NAME: G. GANESH

REG NO: 192373008

EXERICSE-30

Write C programs to demonstrate the following thread related concepts.
(i)create (ii) join (iii) equal (iv) exit

AIM:

To write a C program demonstrating the following thread-related concepts:

i. **create** - Creating threads.
ii. **join** - Joining threads.
iii. **equal** - Checking if two threads are equal.
iv. **exit** - Exiting a thread.

Algorithm:

1 . Create Thread:

- Use pthread_create() to create a new thread and specify the function to execute.

2. Join Thread:

- Use pthread_join() to make the main thread wait for the created thread to finish.

3. Equal Threads:

- Use pthread_equal() to compare two thread IDs to check if they are the same.

4. Exit Thread:

- Use pthread_exit() to terminate a thread explicitly.

Procedure:

1. **Create a Thread:**
   - Define a thread function to be executed.
   - Call pthread_create() to create the thread.
2. **Join Threads:**
   - The main thread waits for the created thread to finish by using pthread_join().
3. **Check Equal Threads:**
   - Compare two thread IDs with pthread_equal().
4. **Exit the Thread:**
   - Each thread can call pthread_exit() to terminate.

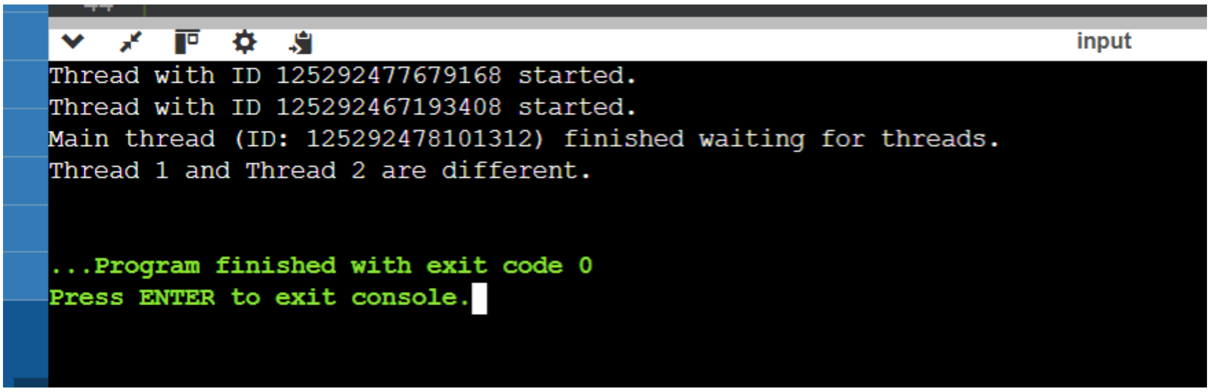
Code:

```
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
void* thread_function(void* arg) {
    printf("Thread with ID %lu started.\n", pthread_self());
    return NULL;
}
int main() {
    pthread_t thread1, thread2;
    pthread_t current_thread = pthread_self(); // Store the main thread ID
    if (pthread_create(&thread1, NULL, thread_function, NULL)) {
        printf("Error creating thread 1\n");
        return 1;
    }
    if (pthread_create(&thread2, NULL, thread_function, NULL)) {
        printf("Error creating thread 2\n");
        return 1;
    }
    pthread_join(thread1, NULL);
    pthread_join(thread2, NULL);
    printf("Main thread (ID: %lu) finished waiting for threads.\n", current_thread);
    if (pthread_equal(thread1, thread2)) {
        printf("Thread 1 and Thread 2 are the same.\n");
    } else {
        printf("Thread 1 and Thread 2 are different.\n");
    }
    pthread_exit(NULL);
    return 0;
}
```

Result:

The program demonstrates thread creation, joining, equality check, and thread exit, where two threads are created, joined, compared for equality, and the main thread exits after all operations are complete.

Output:



```
Thread with ID 125292477679168 started.
Thread with ID 125292467193408 started.
Main thread (ID: 125292478101312) finished waiting for threads.
Thread 1 and Thread 2 are different.


...Program finished with exit code 0
Press ENTER to exit console.
```