



Exercícios – Funções I

1. Implemente funções para calcular:

- A área de um retângulo $A_{ret} = b \times h$, onde b é a sua base e h é a sua altura
- A área de um triângulo $A_{tri} = (b \times h)/2$, onde b é a sua base e h é a sua altura
- A área de um círculo $A_{circ} = \pi \times r^2$, onde r é o raio

Dica: pense em termos de entrada e saída para descobrir os parâmetros de cada uma destas funções. Em seguida, implemente a função **main** de forma que o usuário entre com os dados necessários para calcular cada uma das áreas utilizando chamadas às funções implementadas.

2. Implemente uma função que receba um número inteiro e retorne o valor absoluto do número. Implemente também a função **main**, de modo que o usuário possa digitar um número inteiro qualquer e visualizar o valor absoluto do número digitado na tela utilizando a função implementada. Exemplos:

```
-- Exemplo 1:
Insira um numero inteiro:
-157
Modulo do numero:
157
-- Exemplo 2:
Insira um numero inteiro:
1021
Modulo do numero:
1021
```

3. Implemente uma função que receba três letras minúsculas como parâmetros. A sua função deve computar qual a maior letra, de acordo com a ordem alfabética. Implemente também a função **main**, de modo que o usuário possa digitar as letras e visualizar na tela qual a maior delas de acordo com a chamada à função.

```
-- Exemplo 1:
Insira tres letras:
p a k
Maior letra:
p
-- Exemplo 2:
Insira tres letras:
a b c
Maior letra:
c
```

4. Implemente uma função chamada `muda_tamanho`, que deve receber como parâmetro um caractere. Caso o caractere seja uma letra minúscula, a função deve retornar a sua versão em maiúscula. Caso o caractere seja uma letra maiúscula, a função deve retornar a sua versão em minúscula. Implemente a função `main` de forma que o usuário possa inserir uma letra e visualizar na tela o resultado computado pela função. Exemplos:

```
-- Exemplo 1:
Informe uma letra:
Q
Conversao:
q
-- Exemplo 2:
Informe uma letra:
e
Conversao:
E
```

5. Implemente uma função para cada uma das funções matemáticas a seguir. Implemente também uma função `main` que receba como entrada um número real `x` e um número inteiro `n` e imprima o resultado de cada uma das funções. **Não utilize a biblioteca de funções matemáticas.**

(a) x^n

(b) $x! = \prod_{i=1}^x i$

(c) $\sum_{i=1}^n 2x$

6. Implemente uma função chamada `eh_primo`, que retorna verdadeiro caso um número inteiro passado como parâmetro seja primo e falso caso contrário. Implemente também a função `main`, que deve ler do usuário vários números inteiros positivos até que o número inserido seja menor ou igual a 0. Quando isto acontecer, o programa deve informar quantos dos valores lidos foram primos. Exemplo:

```
Informe um numero positivo (negativo para encerrar):
3
4
2
13
-1
3 numeros primos foram digitados
```

7. Utilizando a mesma função `eh_primo` do exercício anterior, implemente uma função `main` que leia do usuário um inteiro positivo `n` e imprima na tela todos os números de 1 a `n`, com uma frase ao lado informando se cada um é primo. Exemplo:

```
Informe um numero positivo:
10
1 nao eh primo
2 eh primo
3 eh primo
4 nao eh primo
```

```

5 eh primo
6 nao eh primo
7 eh primo
8 nao eh primo
9 nao eh primo
10 nao eh primo

```

8. Implemente uma função chamada `conta_digitos`, que deve receber como parâmetro um número inteiro maior ou igual a zero e computar a quantidade de dígitos do número. Implemente também a função `main`, de modo que o usuário possa digitar um número inteiro maior ou igual a zero e, utilizando a função implementada, visualizar na tela quantidade de dígitos do número de acordo com o resultado da função. Exemplos:

```

-- Exemplo 1:
Informe um numero:
1461
0 numero tem 4 digitos
-- Exemplo 2:
Informe um numero:
0
0 numero tem 1 digitos

```

9. A sequência de Fibonacci é dada por

1, 1, 2, 3, 5, 8, 13, 21, ...,

isto é, ela é uma sequência infinita de números inteiros e positivos onde cada termo é formado pela soma dos dois anteriores, sendo os dois primeiros termos iguais a 1. Implemente uma função que compute o termo k da sequência, onde k é um parâmetro dado por um número inteiro maior ou igual a 0. Implemente também a função `main`, de modo que o usuário possa digitar um número inteiro n correspondente a quantidade de termos da sequência e visualizar na tela todos os n primeiros termos da sequência. Exemplos:

```

-- Exemplo 1:
Informe a quantidade de termos:
1
Sequencia de Fibonacci:
1
-- Exemplo 2:
Informe a quantidade de termos:
9
Sequencia de Fibonacci:
1 1 2 3 5 8 13 21 34

```