EX NO: 8A	**IMPLEMENTATION OF THREADING AND SYNCHRONIZATION APPLICATIONS**
DATE:	

AIM:

To implement threading and synchronization techniques using c language.

ALGORITHM:

Step 1: Start the program.
Step 2: Identify the thread by an id called ThreadId.
Step 3: Represent the thread id by the type pthread_t.
Step 4: Include the header file “#include<pthread.h>” to access the thread functions. Step
5: This function is used to create a thread pthread_create.
Step 6: If the thread is created successfully, return value will be zero,
Otherwise pthread_create will return an error number of type integer. Step 7:
Stop the program.

PROGRAM:

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

/*thread function definition*/
void* threadFunction(void* args)
{
while(1)
   {
      printf("I am threadFunction.\n");
   }
} int
main()
{
   /*creating      thread
id*/        pthread_t  id;
int ret;

   /*creating thread*/

ret=pthread_create(&id,NULL,&threadFunction,NULL);
if(ret==0){
      printf("Thread created successfully.\n");
   }    else{        printf("Thread
not created.\n");
      return 0; /*return from main*/
   }

   while(1)
   {
      printf("I am main function.\n");
   }

   return 0;
}
```

OUTPUT:

```
Thread created successfully.
I am threadFunction.
I am threadFunction.
I am threadFunction.
I am threadFunction.
…
…
I am threadFunction.
I am main function.
I am main function.
I am main function.
I am main function.
…
…
I am main function.
I am threadFunction.

… and so on.
```

RESULT:

Thus to implement threading and synchronization techniques using c language has been executed and verified successfully.