

ALGORITMOS E LÓGICA DE PROGRAMAÇÃO

AULA 3 – ESTRUTURAS CONDICIONAIS

WALTER TRAVASSOS SARINHO

@PROF.WALTERTRAVASSOS

WALTER.TRAVASSOS@SECTRAS.EDU.BR



```
RegisterController.php  x app.js
You, 7 months ago | 1 author (You)
import VueRouter from "vue-router";
import routes from "../routes/routes";
import store from "../store/index";
import vuexI18n from "vuex-i18n";
import enLangFile from "../lang/en";

// Set config file into the global variable
window.config = require("../vue.config");

// Import bootstrap file
require("../bootstrap");

// Set the globally
const Vue = require("vue");

require("../directives");

// App-level router management
const router = new VueRouter({
  routes,
  store,
  // ...
});

// Use the globally
Vue.use(VueRouter);
Vue.use(vuexI18n.plugin);
Vue.use(enLangFile);

// ...

```

Aula de Hoje

Estruturas condicionais

- se senao
- escolha caso

Portugol Studio

- <https://portugol.dev> – on-line
- <https://univali-lite.github.io/Portugol-Studio/> - download

Revisão

Tipos de dados: inteiro, real, cadeia e logico.

- Exemplo: inteiro `n1 = 10`

Escrever no console

- Exemplo: `escreva("Digite um número:\n")`

Ler informação do usuário no console

- Exemplo: `leia(n1)`

Exercício de Revisão

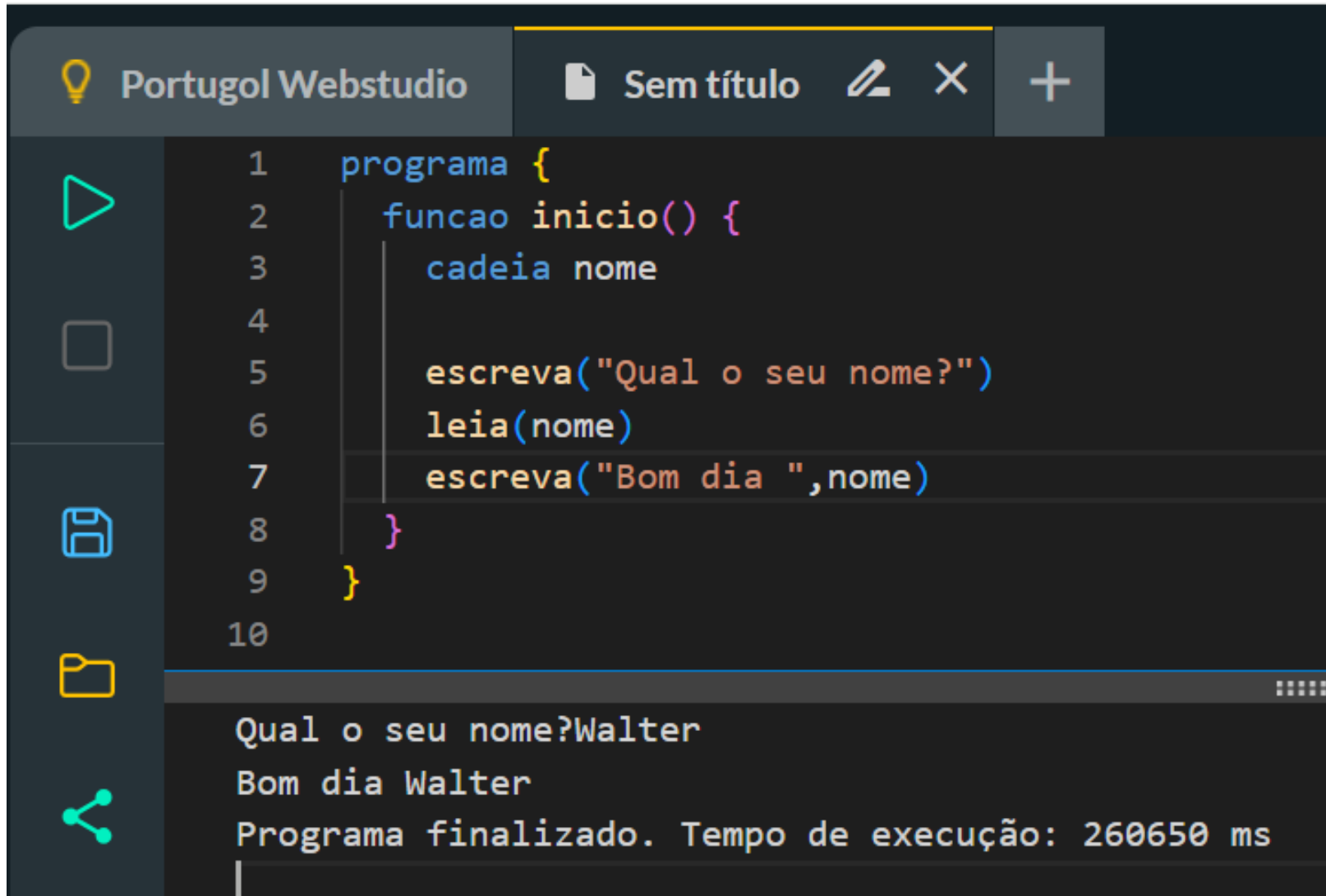
Utilizando o Portugol, faça um algoritmo que pergunta o nome ao usuário e em seguida apresenta a mensagem de “bom dia ” seguido do nome do usuário.

```
Qual o seu nome?Walter
```

```
Bom dia Walter
```

```
Programa finalizado. Tempo de execução: 260650 ms
```

