Operating systems lab exercise

Name: Alooysius Vitalian

Roll no: CB.EN.U4CYS21006

1st program:

#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[5];

// parameter to be passed to the called function - printWelcomeMessage

char names[10][15] = {"Amritha","Praveen","Saurabh","Sangeetha","Lakshmy","Srinivasan","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);

}

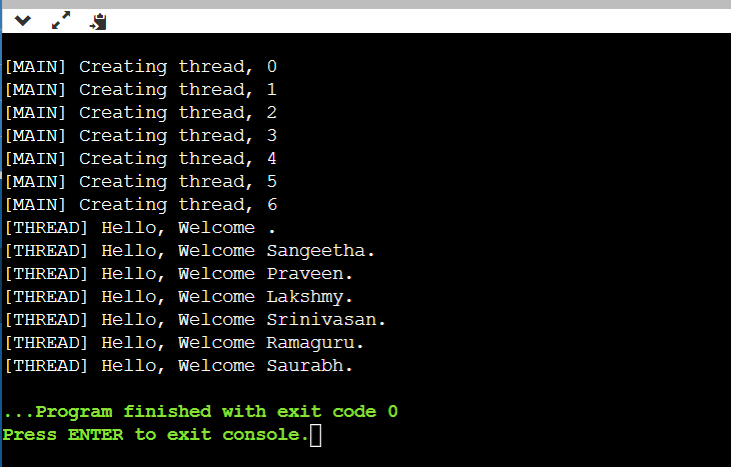
}

// Exit the thread

pthread\_exit(NULL);

}

Output:



2nd Program:

#include <pthread.h>

#include <stdlib.h>

#include <stdio.h>

#include <unistd.h>

struct number {

int a;

int b;

};

void \*addition ( void \*s){

struct number \*z = (struct number \*)s;

printf("Adding %d and %d gives %d\n",z->a,z->b,z->a + z->b);

pthread\_exit(NULL);

}

int main(){

int result;

pthread\_t threads[5];

struct number k[5];

for ( int i = 0; i < 5; i++ ){

scanf("%d", &k[i].a);

scanf("%d", &k[i].b);

result = pthread\_create(&threads[i], NULL, addition,&k[i]);

if (result) {

printf("Error in creating thread, %d ", result);

exit(-1);

}

}

pthread\_exit(NULL);

}

Output:

