

Linguagem de Programação I

Lista de Exercícios VI

Profa: Susana Marrero Iglesias

1. Defina uma função hipotenusa que calcule o comprimento da hipotenusa de um triângulo retângulo, ao serem fornecidos os catetos. Use a função em um programa para determinar o comprimento da hipotenusa onde o comprimento dos catetos são lidos pelo programa. A função deve receber dois argumentos float e retornar um valor float.
2. Crie um programa que leia a *base* e o *expoente* e imprima $base^{expoente}$. Admita que o expoente é um valor inteiro. Escreva uma função potenciaInt, que utilize um for para controlar o cálculo. Não use a função power().
3. Escreva um programa que receba *n* duplas de números inteiros e determine se o segundo é múltiplo do primeiro. Escreva uma função multiplo que receba dois números inteiros e retorne 1 se o segundo for múltiplo do primeiro ou 0 em caso contrário.
4. Diz-se que um número é perfeito se a soma de seus fatores, incluindo 1 (mas não o número em si), resulta no próprio número. Por exemplo 6, é um número perfeito porque $6 = 1+2+3$. Escreva uma função perfeito que determine se o parâmetro numero é um numero perfeito. Use essa função em um programa que determine e imprima todos os números perfeitos entre 1 e 100. Imprima os fatores de cada número encontrado para confirmar que ele é realmente perfeito.
5. Escreva uma função que utilize um valor inteiro e retorna o numero com os dígitos invertidos. Por exemplo, dado o número 7631, a função deve retornar 1367.
6. Diz-se que um número inteiro é primo se for divisível apenas por 1 por se mesmo. Por exemplo, 2, 3, 5, e 7 são números primos, mas 4, 6, 8 e 9, não são.
 - a) Escreva uma função que determine se um número é primo.
 - b) Use essa função em um programa que determine e imprima todos os números primos entre 1 e 1000.
7. Escreva um programa que simule o lançamento de uma moeda. Para cada lançamento da moeda o programa deve imprimir Cara ou Coroa. Deixe o programa lançar a moeda 100 vezes e conte o número de vezes que cada lado da moeda aparece. Imprima os resultados. O programa deve chamar uma função separada jogada que não utiliza argumentos e retorna 0 para coroa e 1 para cara.
8. Os computadores estão desempenhando um papel cada vez maior na educação. Escreva um programa que ajudara a os alunos de primeiro grau a aprender a multiplicar. Use rand para produzir dois números positivos de um dígito. O programa deve imprimir uma pergunta do tipo: Quanto e 6 vezes 7?
O aluno deve digitar a resposta. Seu programa deve examinar a resposta do aluno. Se ela estiver correta o programa deve imprimir Muito Bem!!! e fazer outra pergunta de multiplicação. Se a resposta estiver errada, o programa deve imprimir Não. Tente novamente. e então deixar que o aluno fique tentando acertar a mesma pergunta até por fim conseguir.
9. Escreva um programa que faça o jogo de "adivinhar o número" da forma que segue: seu programa escolhe um numero para ser adivinhado selecionando um inteiro aleatório entre 1 e 100 e pede ao jogador para digitar um palpite, logo o programa mostra uma das respostas seguintes:

Excelente! Voce adivinhou o numero!

Muito baixo. Tente novamente.

Muito alto. Tente novamente.

Seu programa deve rodar até o jogador acertar o número.

10. Um dos jogos de azar mais populares é o jogo conhecido como \craps". Escreva um programa que simule o jogo de \craps". As regras do jogo são simples:

Um jogador joga dois dados. Cada dado tem seis faces. Essas faces contêm 1, 2, 3, 4, 5, ou 6 pontos. Depois dos dados pararem a soma das faces superiores do dado é calculada. Se a soma for 7 ou 11 no primeiro lançamento o jogador vence. Se a soma for 2, 3 ou 12 no primeiro lançamento (chamado \craps"), o jogador perde (i.e. a \casa" vence). Se a soma for 4, 5, 6, 8, 9 ou 10 no primeiro lançamento esta soma se torna o \ponto" do jogador.

Para vencer, o jogador deve continuar lançando dados até \fazer o ponto". O jogador perde se tirar um 7 antes de fazer o ponto.

Seu programa deve imprimir o resultado de cada lançamento dos dados.

11. Escreva uma função que receba a hora do dia em segundos (a quantidade de segundos desde a medianoite) e imprima a hora no formato *hh:mm:ss*. Por exemplo se receber 0 imprime 00:00:00, se receber 3725 imprime 01:02:05. A função deve verificar que recebe uma quantidade de segundos válida, entre 0 e a quantidade de segundos que tem um dia.

12. Escreva uma função que calcule o fatorial de um numero inteiro. Utilize essa função num programa que imprime os fatoriais dos números entre 1 e *n*, onde *n* e fornecido via teclado.

13. Escreva uma função que receba as coordenadas de dois pontos (*x0*; *y0*) e (*x1*; *y1*) e retorne a distancia entre esses dois pontos. Use a função num programa que recebe uma lista de pares de pontos e imprime a distancia entre eles, e a distancia média. A quantidade de pares de pontos é desconhecida.

14. Escreva um programa que gere *n* números inteiros aleatórios entre 1 e 100. O valor de *n* deve ser lido pelo teclado. Seu programa deve informar:

- a) O maior numero,
- b) O menor numero,
- c) O produtório dos números pares,
- d) O somatório dos números ímpares,
- e) Imprimir a lista de números,

15. Escreva uma função para desenhar quadrados na tela. A função recebe dois parâmetros; o primeiro toma valores 0 ou 1, 0 indica quadrado vazado e 1 quadrado cheio; o segundo parâmetro a quantidade de asteriscos no lado do quadrado. Utilize a função num programa que imprima o seguinte padrão:

```
****
****
****
****

****
*  *
*  *
*  *
****
```