



# Projeto 1 - Aplicação Multithread

Cristiano (256352), George (216741),  
Jhonatan (256444) e Mylena (222687)

# TABELA DE CONTEÚDOS

01

Tema do  
Projeto

02

Estrutura  
de Dados  
Principal

03

Detalhes da  
Implementação

04

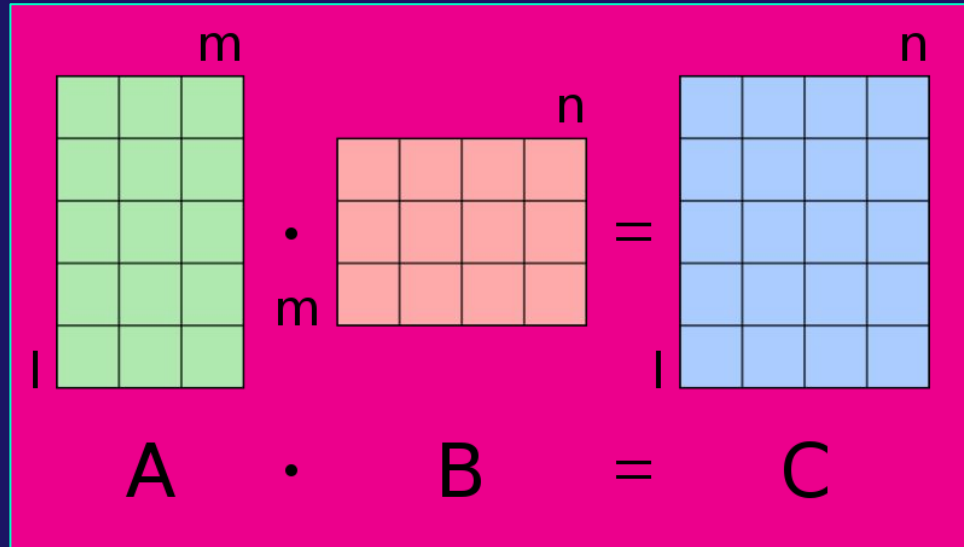
Demonstração



01

Tema do Projeto

# MULTIPLICAÇÃO DE MATRIZES





02

# Estrutura de Dados Principal

# STRUCT MATRIX

```
typedef struct matrix matrix_t;  
...  
typedef unsigned long long uli;  
  
struct matrix  
{  
    int *mat;  
    uli rows, cols;  
};
```



03

## Detalhes da Implementação

# ENTRADAS PSEUDO-ALEATÓRIAS DE TAMANHOS PRÉ-DEFINIDOS

```
// 1.in
```

```
15 10      → matriz A  
10 20      → matriz B  
3243       → seed
```



# ENTRADAS PSEUDO-ALEATÓRIAS DE TAMANHOS PRÉ-DEFINIDOS

```
#define MOD_N 10
...

void initialize_matrices(uli ra, uli
ca, uli rb, uli cb, uli seed)
{
    srand(seed);

    A = allocate_matrix(ra, ca);
    ...

    for(uli i = 0; i < ra; i++)
        A→mat[i] = rand() % MOD_N;

    ...
}
```

# ARMAZENAMENTO CONTÍNUO

```
matrix_t *allocate_matrix(uli rows,
uli cols)
{
    matrix *m;
    m = (matrix_t *) malloc
(sizeof(matrix_t));
    m->mat = (int *) malloc
(rows*cols*sizeof(int));
    m->rows = rows;
    m->cols = cols;
    return m;
}
```

# CRIAÇÃO DAS THREADS

```
...
typedef struct thread_data
thread_data_t;
typedef unsigned long long uli;

...

struct thread_data
{
    uli row;
};
```

# CRIAÇÃO DAS THREADS

```
...
```

```
pthread_t t_ids[rows_a];  
thread_data_t t_data[rows_a];  
  
for(uli i = 0; i < rows_a; i++)  
{  
    t_data[i].row = i;  
    pthread_create(&t_ids[i], NULL,  
multiply, &t_data[i]);  
}
```

1000 JOURNAL OF CLIMATE



04

Demonstração

# CASOS DE TESTE

- Matrizes impróprias para multiplicação;
- Matrizes quadradas;
- Matrizes não quadradas.

# OBRIGADO!

Alguma pergunta?

CREDITS: This presentation template was created by Slidesgo, including icons by Flaticon, and infographics & images by Freepik.