

## Lógica de Programação

### Atividades:

As atividades são atualizadas de acordo com o aprendizado da turma em relação ao conteúdo.

#### (Atividade 1 - MEDIA DO ALUNO)

Desenvolva um programa capaz de receber 3 notas de um determinado aluno. Faça a média aritmética dessas notas e exiba o resultado para o usuário. SE a média for maior ou igual a 7, exiba “Aluno Aprovado”, senão, exiba “Aluno Reprovado”.

(melhorando)

SE a média for maior ou igual a 7, exiba “Aluno Aprovado”, Se o aluno tiver uma média maior que 3, exiba “Aluno em Recuperação”, caso contrário exiba “Aluno Reprovado”.

### RESOLUÇÃO

```
1 programa {
2     funcao inicio() {
3         real n1, n2, n3, media
4
5         escreva("Digite a primeira nota: ")
6         leia(n1)
7         escreva("Digite a segunda nota: ")
8         leia(n2)
9         escreva("Digite a terceira nota: ")
10        leia(n3)
11
12        media = (n1 + n2 + n3) / 3
13
14        escreva("Média do Aluno é: ", media, "\n")
15
16        se(media >= 7){
17            escreva("Aluno está aprovado!")
18        } senao se(media > 3){
19            escreva("Aluno está em recuperação!")
20        } senao{
21            escreva("Aluno está reprovado!")
22        }
23    }
24 }
```

## (Atividade 2 - IMC DE UMA PESSOA)

Desenvolva um programa que calcule o IMC de uma pessoa e mostre o valor na tela. (fórmula = peso dividido pela altura ao quadrado)

SE o IMC for maior ou igual a 30.0, exiba "Cuidado com a Saúde", senão, exiba "Tudo ok"

## RESOLUÇÃO

```
1 programa {
2     funcao inicio() {
3         real altura, peso, imc
4
5         escreva("Digite sua altura: ")
6         leia(altura)
7
8         escreva("digite seu peso: ")
9         leia(peso)
10
11        imc = peso / (altura * altura)
12
13        escreva("Seu IMC é: ", imc, "\n")
14
15        se(imc >= 30){
16            escreva("Cuidado com a Saúde")
17        }senao{
18            escreva("Tudo ok")
19        }
20
21    }
22 }
23
```

**(Atividade 3 – Agenda Comum)**

Desenvolva uma agenda onde é pedido ao usuário as seguintes informações:

(Nome, Idade, Email, Data de nascimento e Telefone). E esses dados devem ser exibidos na tela, onde cada variável estará em uma linha. **Exemplo:**

Nome da pessoa: Fulano

...

**RESOLUÇÃO**

```
1 programa {
2     funcao inicio() {
3         cadeia nome, idade, email, data_nascimento, telefone
4
5         escreva("=====\n")
6         escreva("===== Agenda Comum =====\n")
7         escreva("=====\n")
8
9         escreva("Digite o nome: ")
10        leia(nome)
11
12        escreva("Digite o idade: ")
13        leia(idade)
14
15        escreva("Digite o email: ")
16        leia(email)
17
18        escreva("Digite a data de nascimento: ")
19        leia(data_nascimento)
20
21        escreva("Digite o telefone: ")
22        leia(telefone)
23
24        escreva("=====\n")
25
26        escreva("Nome digitado foi: ", nome, "\n")
27        escreva("Idade digitado foi: ", idade, "\n")
28        escreva("E-mail digitado foi: ", email, "\n")
29        escreva("Data de nascimento digitado foi: ", data_nascimento, "\n")
30        escreva("Telefone digitado foi: ", telefone, "\n")
31
32    }
33 }
```

**(Atividade 4 – Porcentagem de um número)**

Crie um programa capaz de calcular e exibir o percentual correspondente a um valor específico

**Contexto:** O programa vai pedir para o usuário digitar um determinado valor. Após isso irá pedir que digite a porcentagem. O Computador irá **calcular a porcentagem do valor e exibir**.

