

# Teoria de Grafos    Trabalho 2

## Enunciado

O trabalho consiste em implementar uma série de funções que serão colocadas no arquivo [grafosEspeciais2.jl](#), disponibilizado junto com os demais arquivos e cujos enunciados serão dados na próxima seção. Os códigos que implementam as estruturas de dados e as funções para manipulação de grafos encontram-se na sala virtual da disciplina no Microsoft Teams na aba **Arquivos>Material de Aula>Trabalho>2>Grafos**. Copie o diretório todo para sua máquina, **por cima** do diretório que usou no trabalho anterior. A ideia é que somente alguns arquivos serão modificados e seus arquivos do trabalho anterior ficarão intactos.

Antes de entregar o trabalho verifique se todo o código compila sem erros. Se existir algum erro que fizer o seu código não compilar, todo o trabalho será **zerado**. Leia o arquivo `Readme.md` dentro do diretório **Grafos** para maiores informações.

Sempre teste o seu código com os testes fornecidos. A corretude da sua implementação será verificada com testes diferentes daqueles fornecidos.

**Não** use biblioteca alguma que implemente diretamente as funções pedidas. A única biblioteca sobre grafos permitida é aquela feita por mim e colocada no Teams.

Se desejar, pode usar qualquer função já implementada por você em trabalhos anteriores.

O trabalho deve ser entregue via Teams e somente o arquivo [grafosEspeciais2.jl](#) deve ser anexado.

## Funções

As funções a serem implementadas são:

1. **kneser n k** que devolve o grafo de Kneser  $K(n, k)$ . Veja o exercício E 1.16 do livro do Paulo Feofiloff.
2. **regular n r** que devolve um grafo  $r$ -regular com  $n$  vértices ou o erro **Grafo regular inexistente**, caso isso não seja possível.
3. **bipartidoCompleto p q**, que recebe dois naturais positivos e devolve um grafo bipartido completo  $K_{p,q}$  com  $p$  vértices  $1, \dots, p$  na primeira partição e  $q$  vértices  $p + 1, \dots, p + q$  na segunda partição.
4. **grafoAleatório n m** cria um grafo com  $n$  vértices  $1, \dots, n$  e **exatamente**  $m$  arestas. As extremidades das arestas devem ser escolhidas **aleatoriamente** e a função deve evitar criar grafos que não sejam simples, ou seja, não pode haver laços nem arestas paralelas. Se  $m$  ultrapassar o limite máximo de arestas de um grafo simples com  $n$  vértices, a função deve gerar o grafo completo.

5. **bipartidoAleatório**  $p$   $q$   $m$  cria um grafo bipartido com  $p$  vértices  $1, \dots, p$  na primeira partição,  $q$  vértices  $p + 1, \dots, p + q$  na segunda partição e **exatamente**  $m$  arestas. As extremidades das arestas devem ser escolhidas **aleatoriamente** e a função deve evitar criar grafos que não sejam simples, ou seja, não pode haver laços nem arestas paralelas. Se  $m$  ultrapassar o limite máximo de arestas de um grafo bipartido simples com  $p+q$  vértices, a função deve gerar o grafo bipartido completo.
6. **grafoArestas**  $g$  que recebe um grafo  $g$  e devolve o grafo de arestas de  $g$ . Veja o exercício E 1.24 do livro do Paulo Feofiloff.