Exercício:

Considere o seguinte programa que utiliza threads para realizar a multiplicação de matrizes:

Obs: criar um projeto incluir os três arquivos

Arquivo func.h

```
typedef struct {
    int tam, lin, col;
    int (*MA)[8], (*MB)[8], (*MC)[8];
} matriz;

int multiplica (void *m);
```

Arquivo func.c

```
#include "func.h"

int multiplica (void *m) {
    matriz *dados = (matriz*) m;
    int i;
    int lin = dados->lin;
    int col = dados->col;
    dados->MC[lin][col] = 0;

    for (i=0; i<dados->tam; i++) {
        dados->MC[lin][col] += dados->MA[lin][i] * dados->MB[i][col];
    }
    return 1;
}
```

Arquivo main.c

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


int main() {
   int lin, col, tam=8, i;
   int MA[8][8], MB[8][8], MC[8][8];
   matriz *dados;
   pthread_t thrs[8*8];
   for (lin=0; lin<tam; lin++) {
      for (col=0; col<tam; col++) {
         dados = (matriz *) malloc (sizeof(matriz));
         dados->tam = tam;
         dados->lin = lin;
         dados->col = col;
         dados->MA = MA;
         dados->MB = MB;
         dados->MC = MC;

         pthread_create(&thrs[col+lin*8], NULL, multiplica, &dados);
      }
   }

   for (i=0; i<tam*tam; i++) {
      pthread_join(thrs[i], NULL);
   }
   return 0;
}
```

1. Explique como funciona o programa: quantas threads são criadas e como cada uma é chamada.
2. Relate a diferença desta solução com a que você adotou no exercício
3. Inclua um contador de tempo e informe o tempo total de execução deste programa
4. Faça com que execute com apenas uma thread e observe a diferença dos tempos.
5. Compare-o com o programa da aula passada e com o programa de apenas uma thread e procure justificar a diferença entre os tempos. Também relate em qual das implementações foi utilizada mais de uma CPU
6. Relate a importância do uso da função pthread_join no final.