

# **Grafos e Algoritmos Computacionais**

## **Professores André Britto e Renê de Gusmão**

### **Implementação 3**

### **Algoritmos dijkstra, bellman-ford, floyd-warshall, prim e kruskal**

#### **1 - Leitura e pesquisa**

A implementação 3 consiste no estudo de Algoritmos para o Caminho mais Curto, menor caminho entre todos os pares de nós e Árvore Geradora Mínima

No livro texto, Grafos: conceitos, algoritmos e aplicações. Goldbarg, E. e Goldbarg M. Elsevier, 2012, é possível encontrar o assunto nas seguintes seções:

Algoritmos para o Caminho mais Curto - Seção 4.3

- Algoritmo de Dijkstra
- Algoritmo de Ford-Moore-Bellman

Menor caminho entre todos os pares de nós - Seção 4.3

- Algoritmo de Floyd-Warshall

Árvore Geradora Mínima - Seção 3.3

- Algoritmo de Prim
- Algoritmo de Kruskal

O aluno também pode fazer buscas por material buscando pelos termos:

Algoritmo de Dijkstra  
Algoritmo de Ford-Moore-Bellman  
Algoritmo de Floyd-Warshall  
Algoritmo de Prim  
Algoritmo de Kruskal

Caminho mais curto (shortest path)

Árvores Geradoras Mínimas (minimum spanning tree (MST))

A leitura do material no livro é essencial, fiquem livres para complementar o estudo através de buscas em fontes de maior interesse. Pode ser material de outras disciplinas, livros disponíveis de forma gratuita, vídeos no youtube. O importante é que a busca e leitura seja bem completa.

O aluno também pode acompanhar as explicações dos algoritmos nas notas de aula enviadas para o SIGAA.

## **2 - Material complementar**

Slides de aula anexos ao tópico de aula no SIGAA:

Aula 05 - Dijkstra

Aula 06 - Bellman-ford

Aula 07 - Floyd warshal

Aula 08 - Prim e kruskal

Vídeo aulas:

Playlists:

Aula 05 - Caminhos mínimos (Algoritmo de Dijkstra, Algoritmo de Ford-Moore-Bellman, Algoritmo de Floyd-Warshall)

Aula 06 - Árvores Geradoras de Custo Mínimo (Prim e kruskall)

## **3 - Implementação**

Após leitura, pesquisa e audição do material, o aluno deverá resolver um conjunto de questões envolvendo os algoritmos estudados. As questões serão disponibilizadas através de juízes on-line. Serão utilizados dois sistemas diferentes:

<https://www.beecrowd.com.br/> (antigo URI Online Judge)

<https://vjudge.net/>

Caso não tenha conta, acesse os links e crie uma conta em cada um dos sites acima.

Esses juízes on-line agregam diferentes questões de programação e disponibilizam ambiente para criação de contests e julgamento das questões. Assim, as questões serão submetidas e corrigidas pelos juízes on-line.

Será criada uma tarefa no SIGAA referente à implementação 3 . Todos os código-fonte das questões submetidas no torneio deverão ser enviados na tarefa do SIGAA num arquivo .zip. O aluno deverá submeter tanto as questões corretas, quanto as questões que foram tentadas mas que não foram aceitas pelo juiz on-line. Os professores irão fazer a análise das respostas enviadas pelos alunos, que poderão ser chamados para uma arguição. Questões que foram aceitas pelo juiz online e que foram corretamente defendidas pelo aluno na arguição receberão a nota completa do quesito.

O nome dos arquivos enviados deve conter somente a informação do juiz automático e o número do problema. Por exemplo, ao enviar a solução para o problema 1266 - Tornado, caso a solução seja feita em c++ o arquivo enviado deve ser uri1266.cpp. No caso do Vjudge podem nomear as questões de acordo com identificação do context (vjA.cpp, vjB.cpp, e assim por diante) Isso deve ser feito para todas as linguagens, mesmo em java onde é necessário que o nome do arquivo seja o mesmo nome da classe. Nesse caso, não precisa alterar o nome da classe, basta apenas trocar o nome do arquivo .java e deixar o resto como foi originalmente feito.

Todas as questões devem ser compactadas em um arquivo .zip, nomeado somente com a sua matrícula.

Vocês podem enviar questões que não foram aceitas pelo juiz automático. Nesse caso, mande também um documento em .pdf indicando quais questões estão incompletas, o erro recebido pelo juiz automático e o que ficou faltando fazer para terminar a questão. A pontuação de cada quesito será definida pelo professor no momento da correção.

### **Informações dos contest**

Resolução de exercícios no site <https://www.beecrowd.com.br/> :

- 1 - Acesse o site <https://www.beecrowd.com.br/> e criem uma conta
- 2 - Após logar no sistema, acesse a opção contest e em seguida times
- 3 - Crie um time com somente uma pessoa, que é justamente você (Coloque o seu nome completo como nome do time)
- 4 - Acesse a opção contest e em seguida torneio
- 5 - Busquem pelo torneio 5637 GAC - Implementação 3 - p1 e em seguida se registre nele (senha: gacufs) -

6 - Acesse as questões do torneio e as resolva. A resolução é feita direto no próprio site, que julga se está certo ou errado. Porém, lembre de salvar as respostas num arquivo no seu computador e enviá-las junto às demais questões.

7 - O beecrowd permite que o torneio fique aberto por somente 5 dias, assim serão criados torneios clone (p2, p3, etc) para que os alunos consigam fazer as questões durante todo o período de resolução. Após a finalização do torneio, os alunos precisarão se inscrever no novo torneio, mas não precisarão submeter novamente as questões avaliadas corretamente no torneio anterior

Resolução de exercícios no site Virtual Judge (<https://vjudge.net/contest/489235>)

ID do torneio: 489235

Nome do torneio: GAC - Implementação 3

Senha para acesso: gacufs