

## Lab 2.1 – Condicionais Encadeadas

Objetivos:

- ☐ Fixar o conhecimento de condicionais
- ☐ Compreender condicionais encadeadas

### Exercício 1 – Qual o tipo de triângulo?

Escreva um programa que leia **três números reais**, correspondentes às medidas dos lados de um triângulo, em ordem qualquer.

Dados de **entrada**:

1. Lado A.
2. Lado B.
3. Lado C.

Como **saída**, o programa deverá imprimir:

**“Triângulo: X”**

No comando **print()**, substitua a letra **X** por um dos seguintes valores:

- **“equilátero”**, se todos os três lados forem iguais;
- **“isósceles”**, se apenas dois lados forem iguais;
- **“escaleno”**, se nenhum dos lados forem iguais;
- **“inválido”**, se pelo menos um dos lados for negativo ou se os três lados não formarem um triângulo.

Lembre-se de que as **condições de existência** de um triângulo são:

1. Todos os lados devem ser **positivos**.
2. A medida de qualquer um dos lados deve ser menor que a soma das medidas dos outros dois.

### Exercício 2 – De volta à área do triângulo

Escreva um programa para calcular a área de um triângulo, a partir das medidas dos três lados, fornecidas pelo usuário, em qualquer ordem. O algoritmo não pode permitir a entrada de dados inválidos, ou seja, medidas menores ou iguais a zero, ou medidas que não correspondam às de um triângulo.

## Exercício 3 – Conta de energia

Escreva um programa que determine o **valor total** a ser pago pela conta de energia elétrica, com base nas seguintes **entradas**:

1. O consumo de energia (em kWh); e
2. O tipo de instalação (**R** para residências, **I** para indústrias, e **C** para comércios).

Como saída :

1. Valor total da conta de energia elétrica arredondado para **duas casas decimais**, caso os dados sejam válidos OU a mensagem **"Dados inválidos"** caso os dados sejam inválidos. Os dados são inválidos quando o **consumo é negativo** ou o **tipo de instalação é diferente** das letras **R**, **I** ou **C**.

O valor da conta depende do consumo e do tipo de local. Use a tabela a seguir para calcular o valor devido:

| Preço por tipo e faixa de consumo |               |                 |
|-----------------------------------|---------------|-----------------|
| Tipo                              | Consumo (kWh) | Preço (R\$/kWh) |
| Residencial                       | Até 500       | 0.44            |
|                                   | Acima de 500  | 0.65            |
| Comercial                         | Até 1000      | 0.55            |
|                                   | Acima de 1000 | 0.60            |
| Industrial                        | Até 5000      | 0.55            |
|                                   | Acima de 5000 | 0.60            |

### Validação de dados

Se o usuário inserir valores de entrada inválidos, imprima:

- **"Entradas: X kWh e tipo Y"**
- **"Dados inválidos"**

Se as entradas forem válidas, imprima:

- **"Entradas: X kWh e tipo Y"**
- **"Valor total: R\$ Z"**

Nas mensagens de saída, **substitua as letras X, Y e Z** pelos valores correspondentes.

## Exercício 4 – Intervalo de valores

Considere dois números reais  $a$  e  $b$ , sendo  $b > a$ . Um número real  $x$  pertence ao intervalo  $[a, b]$  se  $a \leq x \leq b$ .

Escreva um programa que leia os números reais  $x$ ,  $a$ ,  $b$ , nesta ordem.

- Se  $x$  pertencer ao intervalo, imprima a seguinte mensagem:  
**“x pertence ao intervalo [a, b]”**
- Caso contrário, imprima a seguinte mensagem:  
**“x não pertence ao intervalo [a, b]”**

Se as entradas forem inválidas, ou seja, se  $b \leq a$ , imprima a seguinte mensagem:  
**“entradas a e b inválidas”**

Nas mensagens, substitua as letras **x**, **a**, **b** pelos valores fornecidos como entrada.

## Exercício 5 – Dois intervalos de valores

Considere dois intervalos numéricos sobre a reta real:  $I_1 = [a, b]$  e  $I_2 = [c, d]$ . Escreva um programa que leia os números reais  $a$ ,  $b$ ,  $c$  e  $d$  (nesta ordem) correspondentes aos intervalos  $I_1$  e  $I_2$  e verifique se existe interseção (pelo menos um ponto em comum) entre os intervalos.

