Exp No:3 Title of the Exercise: Multithreading and pthread in C

Date: --13/09/2022

Aim: --

Creation of Multithreading using posix threads in C

Procedure: --

Thread:

A Thread is a single sequence stream within a process. In short thread is a unit of process.

Multithreading:

Multithreading is the ability of a program or an operating system to enable more than one user at a time without requiring multiple copies of the program running on the computer. Multithreading can also handle multiple requests from the same user.

POSIX threads:

The POSIX thread libraries are a C/C++ thread API based on standards. It enables the creation of a new concurrent process flow. It works well on multi-processor or multi-core systems, where the process flow may be scheduled to execute on another processor, increasing speed through parallel or distributed processing. Because the system does not create a new system, virtual memory space and environment for the process, threads need less overhead than “forking” or creating a new process. While multiprocessor systems are the most effective, benefits can also be obtained on uniprocessor systems that leverage delay in I/O and other system processes that may impede process execution

CODE: --

#include <stdio.h>

#include <stdlib.h>

#include <unistd.h>

#include <pthread.h>

void \*threadCreation(void \*vargp)

{

sleep(1);

printf("A new thread is created \n");

return NULL;

}

int main()

{

pthread\_t thread\_id;

printf("Before Thread\n");

pthread\_create(&thread\_id, NULL, threadCreation, NULL);

pthread\_join(thread\_id, NULL);

printf("The Thread ID is of thread is %ld\n",thread\_id);

printf("After Thread\n");

exit(0);

}

Output: --



Result: --

A thread is created using Multithreading concept using posix threads.