**Exemplo:** valor digitado -> 100

porcentagem digitada -> 30%

resultado -> 30

**Observação:** não utilizamos o símbolo % na programação. Utilize cálculo matemático.

**RESOLUÇÃO**

```
1 programa {
2     funcao inicio() {
3         real numero, porcentagem, resultado
4
5         escreva("Digite um número [de 1 a 10000]: ")
6         leia(numero)
7
8         escreva("Digite uma porcentagem [de 1 a 100]: ")
9         leia(porcentagem)
10
11        resultado = numero * (porcentagem / 100)
12
13        escreva("O valor do processo é: ", resultado)
14    }
15 }
```

**(Atividade 5 – Dia da Semana)**

Desenvolva um programa onde o usuário irá digitar um número. Cada número representa um dia da semana (conforme seu critério). Caso o usuário digite um valor que não esteja no escopo da semana é exibido para o usuário: **"Digite um dia da semana"**. Caso contrário, ao digitar um dia certo será exibido o dia da semana.

Exemplo:

1. Será exibido **domingo**
2. Será exibido **segunda-feira**
3. Será exibido **terça-feira**
4. Será exibido **quarta-feira**
5. Será exibido **quinta-feira**
6. Será exibido **sexta-feira**
7. Será exibido **sábado**

**RESOLUÇÃO**

```
1 programa {
2     funcao inicio() {
3         inteiro opcao
4
5         escreva("Escolha: \n")
6         escreva("[1] - Para domingo\n")
7         escreva("[2] - Para segunda-feira\n")
8         escreva("[3] - Para terça-feira\n")
9         escreva("[4] - Para quarta-feira\n")
10        escreva("[5] - Para quinta-feira\n")
11        escreva("[6] - Para sexta-feira\n")
12        escreva("[7] - Para sábado\n")
13        escreva("Digite um número: ")
14        leia(opcao)
15
16        se(opcao == 1){
17            escreva("O valor escolhido foi 1 - Domingo")
18        } senao se(opcao == 2){
19            escreva("O valor escolhido foi 2 - segunda-feira")
20        } senao se(opcao == 3){
21            escreva("O valor escolhido foi 3 - terça-feira")
22        } senao se(opcao == 4){
23            escreva("O valor escolhido foi 4 - quarta-feira")
24        } senao se(opcao == 5){
25            escreva("O valor escolhido foi 5 - quinta-feira")
26        } senao se(opcao == 6){
27            escreva("O valor escolhido foi 6 - sexta-feira")
28        } senao se(opcao == 7){
29            escreva("O valor escolhido foi 7 - sábado")
30        } senao {
31            escreva("O valor escolhido não representa um dia da semana ")
32        }
33    }
34 }
35 }
```

**(Atividade 6 – Tabuada)**

Desenvolva um programa que solicite ao usuário a entrada de um número inteiro. O programa deverá exibir a tabuada de 1 até 10 para o número fornecido.

**Instruções:**

- Peça ao usuário para digitar um número inteiro.
- Uma vez que um número inteiro seja fornecido, exiba a tabuada desse número de 1 a 10 em um formato claro e legível. Por exemplo, se o usuário digitar "5", o programa deve exibir:

5 x 1 = 5  
5 x 2 = 10  
5 x 3 = 15  
...  
5 x 10 = 50

**RESOLUÇÃO**

```
1 programa {  
2     funcao inicio() {  
3         inteiro numero  
4  
5         escreva("Digite um número para tabuada: ")  
6         leia(numero)  
7  
8         escreva(" 1 x ", numero, " = ", (1*numero), "\n")  
9         escreva(" 2 x ", numero, " = ", (2*numero), "\n")  
10        escreva(" 3 x ", numero, " = ", (3*numero), "\n")  
11        escreva(" 4 x ", numero, " = ", (4*numero), "\n")  
12        escreva(" 5 x ", numero, " = ", (5*numero), "\n")  
13        escreva(" 6 x ", numero, " = ", (6*numero), "\n")  
14        escreva(" 7 x ", numero, " = ", (7*numero), "\n")  
15        escreva(" 8 x ", numero, " = ", (8*numero), "\n")  
16        escreva(" 9 x ", numero, " = ", (9*numero), "\n")  
17        escreva("10 x ", numero, " = ", (10*numero), "\n")  
18    }  
19 }  
20
```

## (Atividade 7 – Lista de Tarefas)

Desenvolva um programa em que você deverá criar uma lista de tarefas diárias (10 variáveis). O programa deve permitir que o usuário insira pelo menos 10 tarefas que ele precisa realizar durante o dia. Cada tarefa deve ser armazenada em uma variável independente (por exemplo, tarefa1, tarefa2, ..., tarefa10). Após a inserção das tarefas, o programa deve exibir todas elas na tela, uma após a outra.

