Franklin Parrini
Educação e Tecnologia

APOSTILA DO ALUNO · MINICURSO DE PROGRAMAÇÃO

Introdução à Programação com Portugol Studio

Lógica, variáveis, decisão e repetição em Portugol Studio

Duração: 2 horas

Nome do aluno: _____

Turma: _____

Data: _____

1. Objetivos do minicurso

Ao final desta apostila, você deverá ser capaz de:

- Explicar o que é programação e o que é lógica de programação.
- Citar linguagens de programação antigas e modernas.
- Escrever e executar programas simples no Portugol Studio.
- Usar variáveis, constantes, decisões (se) e repetições (para, enquanto).

2. O que é programação?

Um computador não pensa sozinho: ele apenas executa instruções. Um programa é um conjunto de instruções escritas em uma linguagem que o computador consegue entender, e programar é o ato de escrever essas instruções para resolver um problema.

Você usa programas o tempo todo: o aplicativo do banco, o jogo no celular, o semáforo, o caixa do mercado e até o micro-ondas são controlados por programas.

> 💡 **Ideia central**
>
> O computador é muito rápido, mas faz exatamente o que mandamos, nem mais, nem menos. Por isso, quem programa precisa dar ordens claras, completas e na ordem certa.

3. Lógica de programação

Lógica de programação é a forma de organizar o raciocínio para resolver um problema em passos. Esse conjunto de passos, finito e ordenado, chama-se algoritmo. Uma receita de bolo é um bom exemplo de algoritmo.

Exemplo: algoritmo para fazer café

- Ferver a água.
- Colocar o pó no filtro.

- Despejar a água quente no filtro.
- Se quiser café adoçado, colocar açúcar; senão, servir puro.
- Repetir a partir do passo 1 enquanto houver pessoas sem café.

Todo algoritmo é feito de apenas três tipos de blocos:

Bloco	O que faz	Exemplo no café
Sequência	Executa os passos um após o outro, na ordem	Ferver a água, depois colocar o pó
Decisão	Escolhe um caminho conforme uma condição	Se quiser açúcar, adoçar; senão, servir puro
Repetição	Repete passos várias vezes	Enquanto houver pessoas sem café, servir

4. Linguagens de programação

Para conversar com o computador usamos linguagens de programação. Existem centenas delas, criadas em épocas e para finalidades diferentes.

Linguagem	Ano	Para que foi criada / onde é usada
FORTRAN	1957	Cálculos científicos e de engenharia (ainda usada em meteorologia)
COBOL	1959	Sistemas de bancos e empresas (ainda presente em sistemas bancários)
BASIC	1964	Ensino de programação para iniciantes
Pascal	1970	Ensino de programação estruturada
C	1972	Sistemas operacionais e programas de alto desempenho
Python	1991	Ciência de dados, automação, inteligência artificial
Java	1995	Sistemas empresariais e aplicativos Android
JavaScript	1995	Sites e aplicações web
C#	2000	Jogos (Unity), sistemas Windows e web
Kotlin	2011	Aplicativos Android
Swift	2014	Aplicativos para iPhone e Mac

> 📌 **E o Portugol?**
>
> O Portugol é uma linguagem em português criada para ensinar lógica de programação. As ideias que você aprender nele (variáveis, decisões, repetições) valem para todas as outras linguagens da tabela acima.

5. Conhecendo o Portugol Studio

O Portugol Studio é um ambiente gratuito, desenvolvido por uma universidade brasileira, onde você escreve o programa, executa e vê o resultado no console (a área de saída, na parte de baixo da tela).

Como usar

- Abra o Portugol Studio e crie um novo arquivo em branco.
- Digite o programa no editor (área central).
- Clique no botão Executar (▶) para rodar.
- Veja a saída no console e digite os dados pedidos quando o programa usar leia().
- Se aparecer erro, leia a mensagem: ela indica a linha e o problema.

Estrutura básica de todo programa

```
programa {
  funcao inicio() {
    escreva("Olá, mundo!")
  }
}
```

O programa sempre fica dentro de programa { }. A execução começa em funcao inicio(), e tudo que estiver entre as chaves dela será executado em ordem. O comando escreva() mostra um texto na tela; o texto vai entre aspas.

