

TRABALHO DE ESTRUTURAS DE DADOS II PARA A TERCEIRA AVALIAÇÃO

1) Um problema muito conhecido na ciências da computação é o da Torre de Hanói. O problema clássico da Torre de Hanói em, dados n discos e 3 anos, mover com o menor número possível de movimentos todos os n discos de um suporte origem, de acordo com as seguinte condições: apenas um disco pode ser movimentado de cada vez; os discos movidos deverão ser apoiados sempre em discos de maior tamanho ou na base de algum pino. A Figura 1 exibe os movimentos do desafio da Torre de Hanói. Por convenção uma configuração de discos será representada por um vetor com tantas posições quantas forem os discos. Na posição do disco será marcado o pino onde o disco está assentado, como mostrado na Figuras 2, 3 e 4. A Figura 5 exemplifica como formar um grafo que representa as possibilidade de movimentos dos discos. Cada vértice representa uma diferente configuração do desafio. As configurações são ligadas por arestas se uma configuração pode ser alcançada a partir de outra pelo movimento legal de um disco. De acordo com o exposto, modele o grafo que represente o grafo de movimentos do desafio da Torre de Hanói para o caso de 4 discos. Em seguida crie o grafo usando matriz de adjacência. Depois dada uma determinada configuração dos discos encontre o menor caminho para o resultado final usando o Algoritmo de Dijkstra, para isso coloque valor 1 em cada uma das arestas. Contabilize o tempo gasto para encontrar a solução. (2,5 pontos)

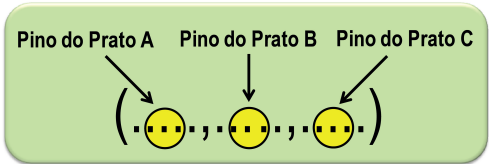


Figura 1

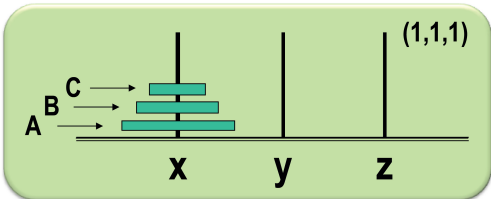


Figura 2

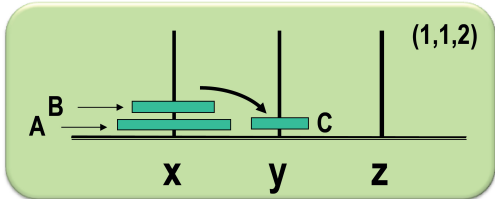


Figura 3

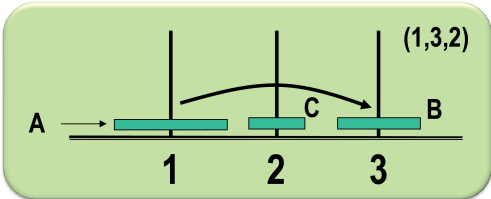


Figura 4

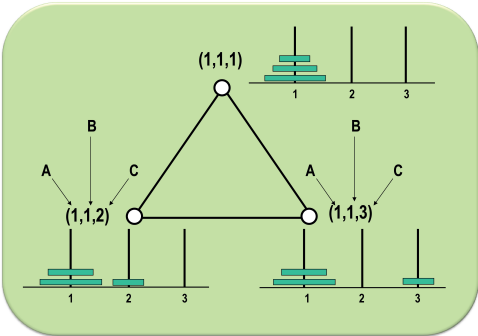


Figura 5

- 2) Refaça o exercício 1 mas agora use lista de adjacência, depois contabilize o tempo gasto para encontrar a solução e compare com a solução do exercício 1. (2,5 pontos)
- 3) Escreva um programa de planilha de cálculo rudimentar, utilize grafo para representar a planilha e implemente utilizando matriz de adjacências. Considere que a planilha possua colunas de A até H e linhas de 1 a 20. O usuário deve entrar com a célula e o valor que deseja colocar na célula, exemplo: A5 4, o usuário deseja colocar o valor 4 na coluna A linha 5. No entanto, o valor que o usuário deseja colocar em uma célula pode ser tanto um número, uma célula precedida por um sinal de igual, por exemplo =B13, ou ainda uma função (soma, max, min, média), a qual deve ser precedida por @ e seguida pelas células (inicial..final) entre parênteses para aplicar a operação correspondente a função, por exemplo: @soma(B1..D2). Modifique o grafo para refletir a situação atual da planilha e exiba a planilha na tela.

Exemplo:
A1 10
B1 20
A2 30
D1 +A1
C1 @soma(A1..B2)
D1 +C1

	A	B	C	D	E	F	G	H
1	10	20	60	60				
2	30							
3								

- 4) Refaça o exercício anterior agora usando listas de adjacências.
- 5) Para a atualização dos dados de uma determinada celular se faz necessário fazer uma busca. Faça uma busca em largura e uma busca em profundidade.
- 6) Calcule o tempo médio de execução dos exercícios anteriores em relação:
- (a) A inserção dos dados no grafo
 - (b) Busca em largura
 - (c) Busca em Profundidade

Depois faça uma comparação e análise dos resultados. A análise deve ser feita considerando o tempo médio calculado, bem como a quantidade de dados.

obs.1: Lembre-se de repetir cada cado 30 vezes para fazer o tempo médio.

- 7) Suponha uma base de dados de 4000 alunos e que ao invés de ordenar os dados o sistema cria uma tabela hashing para localizar os dados de um aluno pelo número de matrícula do aluno. Os dados dos alunos são: Matrícula, Nome, Curso, Período e Ano de ingresso. A matrícula é uma string de 11 dígitos, onde os quatros primeiros significa o ano de ingresso na Universidade, o quinto o número do curso, e do sexto até o décimo primeiro o número específico do aluno, ou seja, o número do aluno corresponde os seis últimos dígitos. Faça um programa para cada um dos itens a seguir que através de uma função hashing e de uma função para colisão organize a base de dados. Depois compare os resultados através do desempenho de cada uma das soluções. E também identifique qual das soluções produz um maior número de colisões. Faça o que se pede considerando um vetor destino de 1211 posições e depois com 1280 posições.
- (a) Função Hashing: rotação dos 5 primeiros dígitos para a direita, depois extrai o 2º, 4º e 5º dígitos da matricula rotacionada e obtenha o resto da divisão pelo tamanho do vetor destino. As colisões devem ser tratadas somando ao resto da divisão o primeiro dígito da matrícula depois de rotacionada. (1,25 pontos)
 - (b) Função Hashing: fold shift com 3 dígitos da seguinte forma: o 6º, 7º e 11º; 8º, 9º e 10º, depois obtenha o resto da divisão do resultado pelo tamanho do vetor destino. As colisões devem ser realizadas somando ao valor obtido o número formado pelo sexto e décimo primeiro dígito da matrícula. (1,25 pontos)

obs.: Em todos os casos se uma matrícula não conseguir ser colocada porque todos os possíveis locais já estão ocupados, retire a informação da primeira posição encontrada e coloque a nova informação.

Equipe: os programas podem ser feitos em dupla, mas os relatórios são individuais. Se os programas forem feitos em dupla, a dupla deve ser identificada no envio do código.

Data de Entrega: data terceira prova escrita

Entregar: Código Fonte, Relatório(Conforme Modelo em PDF)

Forma de Entrega: pelo SIGAA, caso tenha algum problema enviar por e-mail(julianaoc@gmail.com).

Entrevista Individual: agendar horário com a Professora.