function for any module? **(5 marks)**

1. _____
2. _____

Command for compiling module _____
Command for adding module to kernel _____

What is the output on the terminal after compiling?

```
printk(KERN_INFO "Hey! \n");
printk(KERN_INFO "Final Paper of OS”);
printk("GoodBye");
return 0;
```

What is the difference between the two program? **(2 marks)**

pthread_t t[N];
for (i = 0; i < N; i++)
pthread_create(&t[i], NULL, thread_func, NULL);
for (i = 0; i < N; i++)
pthread_join(t[i], NULL);
	pthread_t t[N];
	for (i = 0; i < N; i++) {
	pthread_create(&t[i], NULL, thread_func, NULL);
	pthread_join(t[i], NULL);
	}

_____
_____
_____
_____

True or false: Code in an OpenMP program that is covered by a pragma is executed by all threads.

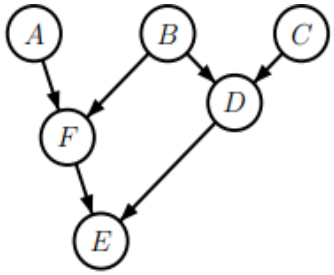
(1 marks)

A bank has four counters to give their services to customers.

- Counter 1 is used for Cash Withdrawals
- Counter 2 is used for Deposits
- Counter 3 is used for submitting Challans
- Counter 4 is used for Bank Inquiry

Implement this system using OpenMP for total 100 customers. Asuming 25 customers for each counter. **(5 marks)**

Write a sketch of a C program that uses Pthreads to execute the five functions in a way that is maximally parallel, but adheres to the above dependency graph.
The edge from node B to node D means that functionB must be called, and must return, before functionD can be called. **(2 marks)**



Write all possible output on executing the code below? **(3 marks)**

```
sem_t mutex;
int i=0;
void* thread(void* arg)
{
   Int a= * ((int*)arg);
   i++;
   printf("\nEntering..\n");
   sem_wait(&mutex);
   i++;
   printf("\n %d Entered..\n",a);
   printf(" Value of i is %d",i);
   sem_post(&mutex);
}
int main()
{
   sem_init(&mutex, 0, 1);
   pthread_t t1,t2;
   pthread_create(&t1,NULL,thread,&0);
   pthread_create(&t2,NULL,thread,&1);
   pthread_join(t1,NULL);
   pthread_join(t2,NULL);
   sem_destroy(&mutex);
   return 0; }
```

Output

A car is manufactured at each stop on a conveyor belt in a car factory. A car is constructed from the following parts - chassis, tires, seats, engine, the top cover, and painting. Thus there are 6 tasks in manufacturing a car. However, tires, seats or the engine cannot be added until the chassis is placed on the belt. The car top cannot be added until tires, seats and the engine are put in. Finally, the car cannot be painted until the top is put on.

A stop on the conveyor belt in your car company has four technicians assigned to it - Abe, Bob, Charlie, and Dave. Abe is skilled at adding tires and painting, Bob can only put the chassis on the belt, Charlie only knows how to attach the seats, and Dave knows how to add the engine as well as how to add the top.

Write code for Abe, Bob, Charlie and Dave to be able to work on the car, without violating the task order outlined above. **(5 marks)**

Write a code snippet which sets default behavior of ctrl+\, ignores ctrl+Z, assign funcA to ctrl+C.and func B to floating point error. (5 marks)

What is the output on executing the code below and pressing ctrl+Z 3 times? (2 marks)

```
int main(void)
{
        int i;
        signal(SIGSTP,  quit);
        signal(SIGKILL, quit);

        for (i = 1; i <= 20000000; i++)
        {
        }
}

void quit(int sig) {
        signal(sig, quit);
        cout<<"Ha Ha";
}
```

Output