MC-448 Projeto e Análise de Algoritmos I
Segundo semestre de 2006
Primeira Lista Complementar de Exercícios
Prof. Zanoni Dias

1. Dados os algoritmos 1 e 2 para o cálculo do produto de dois números naturais m e n, calcule a complexidade do pior caso. Use para o cálculo da complexidade o número de repetições do loop.

```
Algoritmo 1.
  begin
    Z:= 0; u:= n; v:= m;
    while u <> 0 do
      z:= z + v;
      u:= u - 1;
    end;
    write(z);
  end;
```

```
Algoritmo 2.
  begin
    z:= 0; u:= n; v:= m;
     while u <> 0 do
      begin
        if odd(u) then z:= z + v;
        u:= u div 2;
        v:= 2*v;
      end;
    write(z);
  end;
```

2. Calcule o número de somas que o algoritmo abaixo efetua:

```
 for i:= 1 to n do
   if odd(n) then
     begin
for j:= i to n do
  x := x + 1;
for j:= 1 to i do
  y:= y + 1;
     end;
```

3. Calcule em função de n, o número de somas e de multiplicações que o algoritmo abaixo efetua

```
for i:= 1 to n/2 do
  begin
    a:= b[i] + c[i];
    for j:= 2*i to n do
     d[j] := a*d[j];
  end;
```

4. Calcule, em função de n o número de multiplicações que o algoritmo abaixo executa:

```
  for i:= 1 to n-1 do
    for j:= i+1 to n do
      for k:= 1 to j do
a[i,k] := j*b[j,k]
```

5. Para cada uma das funções abaixo

   - $f(n) = (\sqrt(n) + \pi)^3 - (\sqrt(n) - \pi)^3$
   - $f(n) = \left(\begin{array}{c} n+1 \\ 3 \end{array} \right)$
   - $f(n) = \sum_{k=1}^{n} \frac{1}{k}$

   indique a ordem de crescimento assintótico conforme as seguintes opções.

   (a) $\theta(n^3)$
   (b) $\theta(1)$
   (c) $\theta(2^n)$
   (d) $\theta(n)$
   (e) $\theta(n \log(n))$
   (f) $\theta(n^2)$
   (g) $\theta(\log(n))$
   (h) $\theta(\log(\log(n))$
   (i) $\theta(n^{\frac{1}{2}})$

6. Rotule as funções $f(n)$ das opções (a) até (i) do exercício anterior, com os números de 1 a 9 de forma que a numeração reflita a ordem de crescimento assintótico das funções.

7. Seja $p(n) = \sum_{i=0}^{d} a_i n^i$, onde $a_d > 0$, um polinômio de grau d em n, e seja k uma constante. Use as definições de notação assintótica para provar as seguintes propriedades.

   - Se $k \geq d$, então $p(n) = \mathcal{O}(n^k)$.
   - Se $k \leq d$, então $p(n) = \Omega(n^k)$.
   - Se $k = d$, então $p(n) = \Theta(n^k)$.

8. Verifique se as afirmações abaixo são falsas ou verdadeiras. Justifique a sua resposta.

   - $2^{n+1} = \mathcal{O}(2^n)$.
   - $2^{2n} = \mathcal{O}(2^n)$.
   - Se $f(n)$ é $\mathcal{O}(g(n))$, então $g(n)$ é $\mathcal{O}(f(n))$.
   - $f(n)$ é $\mathcal{O}(f(n))^2$.

9. Mostre que $\log(n!) = \Theta(n \log(n))$.

10. Um algoritmo age sobre dados que dependem de um parâmetro n. Explique o significado das seguintes expressões:

    - "O consumo de tempo do algoritmo é $\mathcal{O}(n^2 \log n)$."
    - "O consumo de tempo do algoritmo é $\Omega(n^2)$."

11. São dadas as funções $f(n) = n$, $g(n) = nlog(n)$ e $h(n) = n^2$. Considere, agora, a função abaixo:

    $$p(n) = \begin{cases} n & \text{se } n \text{ é par} \\ n^2 & \text{se } n \text{ é ímpar} \end{cases}$$

    Relacione cada uma das funçòes f, g, h à função p usando O, Ω ou Θ, ou indique que tal relação não é possível.

12. Dado um vetor de n elementos, qual é o número médio (número esperado) de elementos que devem ser movidos para que se faça uma inserção, considerando igualmente prováveis as $n+1$ possíveis posições de inserção. Justifique.

13. Considere o seguinte algoritmo para resolver o problema da *Torre de Hanoi*.

```
Procedure Hanoi(n, ori, des, aux)

 inicio
   se n = 1 entao ‘move pino’ de ori para des
     senao
       inicio
         Hanoi(n-1, ori, aux, des)
         ‘move pino’ de ori para des
         Hanoi(n-1, aux, des, ori)
       fim
 fim
```

A operação básica é mover pinos.

a. Determine a relação de recorrência para T(n), inclusive T(1).

b. Calcule T(n).