

Projetos de Algoritmos - SCC5900

Lista Exercícios para a primeira prova

Força Bruta

- 1) Dê um exemplo de um problema que não pode ser resolvido pelo paradigma de força-bruta.
- 2) Projete um algoritmo para computar, por força-bruta, o valor do polinômio:

$$p(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0 \quad \text{para um certo valor } x$$

Determine a complexidade para o pior caso...

Se o seu algoritmo for $\Theta(n^2)$, então projete um que tenha ordem linear

- 3) Ordene a lista E,X,A,M,P,L,E em ordem alfabética por meio de seleção direta. Idem para bubblesort.
- 4) Seleção direta é um método de ordenação estável? O Bubblesort é estável?
- 5) Seja uma busca sequencial com sentinela. Encontre o nro de comparações que o algoritmo faz: a) no pior caso; no caso médio (com probabil. de sucesso $0 \leq p \leq 1$)
- 6) Determine o nro de comparações de caracteres pelo método força-bruta ao buscar o padrão ("GANDHI") em "THERE_IS_MORE_TO_LIFE_THAN_INCREASING_ITS_SPEED". Assuma que o comprimento da string (47) é sabido antes da busca começar.
- 7) No problema de procura em strings, há alguma vantagem em se comparar os padrões da direita-para-esquerda, ao invés de esquerda-para-direita?
- 8) Você consegue implementar um algoritmo mais rápido que o baseado em força-bruta para resolver o problema de encontrar, para n pontos, x_1, \dots, x_n o par mais próximo, na linha?
- 9) Que modificações precisam ser feitas no algoritmo de força-bruta de fecho convexo, para lidar com mais de dois pontos na mesma linha reta?
- 10) Esboce o algoritmo de força-bruta (busca exaustiva) para o problema de circuitos Hamiltonianos.

Dividir e Conquistar (DC)

- 11) Escreva um algoritmo DC para encontrar a posição do maior elemento em um vetor de n valores.
 1. Qual seria a saída do seu algoritmo se o vetor tiver vários elementos maiores
 2. monte e resolva a regra de recorrência para o nro de comparações de chaves
 3. como seu algoritmo se compara com o de força-bruta
- 12) Escreva um algoritmo DC para encontrar a posição do maior e do menor elemento em um vetor de n valores.
- 13) Encontre a ordem de crescimento para problemas com recorrências do tipo:
 1. $T(n) = 4T(n/2) + n$, $T(1) = 1$
 2. $T(n) = 4T(n/2) + n^2$, $T(1) = 1$
 3. $T(n) = 4T(n/2) + n^3$, $T(1) = 1$
- 14) Execute o mergesort para a sequência: E,X,A,M,P,L,E. Ele é estável?
- 15) Execute o quicksort para a sequência: E,X,A,M,P,L,E. Ele é estável?
- 16) Escreva uma versão recursiva para a busca binária.
- 17) Escreva um algoritmo DC para contar os nros de níveis em uma árvore binária (o alg. deve retornar 0 e 1 para as árvores vazia e com apenas a raiz, respectivamente). Qual a eficiência de seu algoritmo?

- 18) Compute 2101×1130 aplicando o algoritmo DC visto em sala de aula.
- 19) Implemente a versão DC para encontrar o par de pontos mais próximos
- 20) Qual a eficiência do algoritmo quickhull (por DC).

Reduzir e Conquistar (RC)

- 21) Ordene a lista E,X,A,M,P,L,E em ordem alfabética por meio de inserção. Ele é estável?
- 22) Existe um método (shellsort) que incrementa o algoritmo de inserção. Explique como ele funciona e que melhoria o algoritmo traz com relação ao de inserção
- 23) Seja o Grafo

(f)---(b) (c)----(g)
 \ / \ / /
 (d)-(a)------(e)

 1. escreva a matriz de adjacência e a lista de adjacência para ele.
 2. Iniciando pelo vértice (a), percorra o grafo por DFS(profundidade) e por BFS(largura). Em caso de "empate", aplique ordem alfabética.
- 24) Como é possível identificar componentes conexas em um grafo? Tanto por DFS como por BSF.
- 25) É realístico implementar um algoritmo que requeira a geração de todas as permutações de um conjunto de 25 elementos no computador? E todos os subconjuntos deste conjunto?
- 26) Gere todas as permutações de $\{1,2,3,4\}$ pelo:
 - algoritmo de mínima mudança bottom-up
 - algoritmo Johnson-Trotter
 - algoritmo ordem lexicográfica
- 27) Escreva o pseudo código de um algoritmo recursivo para gerar todas as 2^n bit strings de comprimento n.
- 28) Escreva um pseudo código para o algoritmo de multiplicação "a la russa".
- 29) No problema de Josephus, analise os padrões de saída para vários valores de n e
- 30) Escreva o algoritmo de Josephus.

Transformar e Conquistar (TC)

- 31) Considere problema de encontrar a distância entre os dois mais próximos números em vetor de n valores (a distância entre x e y quaisquer é calculada assim: $|x-y|$)
 - implemente um algoritmo baseado na pré-ordenação para resolver o problema e estime sua eficiência.
 - compare com o algoritmo de força-bruta
- 32) Considere o problema de encontrar o menor a maior valor em um vetor de n elementos
 - projeto um alg. baseado em pré-ordenação para o problema e determine sua eficiência
 - compare-o com: a) força-bruta; b_ dividir e conquistar;
- 33) Implemente o algoritmo para a eliminação de Gauss e dê a sua eficiência.
- 34) A eliminação por Gauss-Jordan difere da de Eliminação de Gauss em que os elementos acima da diagonal principal da matriz de coeficientes são substituídos por zeros ao mesmo tempo e também por usar uma mesma linha-pivô que os elementos abaixo da diagonal principal
 1. aplique o problema de Gauss jordan ao sistema de equações que vimos em sala
 2. Qual é a estratégia que este algoritmo usa?
- 35) Para cada sequencia abaixo, construa um árvore AVL:
 1. 1,2,3,4,5,6
 2. 6,5,4,3,2,1
 3. 3,6,5,1,2,4

- 36) Construa um heap para a lista 1,8,6,5,3,7,4
- 37) O Heapsort é um algoritmo estável?
- 38) Escreva o algoritmo de Horner para calcular o polinômio do exercício 2.
- 39) Baseado na regra de Horner, escreva um algoritmo para computar a exponenciação binária de a^n .