Resposta



The image shows a screenshot of the Portugol Webstudio IDE. The interface includes a top toolbar with a lightbulb icon, a file icon, and a tab labeled "Sem título". On the left, there is a vertical toolbar with icons for running (play), saving (disk), and a folder icon. The main area displays a program in Portugol syntax. The code defines a function named "inicio" that prompts the user for their name and prints a greeting. The output window at the bottom shows the execution results, including the input "Walter" and the output "Bom dia Walter", along with the execution time of 260650 ms.

```
1  programa {  
2      funcao inicio() {  
3          cadeia nome  
4  
5          escreva("Qual o seu nome?")  
6          leia(nome)  
7          escreva("Bom dia ", nome)  
8      }  
9  }  
10
```

Qual o seu nome?Walter
Bom dia Walter
Programa finalizado. Tempo de execução: 260650 ms

Estruturas Condicionais

se

Estruturas Condicionais

- Na vida real tomamos decisões a todo momento, baseadas em escolhas.
- Em algoritmos, chamamos tais decisões de **condições**.
- Associada a uma condição, existirá uma possível alternativa.

Exemplos:

- **Se** a pessoa tiver mais de 18 anos, **então** pode dirigir.
- **Se** a data de validade do pó de café estiver vencida, **então** não deve fazer o café.

Se a data de validade do pó de café estiver vencida, então não deve fazer o café

- A condição do exemplo: “a data de validade do pó de café estiver vencida”.
- Possíveis decisões baseadas na condição: Sim (Verdadeiro) ou não (Falso). Logo, condições geram respostas **lógicas**!
- Se a condição for verdadeira, a ação de “**não fazer o café**” será executada.
- Caso contrário (**senão**), o café será coado.