- Se houver interseção, o programa deverá imprimir:  
**“Há intersecção”**
- Se não houver interseção, o programa deverá imprimir:  
**“Não há intersecção”**
- Por fim, se as entradas forem **inválidas**, o programa deverá imprimir:  
**“Entradas invalidas”**

### Observações:

1. Verifique se os intervalos são válidos, ou seja, se  $b > a$  e  $d > c$ .
2. Não pressuponha nada com respeito à posição relativa entre os intervalos  $[a, b]$  e  $[c, d]$ . Ou seja, eles podem estar situados antes ou depois um do outro.

## Exercício 6 – Quadrantes

Escreva um programa que leia 02 valores, **x** e **y**, que representam as coordenadas de um ponto no plano cartesiano.

Como saída, determine em que quadrante (**Q1**, **Q2**, **Q3** ou **Q4**) o ponto está situado, ou se ele está sobre um dos eixos cartesianos (**Eixo X**, **Eixo Y**), ou se ele está na origem  $x = y = 0$  (**Origem**).

## Exercício 7 – Índice de massa corporal (IMC)

Faça um programa que informe o risco de problemas cardíacos de uma pessoa, a partir da leitura da idade e do índice de massa corporal (IMC), nessa ordem. Os riscos são definidos de acordo com a tabela a seguir:

| IMC    | Idade |       |
|--------|-------|-------|
|        | < 45  | ≥ 45  |
| < 22.0 | Baixo | Médio |
| ≥ 22.0 | Médio | Alto  |

### Validação de dados

Verifique se os dados informados são válidos. Se a idade for menor ou igual a zero ou maior que 130 anos, ou se o IMC for menor ou igual a zero, imprima a seguinte mensagem:

**“Entradas: X anos e IMC Y”**

**“Dados inválidos”**

Se as entradas forem **válidas**, imprimir a seguinte mensagem:

**“Entradas: X anos e IMC Y”**

**“Risco: Z”**

Nas mensagens, substitua as letras X, Y e Z pelos valores correspondentes.

## Exercício 8 – Gratificação ao funcionário do ano

Uma empresa decidiu dar uma gratificação diferenciada ao melhor funcionário do ano. A gratificação é determinada com base no número de horas extras trabalhadas e no número de horas não trabalhadas (horas que funcionário faltou ao serviço), conforme a tabela a seguir e o índice **H** determinado da seguinte forma:

$$H = (\text{número de horas extra}) - \frac{1}{4} \times (\text{número de horas não trabalhadas})$$

| Tipo               | Índice H      | Gratificação |
|--------------------|---------------|--------------|
| Melhor funcionário | Maior que 400 | R\$500       |
| Padrão             | Até 400       | R\$100       |

Escreva um programa que leia:

1. O número de horas extras;
2. O número de horas que o funcionário faltou.

Considere cada hora informada como sendo um número real, por exemplo 3.5 horas.

Como saída, imprima a seguinte mensagem:

**“E extras e F de falta”**  
**“R\$ G”**

Onde, **E** é o valor das **horas extras**, **F** é o valor das **horas de faltas** e **G** é o **valor da gratificação**.

## Exercício 9 – Datas

### 3.a. Dia da semana

Escreva um programa que leia um **número inteiro**, correspondente ao dia de hoje na semana. Por exemplo, domingo é 0, segunda é 1, terça é 2, ..., sábado é 6.

Se o usuário digitar um número inteiro diferente destes, imprima:

**“A entrada X é inválida”**, onde **X** é o valor fornecido.

Após isso, peça que o usuário também digite um número de dias no futuro a partir de hoje. Como saída, determine qual é o dia da semana após essa quantidade de dias, com a seguinte mensagem:

**“Hoje é X e o dia futuro é Y”**

#### Caso de exemplo 1

Entre com o número do dia de hoje: 1

Entre com o número de dias após hoje: 3

Hoje é segunda e o dia futuro é quinta

### Caso de exemplo 2

Entre com o número do dia de hoje: 2

Entre com o número de dias após hoje: 8

Hoje é terça e o dia futuro é quarta

### 3.b. Qual o mês?

Escreva um programa que **leia** um **valor inteiro**  $m$  tal que  $1 \leq m \leq 12$ . Como saída, imprima por extenso o **nome do mês** correspondente no ano. Se a entrada não corresponder a nenhum dos meses do ano, imprima: “**número de mês inválido**”.

Fonte: URI Online Judge | 1052