Faculdade de Tecnologia de São Paulo
Departamento de Tecnologia da Informação
Prof. Dr. Silvio do Lago Pereira
Aux. Doc. Lucio Nunes de Lira

1. [0,5] Analise as afirmações a seguir e indique a alternativa correta.

I. *Asserção é uma condição que deve ser satisfeita num estado de execução de um algoritmo.*
II. *Precondições, poscondições e invariantes são asserções usadas em provas de correção empírica.*
III. *Uma invariante é um dos elementos essenciais na especificação de um algoritmo.*

(***a***) Apenas as afirmações I e II são verdadeiras.
(***b***) Apenas as afirmações I e III são verdadeiras.
(***c***) Apenas as afirmações II e III são verdadeiras.
(***d***) Nenhuma das anteriores.

2. [0,5] Analise as afirmações a seguir e indique a alternativa correta.

I. *Mesmo sem conhecer a especificação de um algoritmo, é possível analisar a sua correção.*
II. *Um algoritmo está correto se ele produz uma saída correta, para qualquer entrada correta.*
III. *A análise de correção de um algoritmo deve ser feita antes de sua análise de eficiência.*

(***a***) Apenas as afirmações I e II são verdadeiras.
(***b***) Apenas as afirmações I e III são verdadeiras.
(***c***) Apenas as afirmações II e III são verdadeiras.
(***d***) Nenhuma das anteriores.

3. [0,5] Analise as afirmações a seguir e indique a alternativa correta.

I. *Para provar a correção de um algoritmo recursivo, precisamos usar prova por indução finita.*
II. *Para provar a correção de um algoritmo não-recursivo, precisamos usar prova de invariante de laço.*
III. *Uma prova de invariante de laço consiste de três etapas: iniciação, manutenção e terminação.*

(***a***) Apenas as afirmações I e II são verdadeiras.
(***b***) Apenas as afirmações I e III são verdadeiras.
(***c***) Apenas as afirmações II e III são verdadeiras.
(***d***) Nenhuma das anteriores.

4. [0,5] Se escolhermos $n_0 = 1$, qual o menor c que podemos escolher para provar que $f(n) = n^2 + 3n$ é $\mathrm{O}(n^2)$?

(***a***) $c = 1.75$.
(***b***) $c = 2.00$.
(***c***) $c = 2.50$.
(***d***) Nenhuma das anteriores.

5. [0,5] Sendo $f(n) = \lg 2^{(\lg n - 3n)} + 3n - 1$, é correto afirmar que:

(***a***) $f(n)$ é $\mathrm{O}(\lg n)$.
(***b***) $f(n)$ é $\Omega(n^2)$.
(***c***) $f(n)$ é $\Theta(n \lg n)$.
(***d***) Nenhuma das anteriores.

6. [0,5] Sabendo-se que os algoritmos a seguir satisfazem à mesma especificação, para $n \in \mathbb{N}$, é correto afirmar que:

```
def A(n):
    if n==0: return 1
    return A(n-1)+A(n-1)

def B(n):
    if n==0: return 1
    return 2*B(n-1)
```

I. *Para algum* $n \in \mathbb{N}$, *temos que* $A(n) \neq B(n)$.
II. *Para* n *grande, o algoritmo* A *é preferível ao* B.
III. *Os algoritmos têm a mesma eficiência temporal.*

(***a***) Apenas as afirmações I e II são verdadeiras.
(***b***) Apenas as afirmações I e III são verdadeiras.
(***c***) Apenas as afirmações II e III são verdadeiras.
(***d***) Nenhuma das anteriores.

7. [1,5] Prove que o algoritmo a seguir satisfaz a especificação $\alpha \doteq (a > 0 \wedge b \geq 0)$ e $\delta \doteq (r = a^b)$, usando a invariante $\gamma \doteq (r = a^i)$.

```
def C(a,b):
    r = 1
    i = 0
    while i<b:
        i += 1
        r *= a
    return r
```

8. [1,5] Prove que o algoritmo a seguir satisfaz a especificação $\alpha \doteq (a>0 \wedge b \geq 0)$ e $\delta \doteq (_res = a^b)$.

```
def D(a,b):
    if b==0: return 1
    return a * D(a,b-1)
```

9. [2,0] Defina a função de tempo $\mathrm{T}(n)$ para o algoritmo a seguir e prove que ela é $\mathrm{O}(n^2)$.

```
def E(n):
    c = 0
    for i in range(0,2*n):
        for j in range(i,2*n):
            c += 1   # conte apenas essa soma!
    return c
```

10. [2,0] Defina a função de tempo $\mathrm{T}(n)$ para o algoritmo a seguir e, supondo que $n=2^k$ para $k \in \mathbb{N}^+$, prove que ela é $\mathrm{O}(\lg n)$.

```
def F(n):
    if n==2: return 0
    return 1 + F(n//2)   # conte apenas essa soma!
```