Estruturas Condicionais

- Para criarmos estruturas condicionais simples no Portugol, utilizamos as palavras reservadas **se**.
- Sintaxe:

```
se (condição) {  
    // Instruções  
}
```

Exercício 1

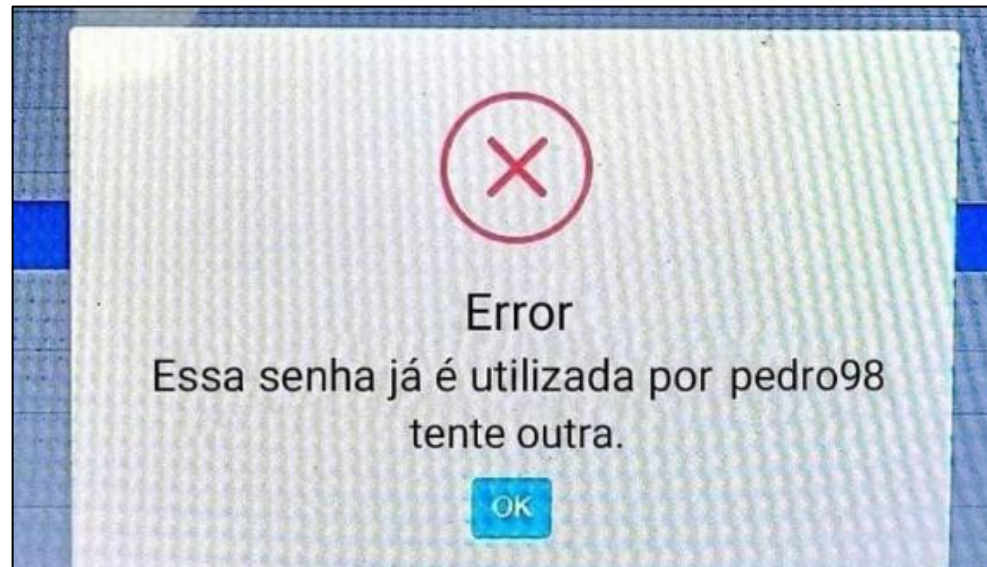
Faça um algoritmo usando o Portugol para **ler a idade** de um indivíduo e **informar se o mesmo atingiu a maioridade**.

Exercício 1 - Resposta

```
programa {  
    funcao inicio() {  
        inteiro idade  
        escreva("Digite sua idade: ")  
        leia(idade)  
        se (idade >= 18) {  
            escreva("Atingiu maioridade")  
        }  
    }  
}
```

Exercício 2

Faça um algoritmo que leia a senha digitada pelo usuário e informe se a mesma é válida (supondo que a senha válida é “12345seis”).



Exemplo

```
programa {  
    funcao inicio() {  
        cadeia senha  
        escreva("Digite sua senha: ")  
        leia(senha)  
        se (senha == "12345seis") {  
            escreva("Senha válida")  
        }  
    }  
}
```

Estruturas Condicionais

Podemos criar estruturas condicionais com duas opções de escolha no Portugol, utilizamos as palavras reservadas **se**, e **senao**.



Sintaxe: se senao

```
se (condicao) {  
    // Instruções que serão executadas se  
    // a condição for verdadeira  
}  
senao {  
    // Instruções que serão executadas se  
    // a condição for falsa  
}
```


Sintaxe: se senao se

```
se (condicao) {  
    // Instruções que serão executadas se  
    // a condição for verdadeira  
}  
senao se (condicao) {  
    // Instruções que serão executadas se  
    // a segunda condição for verdadeira  
} senao {  
    // Instruções que serão executadas se  
    // nenhuma condição for verdadeira  
}
```

Mais Exercícios

Exercício 3

- Faça um algoritmo para ler três notas, calcular a média e, em seguida, informar se o aluno está aprovado ou não.
- O aluno estará aprovado se sua média for maior ou igual a 7.

```
Digite a primeira nota
```

```
8
```

```
Digite a segunda nota
```

```
9
```

```
Digite a terceira nota
```

```
10
```

```
Aprovado. Média: 9
```

```
Programa finalizado. Tempo de execução: 3055 ms
```