## RESOLUÇÃO

```
1 programa {
2     funcao inicio() {
3         cadeia t1, t2, t3, t4, t5, t6, t7, t8, t9, t10
4
5         escreva("Digite a tarefa 01: ")
6         leia(t1)
7         /*
8          VAI REPETINDO MANUALMENTE
9          escreva("Digite a tarefa x: ") .. ATÉ 10
10         leia(tx) .. ATÉ 10
11        */
12        escreva("Digite a tarefa 09: ")
13        leia(t9)
14
15        escreva("Digite a tarefa 10: ")
16        leia(t10)
17
18        escreva("tarefa 01 - ", t1)
19        escreva("tarefa 02 - ", t2)
20        escreva("tarefa 03 - ", t3)
21        escreva("tarefa 04 - ", t4)
22        escreva("tarefa 05 - ", t5)
23        escreva("tarefa 06 - ", t6)
24        escreva("tarefa 07 - ", t7)
25        escreva("tarefa 08 - ", t8)
26        escreva("tarefa 09 - ", t9)
27        escreva("tarefa 10 - ", t10)
28    }
```

## (Atividade 8 – Situação de Aprendizagem: Conversão Monetária - FORMATIVA)

Imagine que você faz parte de uma equipe de estudantes de programação que foi desafiada a criar um projeto prático. O projeto consiste no desenvolvimento de um Conversor de Moedas, especificamente para converter valores em Dólar (USD) para Real (BRL). Esse programa será especialmente útil para estudantes de intercâmbio e viajantes que precisam calcular despesas em diferentes moedas.

**O Problema a Resolver:**

Sua tarefa é desenvolver um programa em Portugol que possa converter eficientemente valores em Dólar para Real e vice-versa, considerando a taxa de câmbio atual. O programa deve ser fácil de usar, preciso e fornecer resultados claros.

**Utilize:** Variáveis, estruturas de controle, instruções de entrada, processamento e saída de dados.

## RESOLUÇÃO

```
1 programa {
2     funcao inicio() {
3         real moeda_r, moeda_d, resultado
4         inteiro tipo_moeda
5         escreva ("Qual a cotação do Dólar? ")
6         leia(moeda_d)
7
8         escreva ("Escolha: \n")
9         escreva ("[1] - Converter Dólar para Real \n")
10        escreva ("[2] - Converter Real para Dólar \n")
11
12        escreva ("Digite um numero para conversão: ")
13        leia(tipo_moeda)
14
15        se(tipo_moeda == 1){
16            escreva ("Qual o valor em Dólar a ser convertido para Real? ")
17            leia(moeda_r)
18            resultado = moeda_r * moeda_d
19        } senao{
20            escreva ("Qual o valor em Real a ser convertido para Dólar? ")
21            leia(moeda_r)
22            resultado = moeda_r / moeda_d
23        }
24        escreva ("O valor convertido é: $", resultado)
25    }
26 }
```



## (Atividade 9 – Categoria Jogador de futebol)

Desenvolva um programa que informe a categoria de um jogador de futebol, considerando sua idade:

- **Infantil:** até 13 anos;
- **Juvenil:** até 17 anos;
- **Sênior:** acima de 17 anos.

## RESOLUÇÃO

```
1 programa {
2     funcao inicio() {
3         inteiro idade
4
5         escreva("Digite sua idade: ")
6         leia(idade)
7
8         se(idade <= 13 ){
9             escreva("Sua categoria é: INFANTIL")
10        } senao se(idade <= 17){
11            escreva("Sua categoria é: JUVENIL")
12        } senao{
13            escreva("Sua categoria é: SÊNIOR")
14        }
15    }
16 }
17
```

**(Atividade 10 – Conversão de distância)**

Conversão de medidas: faça um programa que solicite uma distância em metros, calcule e exiba a distância em centímetros.

