

Especificação do Primeiro Trabalho (EP1)

Composição dos grupos: de 1 a 3 alunos .

O trabalho consiste em desenvolver um programa em C que recebe um grafo orientado e encontra as componentes fortemente conectadas usando as duas versões mostradas em sala de aula do algoritmo de Kosarayu e usando os métodos e estruturas de dados utilizadas em sala de aula. Os grafos serão representados como uma coleção de listas de adjacências.

Além disso, o programa deverá imprimir:

1. Se o grafo é fortemente conexo
2. O número de componentes fortemente conectados
3. Uma ordenação topológica do grafo de componentes fortemente conectados
4. A representação em texto do grafo de componentes fortemente conectados

Deverá ser implementada também uma função para gerar um grafo aleatório com $|V|$ vértices que têm $|A|$ arestas em média. Ver a seção “Outro construtor de grafos aleatórios” de:

https://www.ime.usp.br/~pf/algoritmos_para_grafos/aulas/random.html

O projeto deverá, ainda, demonstrar a habilidade do grupo em implementar os algoritmos de maneira eficiente e clara.

Adicionalmente, deverá ser mostrada uma descrição detalhada de um problema da vida real que pode ser modelado com grafos dirigidos e resolvido encontrando os componentes fortemente conexos. Para tal, deverá ser incluído o código necessário e mostrar um caso de teste com essa aplicação (2 pontos).

Entrada

A entrada conterá a especificação do grafo orientado. A primeira linha do texto conterá a quantidade de vértices do grafo, $|V|$. Para cada vértice listamos os nós adjacentes de cada vértice em cada linha, separando cada **vértice adjacente** por “;”. E a última linha especifica a versão do algoritmo de Kosarayu a ser utilizada, que pode ser: 1 ou 2.

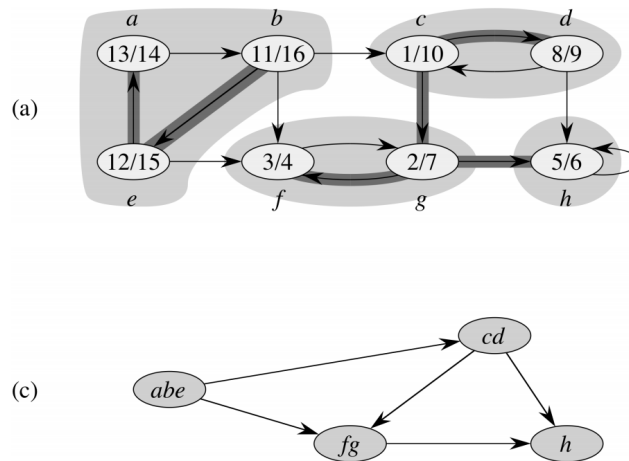


Figura 1: a) Grafo dirigido e b) Grafo de componentes fortemente conectados relacionado com o caso de teste 1

A entrada para o grafo da Figura 1.a é:

```
8
a: b;
b: c; e; f;
c: d; g;
d: c; h;
e: a; f;
f: g;
g: f; h;
h: h;
1
```

Saída

A saída conterá as seguintes linhas.

A primeira linha conterá "Sim" se o grafo é fortemente conexo e "Não" caso contrário.

A segunda linha conterá o número de componentes fortemente conectados.

A terceira linha conterá uma ordenação topológica do grafo de componentes fortemente conectados criado usando a versão 1 ou 2 do algoritmo de Kosarayu, dependendo do valor da última linha da entrada.

As demais linhas conterão a representação em texto do grafo de componentes fortemente conectados seguindo o mesmo formato da entrada.

