

MC202 — ESTRUTURAS DE DADOS

Laboratório 14 — Grafos

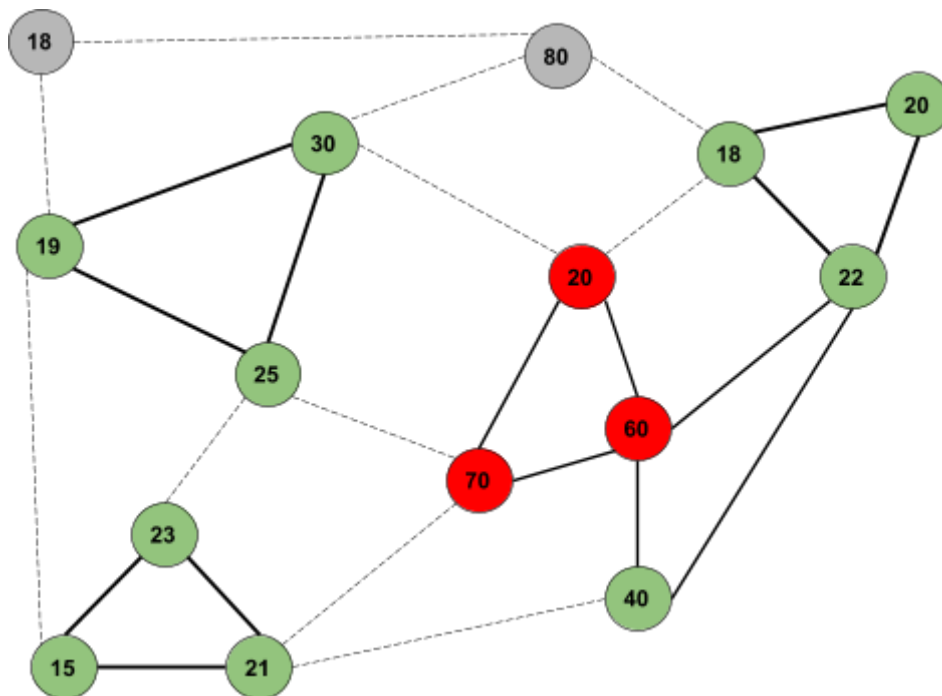
Tarefa

Um conjunto de pessoas de diferentes idades vai à Festa do Peão de Boiadeiro, que acontece anualmente na cidade de Barretos em São Paulo. Nessas festas, as pessoas costumam se reunir em grupos de três amigos. Acontece que algumas pessoas podem ficar entediadas dependendo do grupo que se formar. Assim, dizemos que um grupo de pessoas **está entediado** se a média das idades entre os três amigos do grupo exceder o dobro da idade de algum deles. Mais precisamente, seja x_i a idade da i -ésima pessoa que está na festa. Também, seja P um grupo de três pessoas que se conhecem mutuamente, então o cálculo da média é:

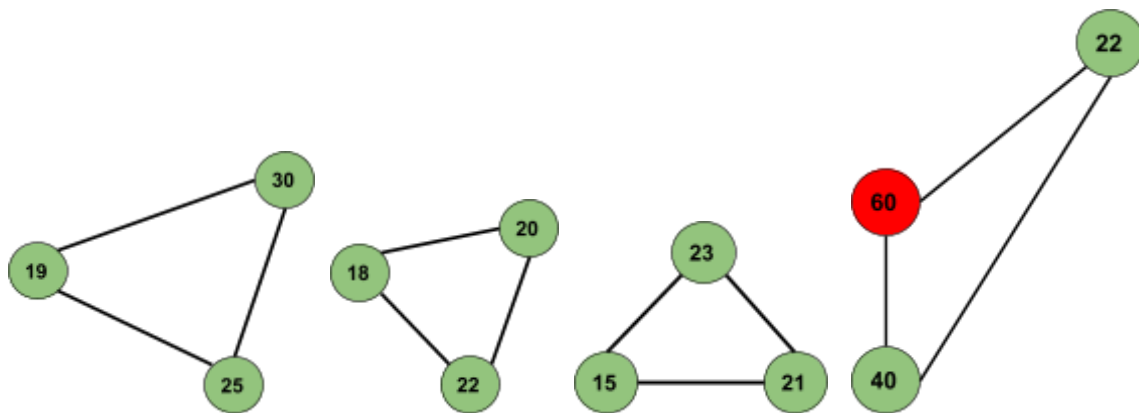
$$M = \frac{1}{3} \sum_{i \in P} x_i .$$

Dizemos que o grupo P está entediado se o valor de M for estritamente maior que $2x_i$, para alguma pessoa $i \in P$.