**RESOLUÇÃO**

```
1 programa {  
2     funcao inicio() {  
3         real distancia, resultado  
4  
5         escreva("CONVERSÃO DE DISTÂNCIA METROS PARA CENTÍMETROS", "\n")  
6         escreva("Digite o valor em metros a ser convertido: ")  
7         leia(distancia)  
8  
9         resultado = distancia * 100  
10  
11         escreva("O valor convertido é: ", resultado)  
12     }  
13 }
```

## (Atividade 11 – Verificação de números)

Utilizando a estrutura condicional, desenvolva um programa que verifique se um número é par ou ímpar e escreva na tela "Número é PAR" (para números pares), "Número é ÍMPAR" (para números ímpares).

## RESOLUÇÃO

```
1 programa {
2     funcao inicio() {
3         inteiro numero, resultado
4
5         escreva("Digite um número: ")
6         leia(numero)
7
8         resultado = numero % 2
9
10        se(resultado == 0){
11            escreva("Número é PAR")
12        }senao{
13            escreva("Número é ÍMPAR")
14        }
15    }
16 }
17
```

**(Atividade 12 – Verificar número é positivo)**

Utilizando a estrutura condicional, desenvolva um programa que verifique se um número é positivo ou negativo. Se ele for positivo escreva “Número é Positivo”, se o número é negativo escreva “Número é Negativo”

**RESOLUÇÃO**

```
1 programa {
2     funcao inicio() {
3         inteiro numero
4
5         escreva("Digite um número: ")
6         leia(numero)
7
8         se(numero > 0){
9             escreva("Número é Positivo")
10        }senao se(numero < 0){
11            escreva("Número é Negativo")
12        }
13    }
14 }
```

## (Atividade 13 – Verificar Senha)

desenvolva um programa que verifique se a senha informada é igual a "AC12". Se sim, exibir a mensagem "Senha correta" e, se não, exibir "Senha errada".

```
1 programa {  
2     funcao inicio() {  
3         cadeia senha  
4  
5         escreva("Digite a senha: ")  
6         leia(senha)  
7  
8         se(senha == "AC12"){  
9             escreva("Senha correta")  
10        }senao{  
11            escreva("Senha errada")  
12        }  
13    }  
14 }  
15
```

## (Atividade 14 – Verificar ANO)

Desenvolva um programa que solicite ao usuário um número inteiro e exibir o mês correspondente a este número, sendo o número 1 o mês de janeiro e 12, o mês de dezembro. Para valores fora da faixa entre 1 e 12, o programa deve informar que não é um mês válido.

```
programa {  
    funcao inicio() {  
        inteiro opcao  
  
        escreva("Escolha: \n")  
        escreva("[1] - Para Janeiro\n")  
        escreva("[2] - Para Fevereiro\n")  
        escreva("[3] - Para Março\n")  
        escreva("[4] - Para Abril\n")  
        escreva("[5] - Para Maio\n")  
        escreva("[6] - Para Junho\n")  
        escreva("[7] - Para Julho\n")  
        escreva("[8] - Para Agosto\n")  
        escreva("[9] - Para Setembro\n")  
        escreva("[10] - Para Outubro\n")  
        escreva("[11] - Para Novembro\n")  
        escreva("[12] - Para Dezembro\n")  
        escreva("Digite um número: ")  
        leia(opcao)  
  
        se(opcao == 1){  
            escreva("O valor escolhido foi 1 - Janeiro")  
        } senao se(opcao == 2){  
            escreva("O valor escolhido foi 2 - Fevereiro")  
        } senao se(opcao == 3){  
            escreva("O valor escolhido foi 3 - Março")  
        } senao se(opcao == 4){  
            escreva("O valor escolhido foi 4 - Abril")  
        } senao se(opcao == 5){  
            escreva("O valor escolhido foi 5 - Maio")  
        } senao se(opcao == 6){  
            escreva("O valor escolhido foi 6 - Junho")  
        } senao se(opcao == 7){  
            escreva("O valor escolhido foi 7 - Julho")  
        } senao se(opcao == 8){  
            escreva("O valor escolhido foi 8 - Agosto")  
        } senao se(opcao == 9){  
            escreva("O valor escolhido foi 9 - Setembro")  
        } senao se(opcao == 10){  
            escreva("O valor escolhido foi 10 - Outubro")  
        } senao se(opcao == 11){  
            escreva("O valor escolhido foi 11 - Novembro")  
        } senao se(opcao == 12){  
            escreva("O valor escolhido foi 12 - Dezembro")  
        } senao {  
            escreva("O valor escolhido não representa um mês correto. ")  
        }  
    }  
}
```

