VENKATA REVAN NAGIREDDY

CB.EN.U4CYS21083

MULTITHREADING

1. Name:

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

// printWelcomeMessage will be called when the Thread is
created in the main function
// which takes string as an argument
void *printWelcomeMessage(void *names) {

   sleep(2);
   char *name = (char *)names;
   printf("\n[THREAD] Hello, Welcome %s.", name);
   pthread_exit(NULL);

}

int main () {

   // thread defintion
   pthread_t threads[7];

   // parameter to be passed to the called function -
printWelcomeMessage
   char names[10][15] =
{"Amritha","Praveen","Saurabh","Sangeetha","Lakshmy","Srinivas
an","Ramaguru"};

   int result;

   for(int i = 0; i < 7; i++ ) {
```

```
      printf("\n[MAIN] Creating thread, %d", i);

      // Creating the threading and thus calling the function
with parameter passed to it
      result = pthread_create(&threads[i], NULL,
printWelcomeMessage, (void *)names[i]);

      if (result) {

         printf("Error in creating thread, %d ", result);
         exit(-1);
      }

   }

   // Exit the thread
   pthread_exit(NULL);
}
```

```
quiver@quiver-machine:~/Documents/MultiTreading$ gcc name.c
quiver@quiver-machine:~/Documents/MultiTreading$ ./a.out

[MAIN] Creating thread, 0
[MAIN] Creating thread, 1
[MAIN] Creating thread, 2
[MAIN] Creating thread, 3
[MAIN] Creating thread, 4
[MAIN] Creating thread, 5
[MAIN] Creating thread, 6
[THREAD] Hello, Welcome Amritha.
[THREAD] Hello, Welcome Praveen.
[THREAD] Hello, Welcome Lakshmy.
[THREAD] Hello, Welcome Srinivasan.
[THREAD] Hello, Welcome Ramaguru.
[THREAD] Hello, Welcome Saurabh.
[THREAD] Hello, Welcome Sangeetha.quiver@quiver-machine:~/Docu
g$ S
```

2. ID:

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

// printWelcomeMessage will be called when the Thread is
created in the main function
// which takes string as an argument
struct myname{
    int val1;
    int val2;
}a;

void *printWelcomeMessage(void *threadid) {
   sleep(2);
   long tid=(long)threadid;
   printf("\n[THREAD] Hello, Welcome %ld.", tid);
   pthread_exit(NULL);

}


int main () {

   // thread defintion
   pthread_t threads[5];

   // parameter to be passed to the called function -
printWelcomeMessage
   char threadid[10][15] =
{"Amritha","Praveen","Saurabh","Sangeetha","Lakshmy","Srinivas
an","Ramaguru"};

   int result;

   for(int i = 0; i < 7; i++ ) {

      printf("\n[MAIN] Creating thread, %d", i);

      // Creating the threading and thus calling the function
with parameter passed to it
```

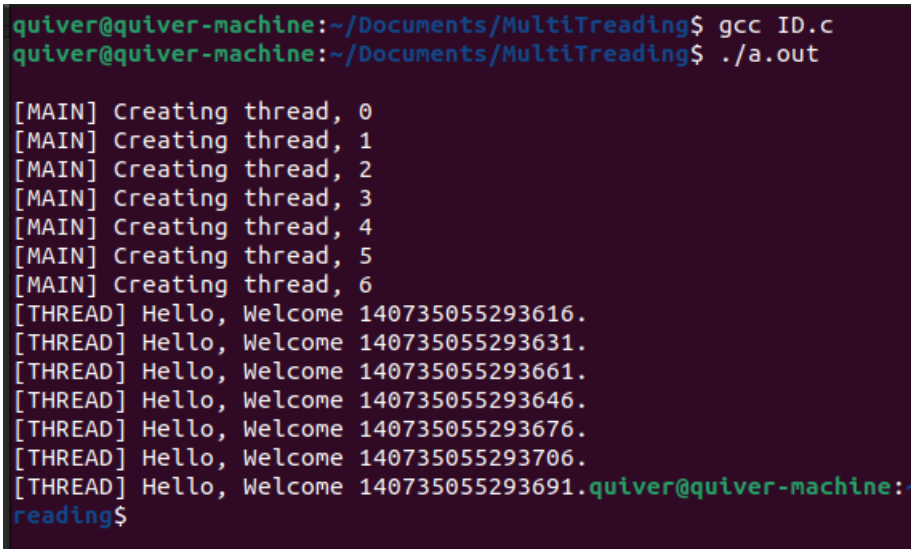
```
        result = pthread_create(&threads[i], NULL,
    printWelcomeMessage, (void *)threadid[i]);

        if (result) {


           printf("Error in creating thread, %d ", result);
           exit(-1);
        }

     }

     // Exit the thread
     pthread_exit(NULL);
    }
```



3. Addition:

```
    #include <pthread.h>

    #include <stdlib.h>

    #include <stdio.h>

    #include <unistd.h>

    // printWelcomeMessage will be called when the Thread is created in
the main function

    // which takes string as an argument

    void *addition() {
```

```
        int value1;
        int value2;
        printf("Enter the 1st element:");
              scanf("%d",&value1);
        printf("Enter the 2nd element:");
        scanf("%d",&value2);
        int total = value1 + value2;
        printf("%d",total);
        }
        int main () {
        // thread defintion
        pthread_t threads;
        int result;
        // Creating the threading and thus calling the function with
parameter passed to it
        result = pthread_create(&threads, NULL, addition, NULL);
        if (result) {
           printf("Error in creating thread, %d ", result);
           exit(-1);
        }
   // Exit the thread
   pthread_exit(NULL);
}
```

```
quiver@quiver-machine:~/Documents/MultiTreading$ gcc addition.c
quiver@quiver-machine:~/Documents/MultiTreading$ ./a.out
4
5
Adding 4 and 5 gives 9
4
5
Adding 4 and 5 gives 9
8
5
Adding 8 and 5 gives 13
9
6
Adding 9 and 6 gives 15
2
5
Adding 2 and 5 gives 7
quiver@quiver-machine:~/Documents/MultiTreading$
```