

Name : **Ibadullah Shaikh**
Roll No : **19k-0259**
Section : **BCS-5G**

Lab 07

Q1.

Code:

One.c

```
#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<pthread.h>

void * fun(pthread_t id)
{
    return NULL;
}

int main()
{
    pthread_t threadID[3];
    for(int i = 0; i < 3 ;i++)
    {
        if(pthread_create(&threadID[i] , NULL , fun , NULL) != 0)
        {
            printf("Error...!!");
        }
        printf("Thread: %d\n" , i);
    }
}
```

```

        for(int i = 0; i < 3 ;i++)
        {
            int x = pthread_join(threadID[i] , NULL);
            printf("Thread %d returned: %d\n" , i , x);
        }
        return 0;
    }
}

```

two.c

```

#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<pthread.h>

int t = 1000;

void *table(int n)
{
    for(int i = 0 ; i < t ; i++)
    {
        printf("%d * %d: %d\n" , n , i+1 , n * (i+1));
    }
    return NULL;
}

int main()
{
    pthread_t threadID[4];
    int tbl[4] = {5,6,7,8};
}

```

```

pthread_create(&threadID[0] , NULL , table , tbl[0] );
pthread_create(&threadID[1] , NULL , table , tbl[1] );
pthread_create(&threadID[2] , NULL , table , tbl[2] );
pthread_create(&threadID[3] , NULL , table , tbl[3] );

pthread_join(threadID[0] , NULL );
pthread_join(threadID[1] , NULL );
pthread_join(threadID[2] , NULL );
pthread_join(threadID[3] , NULL );

return 0;
}

```

Q2

Code:

```

#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<pthread.h>
#include<sys/wait.h>
#include<time.h>

int no_of_sponsors=0;
int registered=0;
void *registration(void *args)
{
    if(registered <=100)
    {
        printf("Participant %d -> Total Registered: %d\n" ,(int)args +1, ++registered);
    }
}

```

```
        else
        {
            printf("Full...!!\n");
        }
        return NULL;
    }
}
```

```
void *announcements(void *args)
{
    printf("Participant %d -> Currently there are no announcements...!!\n",(int)args +1);
    return NULL;
}
```

```
void *handleSponsors(void *args)
{
    printf(" Participant %d -> Number of Sponosors: %d\n" ,(int)args +1, ++no_of_sponsors);
    return NULL;
}
```

```
void *queries(void *args)
{
    printf("Participant %d -> Query resolved...!!\n",(int)args +1);
    return NULL;
}
```

```
int main()
{
    void *volunteers[4] = {registration , announcements , handleSponsors , queries };
    char *array[4] = {"Register" , "Announcments" , "Handle Sponsors" , "Queries"};
```

```
    srand(time(NULL));
    pthread_t participants[100];
    for(int i = 0 ; i < 100 ; i++)
    {
        int s = rand()%4;
        printf("Participant %d went to volunteer %d to (%s) Counter\n" , i+1 , s+1 , array[s] );
        pthread_create(&participants[i] , NULL , volunteers[rand()%4] , (void*)i );
    }
    for (int i = 0 ; i < 100 ; i++)
    {
        pthread_join(participants[i],NULL);
    }
    return 0;
}
```