## (Atividade 15 – Livraria)

Uma livraria está fazendo uma promoção para pagamento à vista em que o comprador pode escolher entre dois critérios de desconto:

- a) Critério A: R\$ 0,25 por livro + R\$ 7,50 fixo
- b) Critério B: R\$ 0,50 por livro + R\$ 2,50 fixo

Fazer um programa em que o usuário digite a quantidade de livros que deseja comprar e o programa diga qual é a melhor opção de desconto.

```
programa {
    funcao inicio() {
        inteiro quantidade
        real criterio_A, criterio_B

        escreva("Digite a quantidade de livros para compra: ")
        leia(quantidade)

        criterio_A = (0.25 * quantidade) + 7.50
        criterio_B = (0.50 * quantidade) + 2.50

        escreva("a) Critério A: R$ 0,25 por livro + R$ 7,50 fixo: VALOR FINAL: ",criterio_A, "\n")
        escreva("b) Critério B: R$ 0,50 por livro + R$ 2,50 fixo: VALOR FINAL: ",criterio_B, "\n\n")

        se(criterio_A > criterio_B){
            escreva("A melhor opção é o critério A)", "\n")
        } senao se(criterio_A < criterio_B){
            escreva("A melhor opção é o critério B)", "\n")
        } senao(
            escreva("Critério A e B possuem o mesmo valor")
        )
    }
}
```

## (Atividade 16 – Números Aleatórios)

Crie um programa capaz de gerar números aleatórios em qualquer quantidade, incluindo a possibilidade de repetições. Entretanto, o processo deve ser interrompido imediatamente caso o número 20 seja gerado.

## RESOLUÇÃO

```
1 programa{
2     inclua biblioteca Util --> u
3     funcao inicio(){
4         logico condicao
5         inteiro valor
6         condicao = verdadeiro
7
8         enquanto(condicao){
9             valor = u.sorteia(1, 100)
10            escreva(valor, "\n")
11            se(valor == 20){
12                condicao = falso
13            }
14        }
15    }
16 }
17
```



## (Atividade 18 – Soma infinita)

Crie um programa que armazene um número e o some ao anterior. O processo inicia com o número 0 e continua até que o usuário opte por encerrar.

## RESOLUÇÃO

```
programa{
    inclui biblioteca Util --> u
    funcao inicio(){
        caracter opcao
        inteiro numero, valor
        opcao = 's'
        numero = 0

        enquanto (opcao != 's'){
            escreva("Digita um valor para somar: ")
            leia(valor)
            numero = numero + valor
            escreva("O valor do número é: ", numero, "\n")

            escreva ("deseja continuar? (S/N)")
            leia (opcao)
        }
    }
}
```

**(Atividade 19 – Sorteio finito)**

Crie um programa que irá sortear a quantidade de números que o usuário desejar.

**RESOLUÇÃO**

```
programa{
    inclui biblioteca Util --> u
    funcao inicio(){
        inteiro quantidade

        escreva("Digite a quantidade de números a serem sorteados: ")
        leia(quantidade)

        para(inteiro i = 1; i <= quantidade; i = i + 1){
            escreva(u.sorteia(1, 100), "\n")
        }
    }
}
```

(Atividade 20 – Números Pares)

Crie um programa que irá exibir ao usuário números pares até 100:

(Atividade 21 – Números Ímpares)

Crie um programa que irá exibir ao usuário números ímpares até 99: