

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA/
PIAUI
Campus Teresina - Central

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA
E TECNOLOGIA DO PIAUÍ

CAMPUS TERESINA-CENTRAL

DIRETORIA DE ENSINO

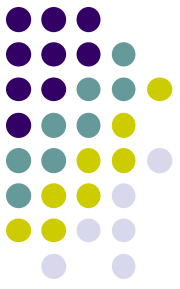
Estrutura de Dados

Aula 4.1 – Filas

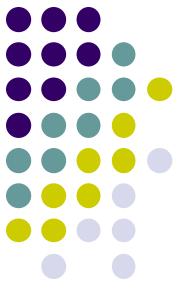
Exemplo Coloração de Área

Professora: Elanne Cristina O. dos Santos

elannecristina.santos@gmail.com
elannecristina.santos@ifpi.edu.br



- Para realizar a **coloração de uma área de pixels** é possível usar o algoritmo de **flood fill com fila (BFS)**.
- Exemplo de aplicação: balde de tinta no Paint.
- **Busca em Largura**. É um algoritmo usado para explorar estruturas de dados como **grafos** ou **matrizes**, visitando os elementos **em camadas**, ou seja, primeiro os vizinhos mais próximos, depois os vizinhos dos vizinhos, e assim por diante.



- **Estrutura geral**

- Representamos a imagem como uma matriz 2D de inteiros (cada número representa uma cor).
- Usamos uma fila para explorar os pixels vizinhos (BFS).
- A função colore todos os pixels conectados à posição inicial que tenham a mesma cor original.



- **Resultado esperado**

- **Antes:**

1	1	1	2	2
1	1	0	2	2
1	0	0	2	2
1	1	0	2	2
1	1	1	2	2

- **Depois:** preencher a partir de (2,2) com cor 9:

1	1	1	2	2
1	1	9	2	2
1	9	9	2	2
1	1	9	2	2
1	1	1	2	2



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

#define MAX 100

typedef struct {
    int x, y;
} Ponto;

typedef struct {
    Ponto dados[MAX * MAX];
    int frente, tras;
} Fila;

void inicializarFila(Fila* f) {
    f->frente = f->tras = 0;
}

int filaVazia(Fila* f) {
    return f->frente == f->tras;
}
```



```
void enfileirar(Fila* f, int x, int y) {  
    f->dados[f->tras].x = x;  
    f->dados[f->tras].y = y;  
    f->tras++;  
}
```

```
Ponto desenfileirar(Fila* f) {  
    return f->dados[f->frente++];  
}
```

```
void floodFill(int imagem[MAX][MAX], int linhas,
int colunas, int x, int y, int novaCor) {
    int corOriginal = imagem[x][y];
    if (corOriginal == novaCor) return;
    Fila fila;
    inicializarFila(&fila);
    enfileirar(&fila, x, y);
    while (!filaVazia(&fila)) {
        Ponto p = desenfileirar(&fila);
        int i = p.x;
        int j = p.y;
        if (i < 0 || i >= linhas || j < 0 || j >= colunas) continue;
        if (imagem[i][j] != corOriginal) continue;
        imagem[i][j] = novaCor;
        enfileirar(&fila, i + 1, j);    enfileirar(&fila, i - 1, j);
        enfileirar(&fila, i, j + 1);    enfileirar(&fila, i, j - 1);    }}
}
```

```
int main() {  
    int imagem[MAX][MAX] = {  
        {1, 1, 1, 2, 2},  
        {1, 1, 0, 2, 2},  
        {1, 0, 0, 2, 2},  
        {1, 1, 0, 2, 2},  
        {1, 1, 1, 2, 2}  
    };  
    int linhas = 5, colunas = 5;  
    int x = 2, y = 2;    // Ponto inicial  
    int novaCor = 9;    // Cor para preencher  
    printf("Imagem original:\n");  
    .....  
    printf("\nImagem após preenchimento:\n");  
    .....  
    return 0;  }
```