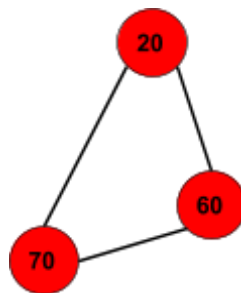
Sua tarefa será determinar o conjunto das pessoas que estão em algum grupo entediado. Veja o exemplo abaixo para um grupo de 15 pessoas, em que os números representam as idades.



Os grupos de amigos que não estão entediados são:



Nesse exemplo, há um único grupo de amigos que está entediado:



Repare que a pessoa com 60 anos está tanto em um grupo não entediado quanto em um grupo entediado e, portanto, fica entediada. Observe também que as pessoas de idades 18 e 80 não pertencem a nenhum grupo de três amigos, então não ficam entediadas.

Entrada

A primeira linha contém dois inteiros n e m , representando o número de pessoas e o número de relacionamentos, respectivamente. As pessoas são identificadas por índices de 0 a $n-1$. A segunda linha contém n inteiros separados por espaços, representando respectivamente as idades das pessoas. Cada uma das m linhas subsequentes contém dois inteiros, u e v , representando um relacionamento de amizade entre as pessoas u e v .

Saída

A saída deve conter os índices das pessoas que estão em algum grupo entediado. Os índices devem ser impressos em ordem crescente e cada um deve aparecer em uma linha uma única vez.

Exemplo

Entrada

```
15 26
18 80 30 18 20 19 20 22 25 70 60 23 15 21 40
0 1
0 5
1 2
1 3
2 5
2 6
2 8
3 4
3 6
3 7
4 7
5 8
5 12
6 9
6 10
7 10
7 14
8 11
8 9
9 10
9 13
10 14
11 12
11 13
12 13
13 14
```

Saída

```
6
9
10
```

Critérios específicos

- Para as turmas E e F, este laboratório tem peso 5.
- Para as turmas G e H, este laboratório tem peso 2.
- Deverão ser submetidos os seguintes arquivos: **lab14.c**, **grafo.h** e **grafo.c**.

- O grafo deve ser implementado utilizando **listas de adjacências**.
- Tempo máximo de execução: 1 segundo.

Testando

Para compilar com o Makefile fornecido e verificar se a solução está correta basta seguir o exemplo abaixo.

```
make  
./lab14 < arq01.in > arq01.out  
diff arq01.out arq01.res
```

onde `arq01.in` é a entrada (casos de testes disponíveis no SuSy) e `arq01.out` é a saída do seu programa. O Makefile também contém uma regra para testar todos os testes de uma vez; nesse caso, basta digitar:

```
make testar_tudo
```

Observações gerais

No SuSy, haverá 3 tipos de tarefas com siglas diferentes para cada laboratório de programação. Todas possuirão os mesmos casos de teste. As siglas são:

1. **SANDBOX:** Esta tarefa serve para testar o programa no SuSy antes de submeter a versão final. Nessa tarefa, tanto o prazo quanto o número de submissões são ilimitados, porém arquivos submetidos aqui **não serão corrigidos**.
2. **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: submeta aqui quando não for mais fazer alterações no seu programa.
3. **FORAPRAZO:** Esta tarefa tem limite de **uma única submissão** e serve para entregar a versão final após o prazo estabelecido para o laboratório, mas com nota reduzida (conforme a ementa). O envio nesta tarefa irá substituir a nota obtida na tarefa ENTREGA apenas se o aluno tiver realizado as correções sugeridas no feedback ou caso não tenha enviado anteriormente em ENTREGA.

Observações sobre SuSy:

- Versão do GCC: C-ANSI 4.8.2 20140120 (Red Hat 4.8.2-15).
- Flags de compilação:
`-ansi -Wall -pedantic-errors -Werror -g -lm`

- Utilize comentários do tipo `/* comentário */`;
comentários do tipo `//` serão tratados como erros pelo SuSy.

Além das observações acima, esse laboratório será avaliado pelos critérios gerais:

- Indentação de código e outras boas práticas, tais como:
 - uso de comentários (apenas quando forem relevantes);
 - código simples e fácil de entender;
 - sem duplicidade (partes que fazem a mesma coisa).
- Organização do código:
 - tipos de dados criados pelo usuário e funções bem definidas e tão independentes quanto possível.
- Corretude do programa:
 - programa correto e implementado conforme solicitado no enunciado;
 - eficiência do algoritmo (complexidade de tempo ou de espaço);
 - inicialização de variáveis sempre que for necessário;
 - desalocar toda memória alocada dinamicamente durante a execução do programa;
 - realizar leitura ou escrita em blocos de memória não alocados;
 - duplicidade de código;
 - má utilização de recursos (overhead de processamento ou de memória);
 - dentre outros critérios caso necessários.