Documentação:(Atividade 1: Componentes f-conexas)

Diogo Honorato

Setembro 2023

1 Introdução

Este documento apresenta a documentação do trabalho realizado na linguagem C,no sistema operacional Linux, para construir a estrutura de dados e manipulação de grafos, destaco que desenvolvi uma biblioteca de minha autoria.

2 Compilação

Para compilar o projeto, certifique-se de ter o programa "make"instalado em seu sistema. Se você já possui o "make"instalado, siga estas instruções:

-Abra um terminal.

-Navegue até o diretório onde está localizado o código-fonte do projeto.

-Digite o seguinte comando no terminal:"make".

-Para limpar os arquivos gerados digite no terminal: "make clean".

3 Instruções para execução

Depois de ter compilado o programa será gerado um arquivo executável no mesmo local onde se encontra a 'main.c' com o nome de "grafo". Digite no terminal o seguinte comando: "./grafo", será impresso no terminal a saída esperada.

4 Principais funções

-Será necessário utilizar primeiro a função para alocar um grafo,utilizando a seguinte função:
'Graph *G = create_graph();'

-Para uma entrada de um arquivo texto será necessário utilizar a seguinte função:
'EdgeFileReader(nome do grafo alocado,seu arquivo.txt);'

-Para gerar os subgrafos f-conexos será necessário utilizar a seguinte função:
'SubGrafoF_Conexo(o nome do seu grafo);'

Exemplo:

```c
int main()
{
   Graph *G = create_graph();

   EdgeFileReader(G,"testes/grafo.txt");

   SubGrafoF_Conexo(G);

}
```

5 Formato da entrada

Os arquivos de texto para teste estão na pasta "testes". Para inserir o grafo desejado basta apenas digitar suas conexões no arquivo de texto com o nome de "grafo.txt", lembrando que cada linha possui duas colunas: a primeira referente a um vértice de saída e a segunda referente a um vértice de entrada, o programa aceita apenas entradas no formato de números, o conteúdo do arquivo texto de entrada ficaria assim:

1 3
2 1
3 2
3 4
4 2

6 Resultados

Testes e saídas esperadas do programa:

Primeira entrada.
Entrada 1:
1 3
2 1
3 2
3 4
4 2
Saída esperada 1:
Componente f-conexa 1:
1 3 4 2

Segunda entrada.
Entrada 2:
1 2
1 3
2 4
2 6
3 4
4 3
5 1
5 2
6 5
7 8
7 5
8 7
8 6
Saída esperada 2:
Componente f-conexa 1:
1 2 6 5
Componente f-conexa 2:
3 4
Componente f-conexa 3:
7 8

Terceira entrada.
Entrada 3:
1 2
2 4
3 5
3 1
4 5
4 3
5 1
Saída esperada 3:
Componente f-conexa 1:
1 2 4 3 5