Para a entrada apresentada acima, uma saída possível é dada a seguir:

Não

4

abe cd fg h

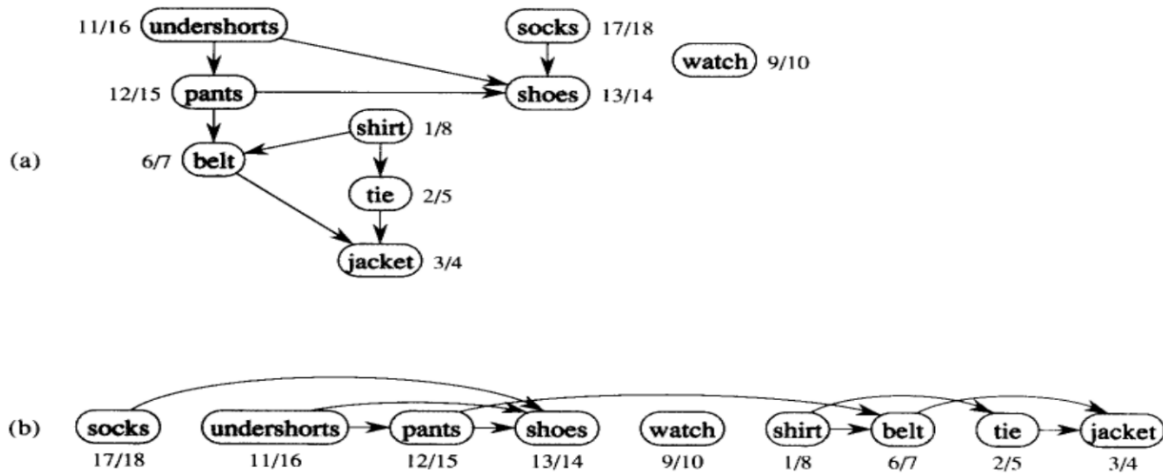
abe: cd; fg;

cd: fg; h;

fg: h;

h:

Caso de teste 2



Cormen. Pag 437.

Figura 2: a) Grafo dirigido e b) Grafo de componentes fortemente conectados relacionado com o caso de teste 2 e 3

A entrada e uma possível saída para o grafo da Figura 2.a é:

Entrada

9

undershorts: pants; shoes;

pants: belt; shoes;

belt: jacket;

shirt: belt; tie;

tie: jacket;

jacket:

socks: shoes;

shoes:

watch:

1

Saída

Não

9

socks undershorts pants shoes watch shirt belt tie jacket

undershorts: pants; shoes;

pants: belt; shoes;

belt: jacket;

shirt: belt; tie;

tie: jacket;

jacket:

socks: shoes;

shoes:

watch:

Caso de teste 3

A entrada e uma possível saída para o grafo da Figura 2.a é:

Entrada

9

undershorts: pants; shoes;

pants: belt; shoes;

belt: jacket;

shirt: belt; tie;

tie: jacket;

jacket:

socks: shoes;

shoes:

watch:

2

Saída

Não

9

socks undershorts pants shoes watch shirt belt tie jacket

undershorts: pants; shoes;

pants: belt; shoes;

belt: jacket;

shirt: belt; tie;

tie: jacket;

jacket:

socks: shoes;

shoes:

watch:

Caso de teste 4

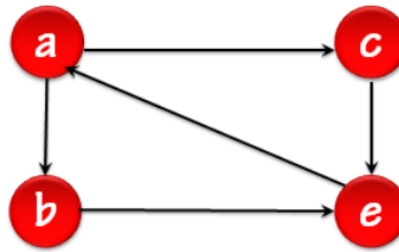


Figura 3: a) Grafo dirigido relacionado com o caso de teste 4

A entrada e saída para o grafo da Figura 4.a é:

Entrada

4

a: b; c;

b: e;

c: e;

e: a;

1

Saída

Sim

1

abce

abce:

Caso de teste 5

Gerar um grafo aleatório com 30 vértices com 500 arestas em média e gerar a saída correspondente para esse grafo usando a versão 1 ou a 2 do algoritmo de Kosarayu.

O que deve ser entregue:

- a) **Código desenvolvido pelo grupo:** o grupo deverá fazer o upload de todos os arquivos fonte no ambiente e-disciplinas.
- b) **Vídeo de apresentação:** o grupo deverá gravar e enviar no ambiente e-disciplinas um vídeo com duração de 10 a 12 minutos. Todos os integrantes do grupo devem participar da apresentação no vídeo.

Detalhes esperados no vídeo (listagem não exaustiva):

- Apresentação dos integrantes do grupo (nomes completos).
- Explicação detalhada do código.
- Demonstração da execução dos algoritmos nos 5 casos de teste.
- Descrição detalhada do problema da vida real escolhido.
- Descrição curta das dificuldades encontradas.