

FACULDADE CAMPO LIMPO PAULISTA
Bacharelado em Ciência da Computação

Projeto e Análise de Algoritmos I

Lista de Exercícios 2

Prof. Osvaldo.

1. Prove por indução que a soma das $n + 1$ primeiras potências inteiras de 2, isto é, $1 + 2 + 4 + \dots + 2^n$, é $S(n) = 2^{n+1} - 1$.
2. Prove por indução que a soma $1 + 3 + 5 + \dots + 2n - 1 = n^2$, i.e., a soma de números ímpares sempre é um quadrado perfeito.
3. Prove que se n é um número natural e $1 + x > 0$, então $(1 + x)^n \geq 1 + nx$.
4. Prove que numa seqüência de números naturais onde $a_k = a_{k-1} + r$ sendo r uma constante, a soma dos n primeiros números desta seqüência é
$$S(n) = a_1 + a_2 + \dots + a_n = n(a_n + a_1)/2.$$
5. Prove que a quantidade de nós de uma árvore binária cheia é igual a $2^{h+1} - 1$, onde h é a altura da árvore. Uma árvore binária cheia é uma árvore para a qual, se v é um nó de alguma de suas subárvores vazias, então v se localiza no último nível.
6. Ache uma expressão para a seguinte soma e prove a sua afirmação.

$$\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \dots + \frac{1}{2^n}$$

7. Prove a seguinte inequação:

$$\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \dots + \frac{1}{2^n} < 1$$

para todo $n \geq 1$.

8. Utilizando indução, prove que o algoritmo a seguir calcula o fatorial de um número inteiro $n \geq 2$.

```
Algoritmo Fatorial ( $n$ )  
Entrada:  $n$ , inteiro,  $n \geq 2$ .  
Saída:  $n!$  .  
{  
     $i := 1$ ;  $f := 1$ ;  
  
    repita  
    {
```

```

        i := i + 1;
        f := f * i;
    }
    até i = n;

    retornar f;
}

```

9. Aplicando a técnica de *design* de algoritmos indutivos estudada, escreva algoritmos, calcule as suas complexidades, para resolver os seguintes problemas:

- Calcular o menor elemento de um vetor de n elementos.
- Multiplicar dois números naturais x e y ($x \geq 0$ e $y \geq 0$).
- Somar os elementos de um vetor A de n elementos ($n \geq 1$).
- Inverter os elementos de um vetor A de n elementos ($n \geq 1$), ou seja, se $A = [1\ 3\ 5\ -2\ 0\ 4]$ então, após invertido, $A = [4\ 0\ -2\ 5\ 3\ 1]$.
- Somar os elementos de uma matriz A de n linhas e m colunas ($n \geq 1, m \geq 1$).
- Calcular o n -ésimo termo da seqüência da Fibonacci (1, 1, 2, 3, 5, 8, 13, ...).

10. Desenvolva um algoritmo recursivo para obter o valor de um polinômio de grau n no ponto igual a x . Qual a complexidade do seu algoritmo? Você seria capaz de desenvolver um algoritmo com complexidade de tempo não superior a $O(n)$?

11. Desenvolva um algoritmo que calcule o maior e o menor elemento de um vetor de n elementos de tal modo que a quantidade de comparações do seu algoritmo não seja maior do que $3n/2$.

12. Desenvolva uma versão com redução do tipo “dividir para conquistar” para os seguintes algoritmos. Calcule as complexidades de pior caso de cada um deles e diga, em cada caso, se melhorou, piorou ou continuou igual às complexidades em relação àquelas outras de soluções diferentes desenvolvidas em sala de aula ou nesta lista de exercícios.

- Calcular a soma dos n primeiros naturais;
- Calcular o menor elemento de A , um vetor de n elementos.

13. Considere a equação de recorrência a seguir:

$$T(n) = T(n - 1) + 4$$

$$T(1) = 4$$

Demonstre, por indução, que $T(n) = 4n$.

14. Use a indução matemática para mostrar que a solução da relação de recorrência abaixo é $T(n) = n \log n$.

$$T(n) = 2 T(n/2) + n, \text{ para } n = 2^k, k > 1.$$

$$T(2) = 2$$

15. Escreva um algoritmo para resolver o problema clássico das Torres de Hanói. Qual a complexidade do seu algoritmo?