

Algoritmos e Estruturas de Dados I

Lista de Exercícios – Modularização

Créditos: Lista de exercícios baseados no material da Professora Rosilane Mota.

1. Faça um procedimento que recebe 3 valores inteiros e os exibe em ordem crescente. Crie um programa que lê 3 valores e aciona o procedimento.
2. Faça uma função que recebe um valor inteiro e retorna -1 caso o valor seja negativo, 1 caso seja positivo e 0 caso seja nulo. Crie um programa para testar sua função.
3. [V] Escreva um procedimento que recebe 3 valores reais X, Y e Z correspondentes ao lado de um triângulo e exibe qual o tipo de triângulo (equilátero, isósceles, escaleno), caso o triângulo seja válido. Para que X, Y e Z formem um triângulo, é necessário que cada lado do triângulo seja menor do que a soma dos outros dois lados. Caso o triângulo seja inválido, o procedimento deve imprimir "Triangulo invalido". Faça um programa que lê triângulos (isto é, os lados do triângulo) e, para cada triângulo, aciona o procedimento. A entrada de dados é finalizada quando informado o valor -1. Formato da entrada: conjunto de linhas, onde cada linha traz os lados de um triângulo separados por espaço, sendo a última linha com -1. Formato da saída: conjunto de linhas, cada linha com a classificação do triângulo correspondente: "equilatero", "isósceles" ou "escaleno".
4. [V] Faça um procedimento que recebe a nota de um aluno, identifica e exibe o seu conceito, conforme a tabela abaixo. Faça um programa que lê a média de N alunos, acionando o procedimento para cada um deles. N é fornecido pelo usuário. Formato da entrada: primeira linha contém N, próximas N linhas contendo a nota de cada aluno. Formato da saída: N linhas com o conceito de cada aluno.

Nota	Conceito
Até 39	F
40 a 59	E
60 a 69	D
70 a 79	C
80 a 89	B
A partir de 90	A

5. Escreva uma função que recebe um número inteiro e retorna o fatorial deste número. Crie um programa para testar sua função.
6. Escreva uma função que recebe um valor inteiro e positivo N e retorna o valor de S, sendo S calculado segundo a fórmula abaixo. Crie um programa que lê N e imprime o valor retornado pela função.

$$S = 1 + 1/1! + 1/2! + 1/3! + \dots + 1/N!$$

7. Escreva uma função que recebe por parâmetro um valor inteiro e positivo N e retorna o valor de S, calculando-o segundo a fórmula abaixo.

$$S = 2/4 + 5/5 + 10/6 + 17/7 + 26/8 + \dots + (n^2+1)/(n+3)$$

Faça um programa que lê N e imprime o valor retornado pela função.

8. [V] Faça uma função que calcula e retorna a nota final de um aluno. Para isso, a função recebe as 3 notas do aluno e uma letra. Se a letra for 'A', a nota final do aluno é calculada como a média aritmética das notas; se for 'P', a nota final é a média ponderada das notas, considerando pesos iguais a 5, 3 e 2. Faça um programa que lê as 3 notas de N alunos e o tipo de média a ser utilizada. Armazene as notas em uma matriz. O programa deve imprimir as três notas e a nota final de cada aluno. Para a nota final, o programa deve acionar a função criada. N deve ser informado pelo usuário. Formato da entrada: primeira linha contém N e o tipo de média, próximas N linhas contêm as 3 notas de cada aluno por linha. Formato da saída: N linhas, cada linha contendo as três notas e a nota final de um aluno.

9. [V] A prefeitura de uma cidade fez uma pesquisa entre os seus habitantes, coletando dados sobre o salário e número de dependentes dos adultos. Crie uma função que recebe o salário e a quantidade de dependentes de cada adulto da cidade e retorna a renda per capita da população. Use o seguinte protótipo para sua função:

```
double calcularRendaPerCapita(int n, double salarios[], int dependentes[]);
```

onde n é o total de adultos, salarios guarda os salarios de cada um dos adultos, e dependentes guarda o número de dependentes de cada um dos adultos. Crie um programa para ler os dados (total de adultos, salários e quantidades de dependentes por adulto), chamar a função e imprimir a renda per capita. Formato da entrada: primeira linha contém a quantidade de adultos n, próximas n linhas contêm o salário de cada adulto, próximas n linhas contêm o número de dependentes de cada adulto. Formato da saída: única linha contendo a renda per capita da cidade com 2 casas decimais.

10. Criar uma biblioteca chamada "arrays" que contenha funções úteis para trabalhar com arranjos. Sua biblioteca deve ter os seguintes métodos:

```
void printArray(int n, int v[]);
```

Recebe o array e seu tamanho. Imprime os elementos do array.

```
float mediaArray(int n, int v[]);
```

Recebe o array e seu tamanho. Retorna a média aritmética dos elementos do array.

```
int maxArray(int n, int v[]);
```

Recebe o array e seu tamanho. Retorna o maior elemento do array.

```
int minArray(int n, int v[]);
```

Recebe o array e seu tamanho. Retorna o menor elemento do array.

Criar um programa para testar cada um dos métodos da biblioteca.