

**Alunos:** Chrystian de Sousa Guth  
Lucas Pereira da Silva  
**Disciplina:** Sistemas Operacionais I – INE5412  
**Professor** Luis Fernando Friedrich

**Matrículas:** 10103131  
10100754

## Roteiro de Laboratório 4

### Resposta

Sim, é possível utilizando o comando `pthread_join()`. Essa função suspende a thread que a chamou até que a thread passada por parâmetro execute o comando `pthread_exit()`.

### Código final em C

```
#include <pthread.h>
#include <stdio.h>
#include <stdlib.h>
#define NUMBER_OF_THREADS 10
void *print_hello_world(void * tid){
    printf("Olá, Aqui é a thread %d\n", tid);
    pthread_exit(NULL);
}
int main(int argc, char *argv[]){
    pthread_t threads[NUMBER_OF_THREADS];
    int status, i;
    for(i = 0; i < NUMBER_OF_THREADS; i++){
        printf("Aqui é o Main. Criando a thread %d\n", i);
        status = pthread_create(&threads[i], NULL, print_hello_world, (void *)i);
        pthread_join(threads[i], NULL); // o primeiro argumento é a thread que o programa vai esperar até que continue o fluxo
        if(status != 0){
            printf("Opa! pthread_create retornou o erro %d\n", status);
            exit(-1);
        }
    }
    exit(NULL);
}
```