Algoritimos de programação II

Professor: Marco Aurélio Stefanes

Lista 01

Aluno:
Augusto Cesar de Aquino Ribas
Análise de Sistemas

1 Aula 01-03 : Exercícios. 1.5 a 1.9

1. (a) Escreva uma função recursiva com a seguinte interface:

```
int soma_digitos(int n)
```

   que receba um número inteiro positivo n e devolva a soma de seus dígitos.

   (b) Escreva um programa que receba um número inteiro n e imprima a soma de seus dígitos. Use a função do item (a).

2. A **seqüência de Fibonacci** é uma seqüência de números inteiros positivos dada pela seguinte fórmula:

$$\begin{cases} F_1 & = & 1 \\ F_2 & = & 1 \\ F_i & = & F_{i-1} + F_{i-2}, \text{ para } i \geq 3 \end{cases}$$

   (a) Escreva uma função recursiva com a seguinte interface:

```
int Fib(int i)
```

   que receba um número inteiro positivo i e devolva o i-ésimo termo da seqüência de Fibonacci, isto é, F_i.

   (b) Escreva um programa que receba um número inteiro $i \geq 1$ e imprima o termo F_i da seqüência de Fibonacci. Use a função do item (a).

3. O **piso** de um número inteiro positivo x é o único inteiro i tal que $i \leq x < i + 1$. O piso de x é denotado por $\lfloor x \rfloor$.

   Segue uma amostra de valores da função $\lfloor \log_2 n \rfloor$:

n	15	16	31	32	63	64	127	128	255	256	511	512
$\lfloor \log_2 n \rfloor$	3	4	4	5	5	6	6	7	7	8	8	9

   (a) Escreva uma função recursiva com a seguinte interface:

```
int piso_log2(int n)
```

   que receba um número inteiro positivo n e devolva $\lfloor \log_2 n \rfloor$.

   (b) Escreva um programa que receba um número inteiro $n \geq 1$ e imprima $\lfloor \log_2 n \rfloor$. Use a função do item (a).

4. Considere o seguinte processo para gerar uma seqüência de números. Comece com um inteiro n. Se n é par, divida por 2. Se n é ímpar, multiplique por 3 e some 1. Repita esse processo com o novo valor de n, terminando quando $n = 1$. Por exemplo, a seqüência de números a seguir é gerada para $n = 22$:

22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1

É conjecturado que esse processo termina com $n = 1$ para todo inteiro $n > 0$. Para uma entrada n, o **comprimento do ciclo de** n é o número de elementos gerados na seqüência. No exemplo acima, o comprimento do ciclo de 22 é 16.

(a) Escreva uma função não-recursiva com a seguinte interface:

```c
int ciclo(int n)
```

que receba um número inteiro positivo n, mostre a seqüência gerada pelo processo descrito acima na saída e devolva o comprimento do ciclo de n.

(b) Escreva uma versão recursiva da função do item (a) com a seguinte interface:

```c
int cicloR(int n)
```

que receba um número inteiro positivo n, mostre a seqüência gerada pelo processo descrito acima na saída e devolva o comprimento do ciclo de n.

(c) Escreva um programa que receba um número inteiro $n \geq 1$ e determine a seqüência gerada por esse processo e também o comprimento do ciclo de n. Use as funções em (a) e (b) para testar.

5. Podemos calcular a potência x^n de uma maneira mais eficiente. Observe primeiro que se n é uma potência de 2 então x^n pode ser computada usando seqüências de quadrados. Por exemplo, x^4 é o quadrado de x^2 e assim x^4 pode ser computado usando somente duas multiplicações ao invés de três. Esta técnica pode ser usada mesmo quando n não é uma potência de 2, usando a seguinte fórmula:

$$x^n = \begin{cases} 1\,, & \text{se } n = 0, \\ (x^{n/2})^2\,, & \text{se } n \text{ é par,} \\ x \cdot x^{n-1}\,, & \text{se } n \text{ é ímpar.} \end{cases}$$

(a) Escreva uma função com interface

```c
int potencia(int x, int n)
```

que receba dois números inteiros x e n e calcule e devolva x^n usando a fórmula acima.

(b) Escreva um programa que receba dois números inteiros a e b e imprima o valor de a^b.

2 Aula 05: Exercícios 2.6 a 2.10

texto