Universidade Federal do Maranhão
Departamento de Informática
Disciplina: ***Estrutura de Dados II 2015.2***
Prof.: ***João Dallyson Sousa de Almeida***

Lista de Exercícios

FONTES:
Algoritmos : teoria e prática 1 Thomas H. Cormen. 2ª Ed. 2002
(Brookshear, 2012)
Algoritmos Genéticos. 3ª Ed. Ricardo Liden
ZIVIANI, Nivio et al. **Projeto de Algoritmos: com Implementações em Pascal e C**. Thomson, 2004.

1) Além da velocidade, que outras medidas de eficiência poderiam ser usadas em uma configuração real?

2) Selecione uma estrutura de dados que você já tenha visto antes e discuta seus pontos fortes e suas limitações.

3) Em que aspectos os problemas do caminho mais curto e do caixeiro-viajante são semelhantes? Em que aspectos eles são diferentes?

4) Mostre um problema real no qual apenas a melhor solução servirá. Em seguida, apresente um problema em que baste uma solução que seja "aproximadamente" a melhor.

5) Vamos supor que estamos comparando implementações de ordenação por inserção e ordenação por intercalação na mesma máquina. Para entradas de tamanho n, a ordenação por inserção é executada em $8n^2$ etapas, enquanto a ordenação por intercalação é executada em 64nlgn etapas. Para que valores de n a ordenação por inserção supera a ordenação por intercalação?

6) Qual é o menor valor de n tal que um algoritmo cujo tempo de execução é $100n^2$? funciona mais rápido que um algoritmo cujo tempo de execução é 2" na mesma máquina?

7) Use o método mestre para mostrar que a solução para a recorrência de pesquisa binária $T(n)=T(n/2) + \theta(1)$ é $T(n)=\theta(lgn)$.

8) Suponha que um programa possa ser solucionado por uma algoritmo em $\theta(2^n)$. O que podemos concluir acerca da complexidade do problema?

9) Suponha que um programa possa ser solucionado por um algoritmo em $O(n^2)$, bem como por outro algoritmo em $\theta(2^n)$. Um algoritmo sempre superará o outro?

10) Se a complexidade do algoritmo X for maior que a do algoritmo Y, o algoritmo X é necessariamente mais difícil de entender que o algoritmo Y? Explique sua resposta.

11) Indique se as afirmativas a seguir são verdadeiras e justifique sua resposta:

a) $2^{n+1} = O(2^n)$

b) $2^{2n} = O(2^n)$

c) $f(n) = O(u(n))$ e $g(n) = O(v(n)) \rightarrow \quad f(n) + g(n) = O(u(n) + v(n))$

d) $f(n) = O(u(n))$ e $g(n) = O(v(n)) \rightarrow \quad f(n) - g(n) = O(u(n) - v(n))$

12) Determine se n^2 é $O(n^3)$

13) Determine o tempo de execução dos seguintes algoritmos

```
a) Public class Exemplo_Aninhados {
      public static void main (String[] args)
         {
            int  n1=-1,n2,lim=Integer.parseInt(args[0]);
            while(++n1<lim)  {
               n2=0;
               while(n2<lim)  {
                  System.out.println(n1+  "  "+n2++);
               }
            }
            System.exit(0);
         }
   }
```

b)

```
function( int n )
{
     if (n == 1) return;
     for( i = 1 ; i <= n ; i + + )
     {
             for( j= 1 ; j <= n ; j + + )
             {
                     print("*" ) ;

                      break;
              }
      }
 }
```

14) Determine se as seguintes afirmações são verdadeiras:

a) 3^{n+1} é $O(2^n)$.

b) 2^{2n} é $O(4^n)$.

c) N^N é $O(2^N)$

d) NlogN é O (log N)

e) n^n é $O(n!)$.

15) Dê um exemplo de um algoritmo em cada uma das seguintes classes: $\theta(\log n)$, $\theta(n)$, $\theta(n^2)$.

16) Liste as classes $\theta(\log n!)$, $\theta(4^n)$, $\theta(1)$, $\theta(n!)$, $\theta(n^3)$ e $\theta(n\log n)$ em ordem crescente de eficiência.

17) Use o teorema mestre para derivar um limite assintótico Θ para as seguintes recorrências:

a) $T(n) = 2T(n/2) + n – 1$

b) $T(n) = 3T(n/2) + n$

c) $T(n) = 4T(n/2) + n^2$

d) $T(n) = 4T(n/2) + n^3$

e) $T(n) = 3T(n/2) + n^2$

f) $T(n) = T(n/2) + 2^n$

g) $T(n) = 16T(n/4) + n$

h) $T(n) = 2T(n/2) + n\log n$

18) Qual a complexidade do algoritmo a seguir? Justifique sua reposta:

```
int test(int n)
{
  int count = 0;
  for (int i = n; i > 0; i /= 2)
     for (int j = 0; j < i; j++)
        count += 1;
  return count;
}
```

19) Encontre o limite superior paras as seguintes funções:

a) $f(n) = 3n + 8$
b) $f(n) = n^2 + 1$
c) $f(n) = n^4 + 100n^2 + 50$
d) $f(n) = 2n^3 - 2n^2$
e) $f(n) = n$
f) $f(n) = 410$

20) Para cada par de funções f(n) e g(n) a seguir, f(n) = O(g(n)) ou g(n) = O(f(n)), mas não ambos. Determine quais são os casos.

a. $f(n) = (n^2 - n)/2, \; g(n) = 6n$

b. $f(n) = n + 2\sqrt{n}, \; g(n) = n^2$

c. $f(n) = n + n\log n, \; g(n) = n\sqrt{n}$

d. $f(n) = n^2 + 3n + 4, \; g(n) = n^3$

e. $f(n) = n\log n, \; g(n) = n\sqrt{n}/2$

f. $f(n) = n + \log n, \; g(n) = \sqrt{n}$

g. $f(n) = 2(\log n)^2, \; g(n) = \log n + 1$

h. $f(n) = 4n\log n + n, \; g(n) = (n^2 - n)/2$