13. Illustrate the concept of multithreading using a C program.

Program:

```
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>

// Function executed by the first thread
void* threadFunction1(void* arg) {
   for (int i = 0; i < 5; i++) {
      printf("Thread 1: Working on task %d\n", i + 1);
   }
   pthread_exit(NULL);
}

// Function executed by the second thread
void* threadFunction2(void* arg) {
   for (int i = 0; i < 5; i++) {
      printf("Thread 2: Processing job %d\n", i + 1);
   }
   pthread_exit(NULL);
}

int main() {
   pthread_t thread1, thread2;

   // Creating thread 1
   if (pthread_create(&thread1, NULL, threadFunction1, NULL) != 0)
{
      perror("Failed to create thread 1");
      return 1;
   }

   // Creating thread 2
   if (pthread_create(&thread2, NULL, threadFunction2, NULL) != 0)
{
      perror("Failed to create thread 2");
      return 1;
   }

   // Wait for threads to complete
   pthread_join(thread1, NULL);
   pthread_join(thread2, NULL);
```

```
    printf("Both threads have completed execution.\n");
    return 0;
}
```

Output: