SSC0503 - Introdução à Ciência de Computação II
2ª Lista
Professor: Claudio Fabiano Motta Toledo (`claudio@icmc.usp.br`)
Estagiário PAE: Jesimar da Silva Arantes (`jesimar.arantes@usp.br`)

---

1. Prove por indução que as proposições são verdadeiras para todo inteiro positivo n.

    1. $2 + 6 + 10 + \cdots + (4n - 2) = 2n^2$
    2. $1 + 5 + 9 + \cdots + (4n - 3) = n(2n - 1)$
    3. $4 + 10 + 16 + \cdots + (6n - 2) = n(3n + 1)$
    4. $1^2 + 3^2 + \cdots + (2n - 1)^2 = \frac{n(2n-1)(2n+1)}{3}$
    5. $1 + a + a^2 + \cdots + a^{n-1} = \frac{a^n - 1}{a-1}$ para $a \neq 0$, $a \neq 1$
    6. $\frac{1}{1\cdot 2} + \frac{1}{2\cdot 3} + \frac{1}{3\cdot 4} + \cdots + \frac{1}{n(n+1)} = \frac{n}{n+1}$
    7. Uma *progressão geométrica* é uma sequência de termos com um termo inicial a tal que cada termo a seguir é obtido multiplicando-se o termo anterior por uma *mesma razão* r. Prove a fórmula para a soma dos n primeiros termos de uma progressão geométrica ($n \geq 1$):

        $a + ar + ar^2 + \cdots + ar^{n-1} = \frac{a - ar^n}{1-r}$
    8. Uma *progressão aritmética* é uma sequência de termos com um termo inicial a tal que cada termo a seguir é obtido somando-se uma *mesma parcela* d ao termo anterior. Prove a fórmula para a soma dos n primeiros termos de uma progressão aritmética ($n \geq 1$):

        $a + (a + d) + (a + 2d) + \cdots + [a + (n - 1)d] = \frac{n}{2}[2a + (n - 1)d]$
    9. Prove que $n^2 > n + 1$ para $n \geq 2$.
    10. Prove que $n^2 > 5n + 10$ para $n > 6$.
    11. Prove que $2^n < n!$ para $n \geq 4$.
    12. Prove que $n! < n^n$ para $n \geq 2$.
    13. (a) Tente usar indução para provar que
        $1 + \frac{1}{2} + \frac{1}{4} + \cdots + \frac{1}{2^n} < 2$ para $n \geq 1$
        O que não funciona?

        (b) Prove que
        $1 + \frac{1}{2} + \frac{1}{4} + \cdots + \frac{1}{2^n} = 2 - \frac{1}{2n}$ para $n \geq 1$
        mostrando, assim que
        $1 + \frac{1}{2} + \frac{1}{4} + \cdots + \frac{1}{2^n} < 2$ para $n \geq 1$
    14. $2^{3n} - 1$ é divisível por 7.
    15. $3^{2n} + 7$ é divisível por 8.

16. $2^n + (-1)^{n+1}$ é divisível por 3.
17. $2^{5n+1} + 5^{n+2}$ é divisível por 27.
18. $10^n + 3 \cdot 4^{n+2} + 5$ é divisível por 9.
19. $n^3 - n$ é divisível por 3.
20. Use indução para provar que o produto de três inteiros positivos consecutivos quaisquer é divisível por 3.
21. Uma tribo obscura tem uma linguagem com apenas três palavras básicas, *mono*, *nono* e *sono*. Novas palavras são compostas agregando-se estas palavras em qualquer ordem, como em *sonononomonononono*. Qualquer justaposição deste tipo é válida.
    (a) Use o princípio de indução (sobre o número de palavras básicas em cada palavra composta) para provar que qualquer palavra nessa linguagem tem um número par de letras iguais a *o*.
    (b) Use o segundo princípio de indução (sobre o número de palavras básicas em cada palavra composta) para provar que qualquer palavra nesta linguagem tem um número par de letras iguais a *o*.
22. Considere fbfs proposicionais contendo apenas os conectivos $\wedge$, $\vee$, $\rightarrow$ (sem a negação) e tais que, ao usar um conectivo para juntar duas fbfs, temos que usar parênteses. Conte cada letra de proposição, conectivo ou parênteses como um símbolo. Por exemplo, $((A) \wedge (B)) \vee ((C) \wedge (D))$ é uma fbf deste tipo, com 19 símbolos. Prove que qualquer fbf deste tipo tem um número ímpar de símbolos.
23. Prove que qualquer quantia em selos maior ou igual a 2 centavos pode ser obtida usando-se apenas selos de 2 e 3 centavos.
24. Prove que qualquer quantia em selos maior ou igual a 12 centavos pode ser obtida usando-se apenas selos de 4 e 5 centavos.
25. Em qualquer grupo de k pessoas, $k \geq 1$, cada pessoa cumprimenta, com aperto de mão, todas as outras pessoas. Encontre uma fórmula para o número de apertos de mão e prove-a usando indução.

2. Prove a corretude dos algoritmos:

    1. Quadrado: retorna o valor de x^2 para $x \geq 1$.
    2. Fatorial: retorna o valor de $x!$ para $x \geq 1$.
    3. Potência: retorna o valor de x^y para $x, y \geq 1$.
    4. Divide: calcula e escreva o quociente q e o resto r quando x é dividido por y, $x \geq 0$, $y \geq 1$.
    5. Diferença: que retorna o valor de x - y para $x, y \geq 0$.
    6. Cálculo: retorna o valor de $x * y^n$ para $n \geq 0$.
    7. PotênciasDe2: retorna o valor de 2^n para $n \geq 1$.

Algorithm 1: Quadrado (inteiro positivo x)

```
begin
    inteiros i, j
    i = 1
    j = 1
    while i ≠ x do
        j = j + 2i + 1
        i = i + 1
    //agora j tem o valor x^2
    return j
```

Algorithm 2: Fatorial (inteiro positivo x)

```
begin
    inteiros i, j
    i = 2
    j = 1
    while i ≠ x + 1 do
        j = j * i
        i = i + 1
    //agora j tem o valor x!
    return j
```

Algorithm 3: Potência (inteiro positivo x, inteiro positivo y)

```
begin
    inteiros i, j
    i = 1
    j = x
    while i ≠ y do
        j = j * x
        i = i + 1
    //agora j tem o valor x^y
    return j
```

Algorithm 4: Divide (inteiro não-negativo x, inteiro positivo y)

```
begin
    inteiros não negativos q, r
    q = 0
    r = x
    while r ≥ y do
        q = q + 1
        r = r - y
    //agora q e r são o quociente e o resto
    escreva(”O quociente é” q ”e o resto é” r)
```

Algorithm 5: Diferença (inteiro não-negativo x, inteiro não-negativo y)

```
begin
    inteiros i, j
    i = 0
    j = x
    while i ≠ y do
        j = j - 1
        i = i + 1
    //agora j tem o valor de x − y
    return j
```

Algorithm 6: Cálculo (inteiro x, inteiro y, inteiro não negativo n)

```
begin
    inteiros i, j
    i = 0
    j = x
    while i ≠ n do
        j = j * y
        i = i + 1
    //agora j tem o valor x ∗ y^n
    return j
```

Algorithm 7: PotênciasDe2 (inteiro positivo n)

```
begin
    inteiros i, j
    i = 1
    j = 2
    while i ≠ n do
        j = j * 2
        i = i + 1
    //agora j tem o valor 2^n
    return j
```