

# MC558 - Complexidade de Algoritmos II

Primeiro semestre de 2025

## Laboratório 1

### Quem são meus amigos?

O Professor Sabin Ada, um famoso especialista em Teoria dos Grafos (e programador nas horas vagas) foi convidado a uma festa. Na festa havia  $n$  pessoas, digamos  $1, 2, \dots, n$ . Dr. Ada sabia quantos amigos cada pessoa tinha, mas não quem era amigo de quem.

O brilhante cientista então pensou numa solução para reconstruir o “grafo de amizade” das pessoas da festa, ao se lembrar da ideia de sequências gráficas. Sua tarefa é ajudar o Dr. Ada a implementar sua ideia. Para os propósitos deste trabalho, suporemos que se  $A$  é amigo de  $B$ , então  $B$  é amigo de  $A$ .

PS: Infelizmente, a ideia do Dr. Ada não é completamente satisfatória, pois pode haver mais de um modo de reconstruir o grafo. Para reconstruir exatamente o grafo, seria necessário mais informações do que ele tinha. Mas isto é assunto para outra disciplina...

### Sequências gráficas

Seja  $G$  um grafo com  $V(G) = \{v_1, \dots, v_n\}$  tal que  $d(v_i) \geq d(v_{i+1})$  para  $i = 1, 2, \dots, n-1$ . Chamamos a sequência  $(d(v_1), \dots, d(v_n))$  de **sequência de graus** de  $G$ . Dizemos que uma sequência não-crescente de inteiros não-negativos  $\mathbf{d} = (d_1, \dots, d_n)$  é uma **sequência gráfica** se existe algum **grafo simples**  $G$  cuja sequência de graus é exatamente  $\mathbf{d}$ . Por exemplo,  $(4, 4, 3, 3, 3, 3, 2)$  e  $(4, 3, 3, 3, 3, 2, 2)$  são sequências gráficas, mas  $(7, 6, 5, 4, 3, 3, 2)$  e  $(6, 6, 5, 4, 3, 3, 1)$  não são.

Seja  $\mathbf{d} = (d_1, \dots, d_n)$  uma sequência não-crescente de inteiros não-negativos. Seja  $\mathbf{d}' = (d_2 - 1, d_3 - 1, \dots, d_{d_1+1} - 1, d_{d_1+2}, \dots, d_n)$ . Pode-se mostrar que  $\mathbf{d}$  é uma sequência gráfica se e somente se  $\mathbf{d}'$ , após ordenado, é uma sequência gráfica (Visto em sala!). Usando este fato, projete um algoritmo que recebe uma sequência não-crescente de inteiros não-negativos  $\mathbf{d} = (d_1, \dots, d_n)$  e que devolve:

- um grafo **simples** descrito por listas de adjacências cuja sequência de graus é  $\mathbf{d}$ , ou
- responde **Não é sequência gráfica!**

## 1 Entrada e Saída

**Entrada:** cada caso de teste terá duas linhas como entrada. A primeira linha contém um inteiro  $n$  ( $1 \leq n \leq 500$ ) que corresponde ao número de elementos da sequência  $d$ . A segunda linha

contém os  $n$  inteiros não-negativos que correspondem à sequência  $d$ . Os elementos estão em ordem não-crescente e são separados por um espaço.

**Saída:** caso exista um grafo **simples** qualquer  $G$  com a sequência de graus exatamente igual à  $d$ , o programa deverá imprimir a lista de adjacências de  $G$  (note que há várias soluções possíveis). Os vértices devem ser rotulados por  $\{1, 2, \dots, n\}$  e cada lista deve estar **ordenada em ordem crescente**: a linha 1 deve conter os rótulos dos vértices adjacentes ao vértice 1, a linha 2 deve conter os rótulos dos vértices adjacentes ao vértice 2, e assim sucessivamente. Em cada linha os rótulos devem estar separados por um espaço. Se houver um vértice  $i$  com grau zero, o programa deve imprimir somente uma quebra de linha (`\n`) na linha  $i$ . Caso não exista grafo simples com sequência de graus  $d$ , o programa deverá imprimir apenas a mensagem **Não é sequência gráfica!** (com quebra de linha).

**Exemplos:**

Entrada	Saída
4 3 2 2 1	2 3 4 1 3 1 2 1
4 4 2 2 2	Não é sequência gráfica!

## 2 Implementação e Submissão

- A solução deverá ser implementada em C, C++ ou Python 3.
- O programa deve ser submetido no SuSy, com o nome principal **t1** (por exemplo, t1.c).
- O número máximo de submissões é 20.
- A tarefa contém 10 testes abertos e 10 testes fechados. A nota será proporcional ao número de acertos nos testes fechados.
- **Casos de plágio implicam em nota ZERO na disciplina para todos os envolvidos.**
- **Não é permitido o uso de bibliotecas que não sejam padrão, bem como diretivas ou flags de otimização.**

## 3 Prazo final de submissão

Sexta-feira 29 de Março às 4h da manhã.