**EX 11: Illustrate the concept of multithreading using a C program.**

**Aim:**

To illustrate the concept of multithreading by writing a C program that demonstrates the simultaneous execution of multiple threads performing independent tasks.

**Algorithm:**

1. **Start**.
2. **Initialize threads**:
   * Include the necessary headers (pthread.h and stdio.h).
   * Define the number of threads required.
3. **Define thread functions**:
   * Create functions for each thread, specifying the task they will perform.
4. **Create threads**:
   * Use the pthread\_create function to initialize and execute threads.
5. **Wait for threads to finish**:
   * Use pthread\_join to ensure the main program waits for the threads to complete.
6. **Print results**:
   * Display messages from each thread to indicate their execution.
7. **Terminate**:
   * Exit the program once all threads have completed execution.

**Program:**

#include <stdio.h>

#include <stdlib.h>

#include <pthread.h>

void\* print\_message(void\* message) {

printf("%s\n", (char\*)message);

return NULL;

}

int main() {

pthread\_t thread1, thread2;

const char\* message1 = "Hello from Thread 1";

const char\* message2 = "Hello from Thread 2";

pthread\_create(&thread1, NULL, print\_message, (void\*)message1);

pthread\_create(&thread2, NULL, print\_message, (void\*)message2);

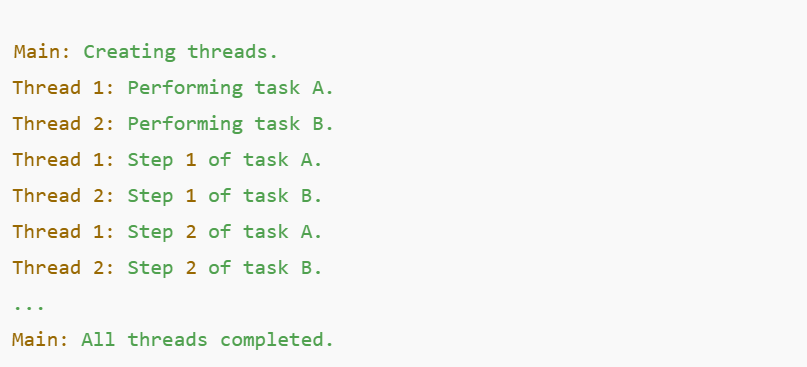
pthread\_join(thread1, NULL);

pthread\_join(thread2, NULL);

return 0;

}

**OUTPUT:**

****