> 💡 **Dica**
>
> Para pular de linha na saída, use \n dentro do texto. Exemplo: escreva("Olá\n"). Comentários começam com // e o computador ignora tudo o que vem depois na linha.

6. Variáveis e constantes

Uma variável é um espaço na memória do computador, com um nome e um tipo, onde guardamos um valor que pode mudar durante o programa. Pense em uma caixa etiquetada: a etiqueta é o nome, e o que está dentro é o valor.

Tipo	Guarda	Exemplo
inteiro	Números sem casa decimal	inteiro idade = 16
real	Números com casa decimal (usa ponto)	real altura = 1.72
cadeia	Textos (entre aspas duplas)	cadeia nome = "Ana"
caracter	Uma única letra (aspas simples)	caracter letra = 'A'
logico	Verdadeiro ou falso	logico ativo = verdadeiro

Entrada e saída de dados

Use escreva() para mostrar valores e leia() para receber o que o usuário digitar. Com o sinal = guardamos um valor na variável (atribuição).

```
programa {
  funcao inicio() {
    cadeia nome
    inteiro idade
    escreva("Qual é o seu nome? ")
    leia(nome)
    escreva("Qual é a sua idade? ")
    leia(idade)
    escreva("Olá, ", nome, "! Você tem ", idade, " anos.\n")
  }
}
```

Constantes

Uma constante é como uma variável cujo valor nunca muda. Usamos a palavra const, e por convenção o nome fica em letras maiúsculas. Se você tentar alterar uma constante, o Portugol Studio mostra um erro.

Exemplo resolvido: área do círculo

```
programa {
  funcao inicio() {
    const real PI = 3.14159
    real raio, area
    escreva("Digite o raio: ")
    leia(raio)
    area = PI * raio * raio
    escreva("Área = ", area, "\n")
  }
}
```

> ⚠ **Regras para nomes de variáveis**
>
> - Comece com letra (nunca com número) e não use espaços nem acentos.
> - Não use palavras reservadas (como programa, se, para, inteiro).
> - Escolha nomes que expliquem o conteúdo: nota1 é melhor que x.
> - Maiúsculas e minúsculas são diferentes: idade e Idade não são a mesma variável.

7. Estruturas de decisão

Nem sempre o programa deve fazer a mesma coisa. Com a estrutura se, o programa escolhe um caminho conforme uma condição. Se a condição for verdadeira, executa o bloco do se; caso contrário, executa o bloco do senao.

```
programa {
  funcao inicio() {
    real nota
    escreva("Digite a nota: ")
    leia(nota)
    se (nota >= 7.0) {
      escreva("Aprovado!")
    } senao se (nota >= 5.0) {
      escreva("Recuperação")
    } senao {
      escreva("Reprovado")
    }
  }
}
```

Operadores para montar condições

Operador	Significado	Exemplo
==	Igual a	idade == 18
!=	Diferente de	opcao != 0
> <	Maior que, menor que	nota > 6
>= <=	Maior ou igual, menor ou igual	nota >= 7
e	As duas condições precisam ser verdadeiras	idade >= 18 e idade <= 65
ou	Basta uma condição ser verdadeira	dia == 6 ou dia == 7
nao	Inverte o resultado (verdadeiro vira falso)	nao ativo

> ⚠ **Atenção**
>
> Um sinal de igual (=) guarda um valor; dois sinais (==) comparam valores. Trocar um pelo outro é um dos erros mais comuns de quem está começando.

8. Estruturas de repetição

Para repetir instruções sem escrevê-las várias vezes, usamos laços de repetição.

para: quando sabemos quantas vezes repetir

```
programa {
  funcao inicio() {
    inteiro n
    escreva("Tabuada de: ")
    leia(n)
    para (inteiro i = 1; i <= 10; i++) {
      escreva(n, " x ", i, " = ", n * i, "\n")
    }
  }
}
```

Dentro dos parênteses do para há três partes separadas por ponto e vírgula: o início (inteiro i = 1), a condição para continuar (i <= 10) e o passo (i++, que soma 1 a cada volta).

enquanto: repete enquanto a condição for verdadeira

```
programa {
  funcao inicio() {
    inteiro senha = 0
    enquanto (senha != 1234) {
      escreva("Digite a senha: ")
      leia(senha)
    }
    escreva("Acesso liberado!")
  }
}
```

	para	enquanto
Quando usar	Sabemos quantas vezes repetir	Não sabemos; depende de uma condição
Exemplo	Contar de 1 a 10	Pedir a senha até acertar

> ⚠ **Cuidado com o laço infinito**
>
> Se a condição do enquanto nunca ficar falsa, o programa repete para sempre. Garanta que alguma variável da condição mude dentro do laço. Se travar, use o botão de parar execução.

