

MC102W - Algoritmos e Programação de Computadores

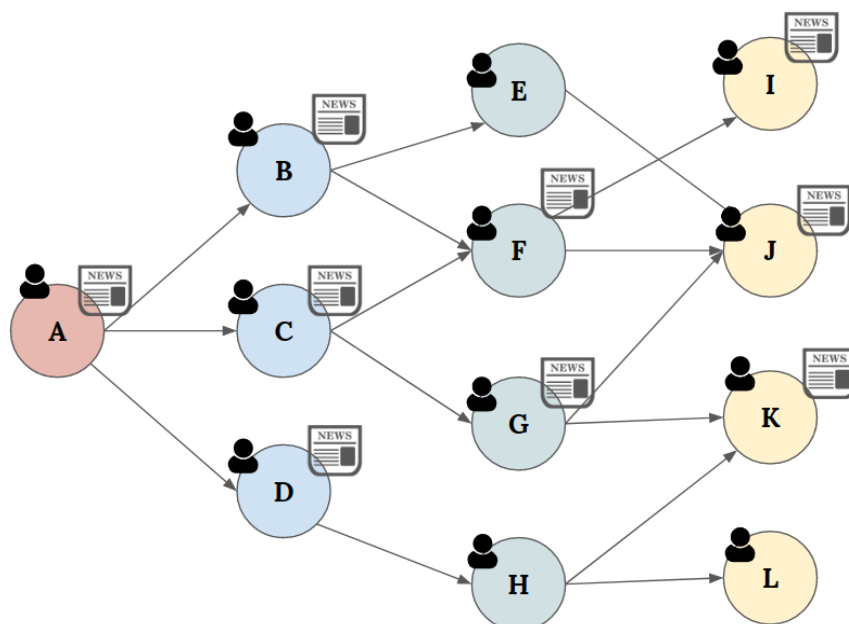
Lab14: Fake News

Prazo: 05 de Julho de 2020

Peso da Atividade: 5

Fake News são histórias que não possuem base factual mas são apresentadas como fatos, segundo Hunt et al. (2017). Tais histórias falsas podem ser espalhadas tanto via mídia tradicional como rádio e tv, quanto através da mídia online por meio das redes sociais.

O impacto da desinformação pode levar a consequências irreversíveis. Atualmente, existem grupos de pesquisa na área da computação que tentam criar mecanismos de busca, checagem e validação das informações compartilhadas com objetivo de denunciar e evitar o tráfego de informações falsas. No cenário de redes sociais, uma notícia falsa compartilhada pode atingir uma grande quantidade de pessoas de forma instantânea. O desafio para você é tentar mensurar quantas pessoas receberam fake news em um cenário de rede social onde várias pessoas estão ligadas e gerar um relatório com o impacto das **Fake News** em uma rede.



Tarefa

Sua tarefa neste laboratório é implementar um programa que retorna o impacto de uma Fake News através de um relatório de quem recebeu fake news (ordenado pelo nome).

- A fake news é gerada por apenas um usuário, no caso da figura acima o **A**, e espalhada pela rede de amigos. Somente quem gerou ou recebeu a fake news pode compartilhá-la (no exemplo, **A**, **C**, **F**, e **G** compartilharam).
- Quem compartilha a notícia falsa envia ela para todos os seus amigos (representado pela seta na figura).
- Com isso, o papel do programa é rastrear todos os que receberam e passaram a notícia falsa, incluindo o gerador. Ou seja, na figura acima seriam todos os usuários que contém o ícone do “News”.
- A busca dos usuários que receberam Fake News **deverá** ser feita através de **funções recursivas**.
- Os alunos **poderão** utilizar a função de ordenação que o Python já fornece.

Entrada

A entrada do programa será a quantidade de usuários da rede social e dados dos usuários:

Dados	Observação
Atuação do usuário	0 - Não compartilhou 1 - Gerador da Fake News 2 - Compartilhou
Nome	Não pode ser duplicado
Lista de amigos	<ul style="list-style-type: none">• Pode ter quantidades diferentes de amigos entre usuários diferentes• Amigos deve existir no sistema, ou seja, cadastrado

Os dados devem estar nesse formato:

n
<0- não compartilhou 1- gerador da Fake 2- compartilhou> <nome 1> <amigo 1>...<amigo k>
...
<0- não compartilhou 1- gerador da Fake 2- compartilhou> <nome n> <amigo 1>...<amigo k>

☐ *k* é diferente para cada entrada.

