

Trabalho 04 - Componentes Fortemente Conexas

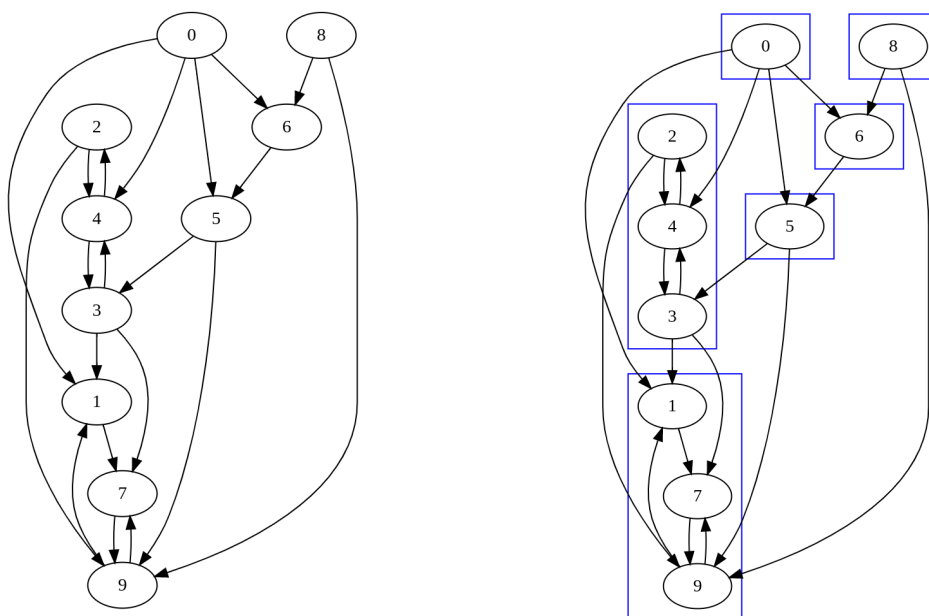
Data de entrega: 13/11/2020

Importante:

- **Não** olhe códigos de outros grupos ou da internet. Exceto o que é fornecido.
- TODOS os membros do grupo devem participar e compreender completamente a implementação.
- Em caso de plágio, fraude ou tentativa de burlar o sistema será aplicado nota 0 na disciplina aos envolvidos.
- Alguns alunos podem ser solicitados para explicar com detalhes a implementação.
- Passar em todos os testes do run.codes não é garantia de tirar a nota máxima. Sua nota ainda depende do cumprimento das especificações do trabalho, qualidade do código, clareza dos comentários, boas práticas de programação e entendimento da matéria demonstrada em possível reunião.
- Você deverá submeter, até a data de entrega, o seu código na plataforma run.codes, onde o “Número de Matrícula” deverá ser o número do seu grupo.

Esse trabalho deverá ser realizado em grupo, com os grupos já definidos na disciplina. Neste trabalho o seu grupo deverá encontrar componentes fortemente Conexas em Grafos Direcionados, utilizando o Algoritmo de Kosaraju.

Uma componente fortemente conexa (*Strongly Connected Components - SCC*) é um subconjunto maximal dos vértices tal que qualquer par de vértices u e v desse subconjunto tem um caminho (direcionado) de u para v , e de v para u . O exemplo abaixo apresenta, na imagem da esquerda, um grafo direcionado com 10 vértices. Na imagem da direita estão destacadas as SCCs do grafo.



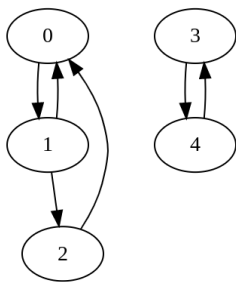
Neste trabalho o seu programa deverá ler, da entrada padrão do sistema, um grafo apresentado no seguinte formato: Dois valores inteiros, representando o número n de

vértices e o número m de arcos, então m linhas contendo dois valores representando o início e o fim de cada arco. Os vértices estão rotulados de 0 até $n-1$.

Veja um exemplo de entrada:

```
5 6
0 1
1 0
1 2
2 0
3 4
4 3
```

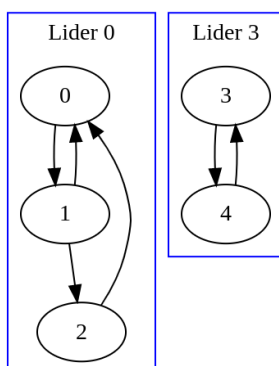
A entrada acima representa um grafo direcionado com 5 vértices e 6 arcos, o primeiro começa no vértice 0 e termina no vértice 1, e assim por diante. A figura abaixo ilustra essa entrada.



Seu programa deverá imprimir n valores, um para cada vértice, representando o líder da SCC a qual pertence aquele vértice. O líder de cada SCC deverá ser o vértice com o menor índice (para facilitar a impressão, existe um espaço depois do último valor e uma quebra de linha). Para a entrada acima seu programa deverá imprimir:

```
0 0 0 3 3
```

Essa saída representa as seguintes SCCs:



Você receberá um arquivo *kosaraju.c* que contém uma implementação (rústica) que representa um grafo direcionado usando lista de adjacência, e algumas funções auxiliares que podem ajudar vocês na implementação. Você pode usar o que quiser desse código, mas não é obrigatório.

- Você pode implementar em qualquer linguagem aceita pelo run.codes.
- Seu programa deve executar no run.codes em poucos segundos.
- Você não pode usar nenhuma biblioteca que representa grafos ou funções em grafos. Você pode usar outras funções caso precise (funções do `math.h`, funções de ordenação, etc)
- Se você não tiver certeza se alguma coisa é permitida ou não no trabalho, não hesite em perguntar ao professor!