Exercício 3 - Resposta

```
1  programa {
2      funcao inicio() {
3          real n1, n2, n3, media
4          escreva("Digite a primeira nota\n")
5          leia(n1)
6          escreva("Digite a segunda nota\n")
7          leia(n2)
8          escreva("Digite a terceira nota\n")
9          leia(n3)
10         media = (n1+n2+n3) / 3
11         se (media >= 7) {
12             escreva("\nAprovado. Média: ", media)
13         } senao {
14             escreva("\nReprovado. Média: ", media)
15         }
16     }
17 }
```

```
Digite a primeira nota
8
Digite a segunda nota
9
Digite a terceira nota
10

Aprovado. Média: 9
Programa finalizado. Tempo de execução: 3055 ms
```

Exercício 4

- Faça um algoritmo para ler dois números e um código.
- O código vai identificar a operação que será efetuada entre os dois números lidos (de acordo com a tabela abaixo).
- O algoritmo deverá exibir o resultado da operação escolhida pelo usuário.

CÓDIGO	OPERAÇÃO
1	ADIÇÃO
2	SUBTRAÇÃO

```
Digite o primeiro número
2
Digite o segundo número
4
1 para somar, 2 para subtrair3
operação inválida
Programa finalizado. Tempo de execução: 3950 ms
|
```

Exercício 4 - Resposta

```
1  programa {
2      funcao inicio() {
3          inteiro n1, n2, op
4          escreva("Digite o primeiro número\n")
5          leia(n1)
6          escreva("Digite o segundo número\n")
7          leia(n2)
8          escreva("1 para somar, 2 para subtrair")
9          leia(op)
10         se (op == 1) {
11             escreva("A soma é: \n", (n1+n2))
12         } senao se (op == 2) {
13             escreva("A subtração é: ", (n1-n2))
14         } senao {
15             escreva ("operação inválida")
16         }
17     }
18 }
```

```
Digite o primeiro número
2
Digite o segundo número
4
1 para somar, 2 para subtrair3
operação inválida
Programa finalizado. Tempo de execução: 3950 ms
```

Exercício 5

Faça um algoritmo para ler o salário atual de um funcionário e calcular seu salário reajustado. O percentual de aumento encontra-se na tabela abaixo:

SALÁRIO	% DE AUMENTO
Se for menor ou igual a R\$ 1000,00	35%
Se maior que R\$ 1000,00	15%

```
Qual o valor do seu salário?  
1000  
Seu salário reajustado é: R$ 1350
```

Exercício 5 - Resposta

```
1  programa {
2      funcao inicio() {
3          real salario
4          escreva("Qual o valor do seu salário?\n")
5          leia(salario)
6          se (salario <= 1000) {
7              escreva ("Seu salário reajustado é: R$ ", (salario*1.35) )
8          } senao {
9              escreva ("Seu salário reajustado é: R$ ", (salario*1.15) )
10             }
11         }
12     }
13
```

.....

Qual o valor do seu salário?

1000

Seu salário reajustado é: R\$ 1350

Exercício 6

1. Faça um algoritmo que receba o valor do salário de uma pessoa e o valor de um financiamento pretendido.
2. Se o financiamento for menor ou igual a 5 vezes o salário da pessoa, o algoritmo deverá escrever “Financiamento concedido”.
3. Senão, deverá escrever “Financiamento negado”.
4. Independente de conceder ou não o financiamento, o algoritmo deverá escrever a frase “Obrigado por nos consultar”.

```
Qual o valor do seu salário?  
1000  
Qual o valor do financiamento pretendido?  
5000  
5000  
Financiamento concedido  
Obrigado por nos consultar
```

Exercício 6 - Resposta

```
1  programa {
2      funcao inicio() {
3          real salario, financiamento, limite
4          escreva("Qual o valor do seu salário?\n")
5          leia(salario)
6          escreva("Qual o valor do financiamento pretendido?\n")
7          leia(financiamento)
8          limite = salario * 5
9          escreva(limite)
10         se (financiamento <= limite) {
11             escreva("\nFinanciamento concedido")
12         } senao {
13             escreva("\nFinanciamento negado")
14         }
15         escreva("\nObrigado por nos consultar")
16     }
17 }
```

```
Qual o valor do seu salário?
1000
Qual o valor do financiamento pretendido?
5000
5000
Financiamento concedido
Obrigado por nos consultar
```

Intervalo: 15 minutos

Estruturas Condicionais

Avaliando mais de uma condição

Estruturas Condicionais

- Também podemos implementar estruturas condicionais com mais de uma condição (expressão).
- Subexpressões podem formar uma única expressão a partir do uso dos operadores lógicos **e** ou **ou**.
- Para sabermos se determinada expressão será verdadeira ou falsa, deveremos checar a saída lógica de cada subexpressão.

Vamos falar um pouco sobre a **Tabela Verdade**

- As tabelas verdade são ferramentas importantes no campo da Lógica.
- Os exercícios e questões da Lógica, muitas vezes, envolvem a análise e descrição de **sentenças propositivas**.
- Nas condições das estruturas de decisão, podemos utilizar operadores lógicos para avaliar mais de uma condição (sentença lógica) ao mesmo tempo.
- A tabela verdade busca extrair o **sentido verdadeiro de uma sentença lógica**, ou seja, vamos entender como funcionam os operadores lógicos.

Operadores Lógicos

- NÃO
- E
- OU

Uso do operador lógico **NAO**

- Símbolo: $\sim$
- Operação lógica: **negação**.
- Designa um valor de falsidade quando a afirmação se propor verdadeira e vice-versa.
- Operador no português: **nao**

p	$\sim p$
V	F
F	V



!false
pode não parecer,
mas é verdadeiro.

Exemplo

Faça um algoritmo com uma **variável lógica** recebendo o valor **verdadeiro** e em seguida use o **operador de negação** para converter seu valor para **falso**. Escreva o valor final da variável.

Resposta

```
1  programa {  
2      funcao inicio() {  
3          logico teste = verdadeiro  
4          teste = nao(teste)  
5          escreva(teste)  
6      }  
7  }  
8
```

falso

Podemos usar a palavra "nao"

Uso do operador lógico **E**

- Símbolo: $\wedge$
- Operação lógica: **conjunção**.
- Valor lógico: é verdade apenas quando todas as proposições forem verdadeiras.

p	q	$p \wedge q$
V	V	V
V	F	F
F	V	F
F	F	F

Exemplo

Faça um algoritmo que lê três números e informa qual o maior número digitado (considere nos testes que o usuário sempre irá digitar valores distintos para cada número).

Exemplo

```
programa {  
    funcao inicio() {  
        inteiro num1,num2,num3  
        escreva("Digite 3 números:\n")  
        leia(num1,num2,num3)  
        se ((num1>num2) e (num1>num3)) {  
            escreva(num1)  
        }  
    }  
}  
  
// ...
```

```
// ...  
    se ((num2>num1) e (num2>num3)) {  
        escreva(num2)  
    }  
    se ((num3>num1) e (num3>num2)) {  
        escreva(num3)  
    }  
}  
}
```

Uso do operador lógico **OU**

- Símbolo: $\vee$
- Operação lógica: **disjunção**.
- Valor lógico: terá valor verdadeiro quando apenas uma das partículas for verdade e a outra falsa.

p	q	$p \vee q$
V	V	V
V	F	V
F	V	V
F	F	F

Exemplo

Faça um algoritmo que lê uma sigla de um estado brasileiro (considere que o usuário só ira digitar um dos seguintes estados: PE, PB, SP ou RJ) e informa se o estado digitado pertence ao Nordeste ou ao Sudeste.

