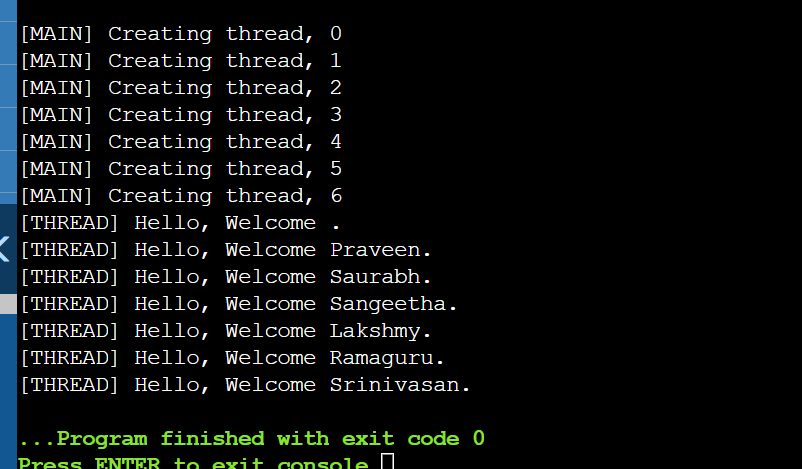
Os lab on C Threads:

Output for the given code:



Modified Code:

#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 \*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 names[10][15] = {"Amritha","Praveen","Saurabh","Sangeetha","Lakshmy","Srinivasan","Ramaguru"};

int result;

for(int i = 0; i < 6; 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 \*)&threads[i]);

if (result) {

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

exit(-1);

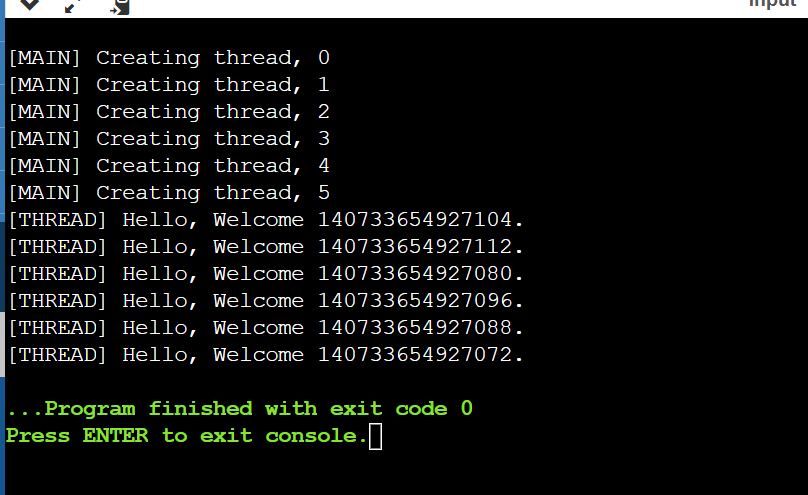
}

}

// Exit the thread

pthread\_exit(NULL);

}



Adding two numbers using threads and struct:

/\*

@Author: Ramaguru Radhakrishnan

@Date: 21 - Dec - 2022

@Description: Creation and Execution of a simple thread

Addition:

after #....

Struct argfun{

int a;

int b;

char name[10];

};

void \*addition(void \*arg)

{

struct argfunc \*obj=arg;

int c=obj->a + obj->c;

printf("%d",c);

}

int main()

{

struct argfunc mobj;

mobj.a=5;

mobj.b=10;

mobj.name=add;

pthread\_create(&thread, NULL,addition, &mobj[])

}

Struct VariableToAdd

{

int a;

int b;

}

P----(&thread, NULL, addition,&variableToAdd)

\*/

#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 \*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 names[10][15] = {"Amritha","Praveen","Saurabh","Sangeetha","Lakshmy","Srinivasan","Ramaguru"};

int result;

for(int i = 0; i < 6; 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 \*)&threads[i]);

if (result) {

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

exit(-1);

}

}

// Exit the thread

pthread\_exit(NULL);

}