- ☐ *Sugestão:* Fazer os armazenamento das informações em dicionários e usar listas para guardar os amigos de alguém.
- ☐ Note que Ana (**A**) pode ser amiga de Beto (**B**) e Beto não ser amigo de Ana.

Saída

A saída do programa será o relatório de usuários que Fake News atingiu, incluindo quem gerou e compartilhou, ordenado por nome.

- ☐ Formato da saída:
Ordenação por nome
<nome 0>

...
<nome n>

Exemplo

Exemplo 1:

Entrada

```
12
1 A B C D
0 B E F
2 C F G
0 D H
0 E J
2 F I J
2 G J K
0 H K L
0 I
0 J
0 K
0 L
```

Saída

```
Ordenação por nome
A
B
C
D
F
G
I
J
K
```

Exemplo 2:

Entrada

```
6
1 Abel Beatriz Dilson
2 Beatriz Caio
0 Caio Marcio Marcelo
0 Dilson Caio
0 Marcio
0 Marcelo
```

Saída

```
Ordenação por nome
Abel
Beatriz
Caio
Dilson
```

Exemplo 3:

Entrada

```
5
1 Abel Beatriz Caio Dilson
0 Beatriz Caio
2 Caio Dilson
0 Dilson Carlos
0 Carlos
```

Saída

```
Ordenação por nome
Abel
Beatriz
Caio
Dilson
```

Exemplo 4:

Entrada

```
5
0 Carlos Jonathas
2 João
1 Camila Jonathas Helena
0 Jonathas Helena
0 Helena
```

Saída

```
Ordenação por nome
Camila
Helena
Jonathas
```

Critérios específicos

Os seguintes critérios específicos sobre o envio, implementação e execução devem ser satisfeitos.

i. Submeter no SuSy o arquivo:

⇒ **Lab14.py**: Arquivo contendo todo o seu programa.

ii. Não serão aceitas soluções contendo estruturas não vistas em sala, exceto as indicadas neste enunciado.

Observações gerais

No decorrer do semestre haverá 3 tipos de tarefas no SuSy (descritas logo abaixo). As tarefas possuirão os mesmos casos de testes abertos e fechados, no entanto o número de submissões permitidas e prazos são diferentes. As seguintes tarefas estão disponíveis no SuSy:

- ☐ **Lab14-AmbienteDeTeste**: Esta tarefa serve para testar seu programa no SuSy antes de submeter a versão final. Nessa tarefa, tanto o prazo quanto o número de submissões são ilimitados, porém os arquivos submetidos aqui **não serão corrigidos**.
- ☐ **Lab14-Entrega**: Esta tarefa tem limite de uma **única** submissão e serve para entregar a **versão final** dentro do prazo estabelecido para o laboratório. Não use essa tarefa para testar o seu programa e submeta aqui apenas quando não for mais fazer alterações no seu programa.
- ☐ **Lab14-ForaDoPrazo**: Esta tarefa tem limite de uma **única** submissão e serve para entregar a versão final fora prazo estabelecido para o laboratório. Esta tarefa irá substituir a nota obtida na tarefa **Lab14-Entrega** apenas se o aluno tiver realizado as correções sugeridas no *feedback* ou caso não tenha enviado anteriormente na tarefa **Lab14-Entrega**.

Avaliação

Este laboratório será avaliado da seguinte maneira: a nota será proporcional ao número de casos **fechados** para os quais o seu programa gerou a resposta correta, **desde que os critérios indicados neste enunciado tenham sido atendidos**. Se o programa apresentou resposta correta para todos os casos, a nota será 10; caso contrário será $p \cdot 10$, onde p é o percentual de respostas corretas. A nota também poderá sofrer descontos de acordo com a qualidade do programa apresentado. Assim, mesmo que o código seja capaz de resolver todos os casos de teste fechados, a nota final ainda pode ser menor do que 10. Por isso, acrescente comentários explicativos, utilize variáveis sugestivas e faça um código claro e de acordo com o que foi solicitado.

Testando seu programa

Para testar se a solução do seu programa está correta, basta seguir o exemplo abaixo no terminal do Linux.

```
python Lab14.py < arq01.in > arq01.out
diff arq01.out arq01.res
```

O `arq01.in` é a entrada e `arq01.res` é a saída esperada, ambos disponíveis no SuSy. O `arq01.out` é a saída gerada pelo seu programa. Após o prazo, os casos de teste fechados serão liberados e podem ser baixados e testados da mesma forma que os testes abertos.