Exemplo

```
programa {  
    funcao inicio() {  
        cadeia sigla  
        escreva("Digite uma sigla (PE, PB, RJ, SP):")  
        leia(sigla)  
        se ((sigla == "PE") ou (sigla == "PB")) {  
            escreva("É do Nordeste!")  
        }  
        se ((sigla == "RJ") ou (sigla == "SP")) {  
            escreva("É do Sudeste!")  
        }  
    }  
}
```

Mais Exercícios

Exercício 7

Faça um algoritmo que verifica quanto cada pessoa deverá pagar para entrar em um cinema. O algoritmo deverá solicitar o sexo e a idade do indivíduo. As condições de pagamento são as seguintes:

- O ingresso custa R\$ 10,00 para pessoas com idade maior ou igual a 17 anos.
- Mulheres com idade inferior a 17 anos pagam meia entrada (R\$ 5,00).
- Homens com idade inferior a 17 anos pagam R\$ 7,00.

Exercício

```
programa {  
    funcao inicio() {  
        cadeia sexo  
        inteiro idade  
        escreva("Digite o sexo e a idade do cliente:\n")  
        leia(sexo,idade)  
        se (idade >= 17) {  
            escreva("Total a Pagar: R$ 10,00")  
        }  
    }  
}
```

```
        se ((idade < 17) e (sexo == "F")) {  
            escreva("Total a Pagar: R$ 5,00")  
        }  
        se ((idade < 17) e (sexo == "M")) {  
            escreva("Total a Pagar: R$ 7,00")  
        }  
    }  
}
```

```
1 programa {
2     funcao inicio() {
3         caracter sexo
4         inteiro idade
5
6         escreva("M para masculino. F para feminino: \n")
7         leia (sexo)
8         escreva("Qual sua idade: \n")
9         leia(idade)
10        se (idade >=17 ) {
11            escreva("Ingresso custa R$ 10,00")
12        } senao se (sexo=="M") {
13            escreva("Ingresso custa R$ 7,50")
14        } senao se (sexo=="F") {
15            escreva("Ingresso custa R$ 5,00")
16        } senao {
17            escreva("Opção inválida")
18        }
19    }
20 }
```

Comando de Seleção Múltipla
escolha caso

Comando de Seleção Múltipla

- Em algumas situações, existem situações mutuamente exclusivas:
- Se uma situação for executada, as demais não serão.
- Em casos desse tipo, um comando de seleção múltipla é mais indicado do que o comando **SE**.

Sintaxe

```
escolha (VARIABEL) {  
    caso VALOR_1:  
        // Instruções que serão executadas caso  
        // VARIABEL seja igual a VALOR_1  
    pare  
    caso VALOR_2:  
        // Instruções que serão executadas caso  
        // VARIABEL seja igual a VALOR_2  
    pare  
    caso contrario:  
        // Instruções que serão executadas caso  
        // VARIABEL NÃO seja igual a VALOR_1 e VALOR_2  
}
```


Comando de Seleção Múltipla

Escolha Caso

- Esta estrutura seleciona uma variável que será comparada com os valores especificados ao lado da palavra reservada **caso**.
- O caso tem funcionamento similar ao **se**.
- O **caso contrario** tem funcionamento similar ao **senao**, visto anteriormente.

Exemplo

Faça um algoritmo que lê o sexo de um indivíduo (M ou F) e informa se o mesmo é homem ou mulher.

Exemplo

```
programa {  
  funcao inicio() {  
    caracter sexo  
    escreva("Digite seu sexo:\n")  
    leia(sexo)  
    escolha(sexo) {  
      caso 'M':  
        escreva("Masculino")  
      pare  
      caso 'F':
```

```
//caso 'F':  
        escreva("Feminino")  
      pare  
    caso contrario:  
      escreva("Caractere inválido")  
    }  
  }  
}
```

Exercícios

Exercício 1

Faça um algoritmo que lê um número de 1 a 7 e informa o dia da semana correspondente, sendo domingo o número 1.

```
Digite um número:
```

```
6
```

```
Sexta
```

```
Programa finalizado. Tempo de execução: 3272 ms
```