9. Exercícios de fixação

Resolva no computador (Portugol Studio) e anote o que for pedido. O gabarito está na seção 11.

1) Faça um programa que mostre na tela seu nome e a sua cidade, cada um em uma linha.

2) Leia dois números inteiros e mostre a soma deles.

3) Leia a idade de uma pessoa e mostre "Maior de idade" (18 anos ou mais) ou "Menor de idade".

4) Mostre na tela os números de 1 a 10 usando o laço para.

5) Peça ao usuário que digite a palavra "sair" e repita o pedido enquanto ele digitar outra coisa. (Dica: variável cadeia e operador !=)

⚠ **Erros comuns**

- Usar vírgula nos números reais (digite 1.72, não 1,72).
- Esquecer de fechar uma chave } ou de colocar as aspas de um texto.
- Usar uma variável sem declará-la antes.
- Confundir = (atribuição) com == (comparação).
- Esquecer de alterar a variável dentro do enquanto, criando um laço infinito.

10. Desafio final

Crie um programa de "Adivinhe o número": o programa guarda um número secreto (use uma constante, por exemplo const inteiro SECRETO = 7), pede palpites ao usuário com enquanto e, a cada tentativa, informa se o palpite é maior, menor ou se acertou. Ao acertar, mostre quantas tentativas foram necessárias.

11. Gabarito

Questão 1

```
programa {
  funcao inicio() {
    escreva("Franklin\n")
    escreva("Rio de Janeiro\n")
  }
}
```

Obs.: cada aluno escreve o próprio nome e a própria cidade; o \n faz pular a linha.

Questão 2

```
programa {
  funcao inicio() {
    inteiro a, b
    escreva("Primeiro número: ")
    leia(a)
    escreva("Segundo número: ")
    leia(b)
    escreva("Soma = ", a + b)
  }
}
```

Questão 3

```
programa {
  funcao inicio() {
    inteiro idade
    escreva("Idade: ")
    leia(idade)
    se (idade >= 18) {
      escreva("Maior de idade")
    } senao {
      escreva("Menor de idade")
    }
  }
}
```

Questão 4

```
programa {
  funcao inicio() {
    para (inteiro i = 1; i <= 10; i++) {
      escreva(i, "\n")
    }
  }
}
```

Questão 5

```
programa {
  funcao inicio() {
    cadeia palavra = ""
    enquanto (palavra != "sair") {
      escreva("Digite a palavra sair: ")
      leia(palavra)
    }
    escreva("Programa encerrado.")
  }
}
```

Desafio final (uma solução possível)

```
programa {
  funcao inicio() {
    const inteiro SECRETO = 7
    inteiro palpite = 0, tentativas = 0
    enquanto (palpite != SECRETO) {
      escreva("Seu palpite: ")
      leia(palpite)
      tentativas++
      se (palpite > SECRETO) {
        escreva("Menor!\n")
      } senao se (palpite < SECRETO) {
        escreva("Maior!\n")
      }
    }
    escreva("Acertou em ", tentativas, " tentativas!")
  }
}
```

12. Resumo para levar

- Programa é um conjunto de instruções; programar é escrever essas instruções para resolver problemas.
- Lógica de programação = organizar o raciocínio em passos (algoritmo), usando sequência, decisão e repetição.
- Existem linguagens antigas (FORTRAN, COBOL, C) e modernas (Python, Java, JavaScript); as ideias básicas são as mesmas.
- Todo programa em Portugol fica em programa { funcao inicio() { ... } }.
- Variável guarda valores que mudam (inteiro, real, cadeia, caracter, logico); constante (const) guarda valores fixos.
- escreva() mostra dados; leia() recebe dados do usuário.
- se / senao se / senao escolhem caminhos; para e enquanto repetem instruções.
- Para aprender a programar, o segredo é praticar: digite, execute, erre e corrija.