Universidade Federal do Pará
Instituto de Ciências Exatas e Naturais
Faculdade de Computação
Análise de Algoritmos

Exercícios
Recorrências e Algoritmos Recursivos

1. [POSCOMP 2009] A sequência de Fibonacci é uma sequência de inteiros, cujo primeiro termo é 0, o segundo termo é 1, e a partir do terceiro, cada termo é igual à soma dos dois anteriores. O seguinte algoritmo recursivo retorna o n-ésimo termo da sequência.

```
Procedimento F(n)
       se n < 3 então retornar n-1
       senão retornar F(n-1) + F(n-2)
```

A chamada externa é F(n), sendo n > 0.

Assinale a alternativa CORRETA:

(A) O algoritmo é incorreto, pois não retorna o n-ésimo termo da sequência.
(B) O algoritmo é ótimo, no que diz respeito ao número de passos.
(C) O número de passos efetuados pelo algoritmo é linear em n.
(D) O número de passos efetuados pelo algoritmo é polinomial em n.
(E) O número de passos efetuados pelo algoritmo é exponencial em n.

2. [POSCOMP 2016] O tempo de execução $T(n)$ de um algoritmo, em que n é o tamanho da entrada, é dado pela equação de recorrência $T(n) = 8T(n/2) + q \cdot n$ se $n > 1$. Dado que $T(1) = p$, e que p e q são constantes arbitrárias, a complexidade do algoritmo é

(A) $O(n)$.
(B) $O(n\ log\ n)$.
(C) $O(n^2)$.
(D) $O(n^3)$.
(E) $O(n^n)$.

3. Considere o algoritmo abaixo.

```
 PROC(n)
se n <= 1 então
   retornar (2)
senão
   retornar (PROC(n/2) + PROC(n/2))
fim se
```

Assinale a alternativa que indica o valor retornado pelo algoritmo considerando a entrada $n = 64$.

(A) 128
(B) 130
(C) 1.024
(D) 4.096
(E) 4.160

4. Marque a alternativa que corretamente apresenta uma definição recursiva T para a multiplicação de dois números inteiros positivos m e n.

(A) $T(1) = 1$ e $T(n) = T(n-1) + m$ para $n \geq 2$.
(B) $T(1) = m$ e $T(n) = T(n-1) + m$ para $n \geq 2$.
(C) $T(1) = m$ e $T(n) = T(n-1)\ m$ para $n \geq 2$.
(D) $T(1) = 1$ e $T(n) = T(n-1)\ T(n-2)$ para $n \geq 2$.
(E) $T(1) = m$ e $T(n) = T(n-1)\ T(n-2)$ para $n \geq 2$.

5. As definições recursivas apresentadas abaixo descrevem o tempo de execução de dois algoritmos recursivos: A e B:

$T_A(1) = 1$ e $T_A(n) = 2T_A(n-1) + 2$ para $n \geq 2$.

$T_B(1) = 1$ e $T_B(n) = T_B(n/2) + n$ para $n \geq 2$.

Assinale a alternativa correta.

(A) Os algoritmos não são eficientes.
(B) Os algoritmos são assintoticamente equivalentes.
(C) O algoritmo B tem complexidade exponencial no tempo.
(D) O algoritmo A é mais eficiente assintoticamente que o algoritmo B.
(E) O algoritmo B é mais eficiente assintoticamente que o algoritmo A.

6. Considere o algoritmo A abaixo.

```
Algoritmo A(n)
Entrada: n, inteiro, n > 0.
{
   se n = 1
       retornar (1);
   senão
       retornar (2 * A(n/2) + 1);
}
```

A complexidade no tempo de pior caso do algoritmo A é

(A) linear.
(B) logarítmica.
(C) quadrática.
(D) exponencial.
(E) $nlog(n)$.

7. O algoritmo recursivo abaixo soma os n primeiros números naturais.

```
Algoritmo Soma(n)
Entrada: n, inteiro, n > 0.
{
   se n = 1
       retornar (1);
   senão
       retornar (Soma(n - 1) + n);
}
```

A complexidade no tempo do algoritmo é

(A) linear.
(B) logarítmica.
(C) quadrática.
(D) exponencial.
(E) $nlog(n)$.

8. Resolva as relações de recorrência abaixo pelo Teorema Mestre.

(a) $T(1) = 1$.
$T(n) = 4T(n/4) + n$ para $n > 1$.

(b) $T(1) = 1$.
$T(n) = 7T(n/2) + n^2$ para $n > 1$.

(c) $T(1) = 1$.
$T(n) = 3T(n/9) + 2n$ para $n > 1$.

9. [POSCOMP 2018] Dadas as seguintes relações de recorrência:

I. $T(n) = 2T(n/2) + O(n)$

II. $T(n) = 8T(n/2) + O(n^2)$

III. $T(n) = T(n/2) + O(1)$

As relações de recorrência I, II, e III pertencem, nessa ordem, às classes de complexidade:

(A) $\Theta(n^2)$, $\Theta(n^3)$ e $\Theta(n)$
(B) $\Theta(n)$, $\Theta(n^2)$ e $\Theta(n^3)$
(C) $\Theta(n\ log\ n)$, $\Theta(n^3)$ e $\Theta(log\ n)$
(D) $\Theta(log\ n)$, $\Theta(n\ log\ n)$ e $\Theta(n^3)$
(E) $\Theta(n^2)$, $\Theta(n^2)$ e $\Theta(n^2)$

10. [POSCOMP 2004] Para um certo problema foram apresentados dois algoritmos de divisão e conquista, A e B, cujos tempos de execução são descritos, respectivamente, por $T_A(n) = 7T_A(n/2)+n^3$ e $T_B(n) = \lambda T_B(n/4)+n^2$. Qual é o maior valor inteiro para λ, tal que o tempo de execução de B seja assintoticamente menor que o de A, isto é, $T_B(n) \in o(T_A(n))$?

(A) 16
(B) 49
(C) 63
(D) 64
(E) 65