Exercício 1

Resposta

```
1  programa {
2      funcao inicio() {
3          inteiro dia
4          escreva("Digite um número:\n")
5          leia(dia)
6          escolha(dia) {
7              caso 1: escreva("Domingo") pare
8              caso 2: escreva("Segunda") pare
9              caso 3: escreva("Terça") pare
10             caso 4: escreva("Quarta") pare
11             caso 5: escreva("Quinta") pare
12             caso 6: escreva("Sexta") pare
13             caso 7: escreva("Sábado") pare
14             caso contrario: escreva("Número inválido")
15         }
16     }
17 }
```

Digite um número:

6

Sexta

Programa finalizado. Tempo de execução: 3272 ms

Exercício 2

1. Faça um algoritmo que receba dois números.
2. Em seguida, receba um caractere que representa uma operação matemática (+, -, *, /)
3. Por fim, efetue o cálculo matemático baseado na escolha do usuário.
4. O resultado deverá ser exibido em seguida.

```
9
Digite outro número
3
Qual a operação? + - * /
/
Divisão: 3
```

Exercício 2 - Resposta

```
programa {  
    funcao inicio() {  
        real n1, n2  
        caracter op  
        escreva("Digite um número\n")  
        leia(n1)  
        escreva("Digite outro número\n")  
        leia(n2)  
        escreva("Qual a operação? + - * /\n")  
        leia(op)  
        escolha(op) {  
            caso '+': escreva("Soma: ", (n1+n2)) pare  
            caso '-': escreva("Subtração: ", (n1-n2)) pare  
            caso '*': escreva("Multiplicação: ", (n1*n2)) pare  
            caso '/': escreva("Divisão: ", (n1/n2)) pare  
            caso contrario: escreva("Operação inválida.")  
        }  
    }  
}
```

```
9  
Digite outro número  
3  
Qual a operação? + - * /  
/  
Divisão: 3
```


Exercício 3

- Simulador de Batalha RPG:
- Implemente uma etapa de batalha de um jogo simples em que o jogador vai escolher entre diferentes ataques em uma batalha contra um monstro, com resultados diferentes dependendo da escolha o golpe especial de acordo com a tabela abaixo.

Número	Ataque	Nível de Dano
1	Kamehameha	100
2	Hadouken	30
3	Razengan	50
4	Meteoro de Pegasus	120

```
Escolha o golpe:
1 - Kamehameha
2 - Hadouken
3 - Razengan
4 - Meteoro de Pegasus
1
Aplicou 100 de dano.
```

Exercício 3 - Resposta

```
1  programa {
2      funcao inicio() {
3          inteiro opcao
4          escreva("Escolha o golpe:\n1 - Kamehameha\n")
5          escreva("2 - Hadouken\n3 - Razengan\n4 - Meteoro de Pegasus\n")
6          leia(opcao)
7          escolha(opcao) {
8              caso 1: escreva("Aplicou 100 de dano.") pare
9              caso 2: escreva("Aplicou 30 de dano.") pare
10             caso 3: escreva("Aplicou 50 de dano. ") pare
11             caso 4: escreva("Aplicou 120 de dano. ") pare
12             caso contrario: escreva("opção inválida. ")
13         }
14     }
15 }
```

ALGORITMOS E LÓGICA DE PROGRAMAÇÃO

AULA 3 – ESTRUTURAS CONDICIONAIS

WALTER TRAVASSOS SARINHO

@PROF.WALTERTRAVASSOS

WALTER.TRAVASSOS@SECTRAS.EDU.BR



```
RegisterController.php  x app.js x
You, 7 months ago | 1 author (You)
import VueRouter from "vue-router";
import routes from "../routes/routes";
import store from "../store/index";
import vuexI18n from "vuex-i18n";
import enLangFile from "../lang/en";

// Set config file into the global variable
window.config = require("../vue.config");

// Import bootstrap file
require("../bootstrap");

// Set the globally
const Vue = require("vue");

require("../directives");

// App-level router management
const router = new VueRouter({
  routes,
  store,
  // ...
});

// ...
Vue.use(VueRouter);
Vue.use(store);
Vue.use(vuexI18n);
Vue.use(enLangFile);
```