NAME: G. GANESH

REG NO: 192373008

EXERICSE-11

Illustrate the concept of multithreading using a C program.

Aim:

To illustrate the concept of multithreading using a C program.

Algorithm:

1. Include the necessary headers for multithreading.
2. Define a function to be executed by threads.
3. Create multiple threads using pthread_create.
4. Perform operations within each thread.
5. Synchronize threads, if necessary, using pthread_join or other synchronization techniques.
6. Compile and execute the program to observe concurrent execution.

Procedure:

1. Define a thread function to perform a specific task.
2. Create multiple threads using pthread_create and pass arguments if needed.
3. Wait for threads to complete using pthread_join.
4. Compile and run the program to observe the results.

Code:

```
#include <stdio.h>
#include <pthread.h>
#include <unistd.h>
void *thread_function(void *arg) {
    int thread_num = *(int *)arg;
    printf("Thread %d is running\n", thread_num);
    sleep(1);
    printf("Thread %d has finished\n", thread_num);
  return NULL;
```

```
}
int main() {
    pthread_t threads[3];
    int thread_args[3];
    for (int i = 0; i < 3; i++) {
        thread_args[i] = i + 1;
        if (pthread_create(&threads[i], NULL, thread_function, &thread_args[i]) != 0) {
            perror("Failed to create thread");
            return 1;
        }
    }
    for (int i = 0; i < 3; i++) {
        if (pthread_join(threads[i], NULL) != 0) {
            perror("Failed to join thread");
            return 1;
        }
    }
    printf("All threads have finished execution\n");
    return 0;
}
```

Result:

The concept of multithreading was successfully demonstrated. Multiple threads executed concurrently, performing tasks and completing execution independently.

Output: