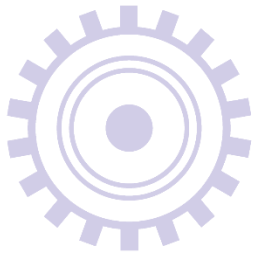


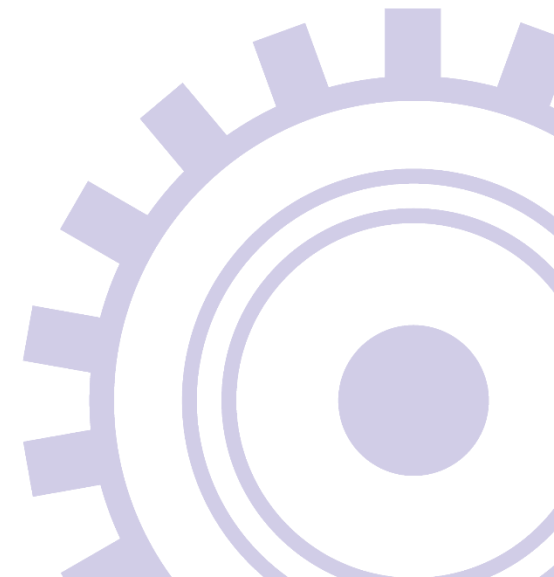
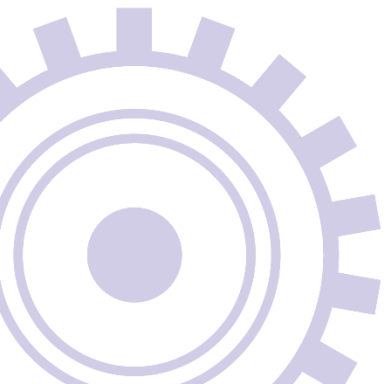
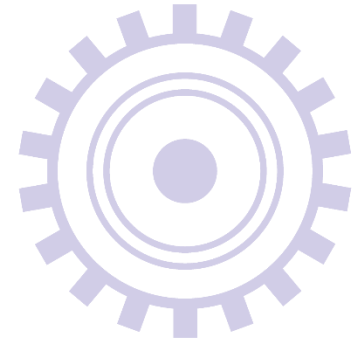


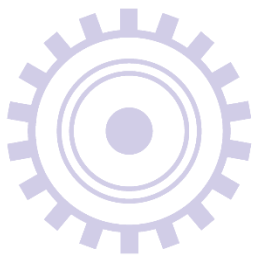
Estrutura condicional em pseudocódigo: SE, SENAO, SENAO SE





Muitas vezes precisamos tomar decisões que podem interferir diretamente no andamento do algoritmo. A representação dessas decisões em nossos programas é feita através do uso de estruturas de seleção, ou estruturas de decisão. A estrutura de seleção que utilizaremos em Portugol (VisualG) será a estrutura SE...ENTAO... SENAO.



**Sintaxe:**

```
SE <expressão lógica> ENTAO
    <bloco de instruções verdade>
FIMSE
```

Sintaxe:

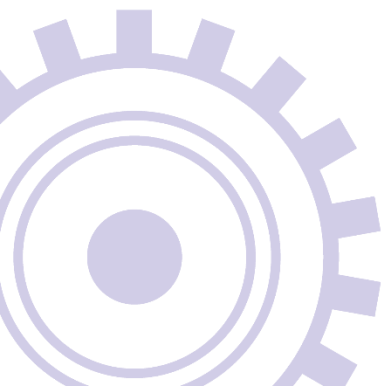
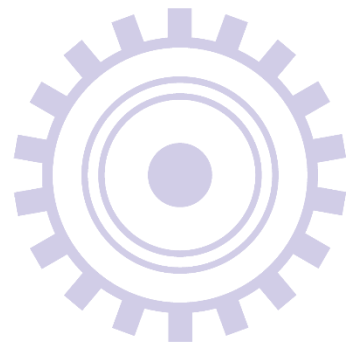
```
SE <expressão lógica> ENTAO
    <bloco de instruções verdade>
SENAO
    <bloco de instruções falso>
FIMSE
```

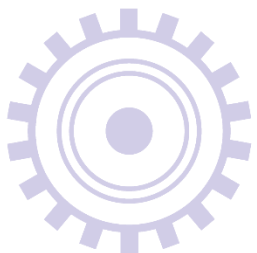
Exemplo VisualG

```
Se media > 6 entao
    Escreva("Aprovado")
fimse
```

Exemplo VisualG

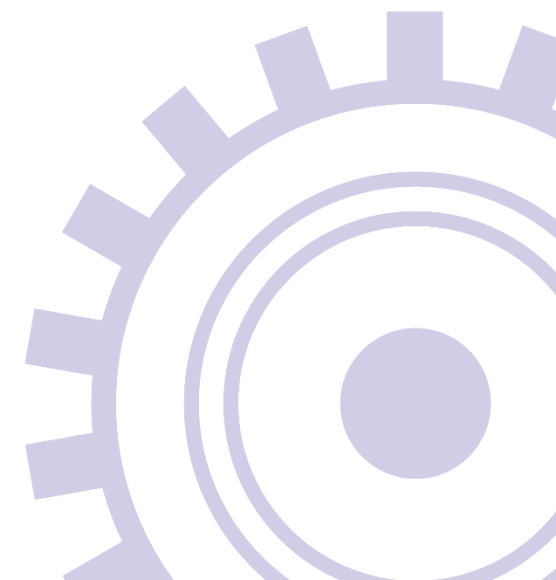
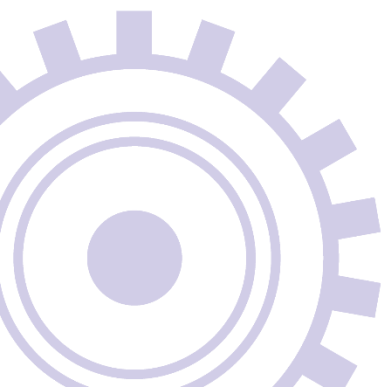
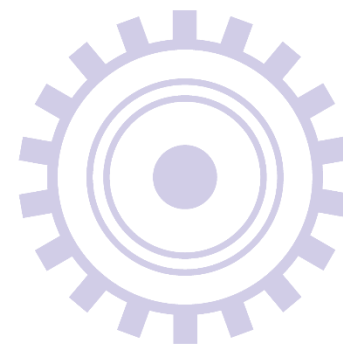
```
Se media > 6 entao
    Escreva("Aprovado")
Senao
    Escreva("Reprovado")
fimse
```

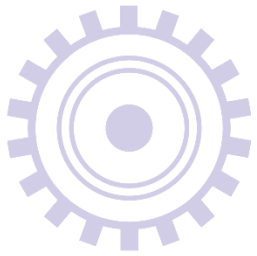




Como funciona?

A palavra reservada SE indica o início da estrutura de seleção. Após essa palavra, vem a condição que definirá o bloco a ser executado. Qualquer expressão lógica poderá ser utilizada como condição, pois deverá retornar verdadeiro ou falso. Caso a expressão da condição seja verdadeira, o bloco de instruções ENTAO será executado. Caso contrário, o bloco SENAO o será. A palavra reservada FIMSE indica o final da estrutura de seleção. Vale ressaltar que o bloco SENAO não é obrigatório: caso só queiramos executar algumas instruções se a expressão lógica for verdadeira, podemos omitir o bloco SENAO.





Desvio condicional sequencial

O desvio condicional sequencial ocorre quando se utiliza tomadas de decisão simples ou composta definidas sucessivamente uma após a outra. O exemplo abaixo mostra de forma simplificada esse tipo de estrutura de decisão.

Se (condição 1) entao

[Ação para condição 1 verdadeira]

fimse

Se (condição 2) entao

[Ação para condição 2 verdadeira]

fimse

Se (condição 1) entao

[Ação para condição 1 verdadeira]

Senao

[Ação para condição 1 Falsa]

fimse

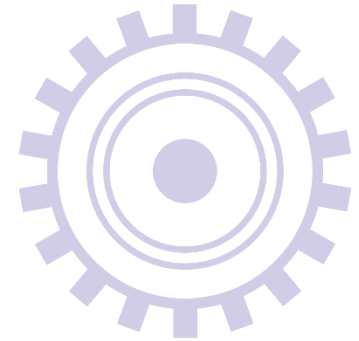
Se (condição 2) entao

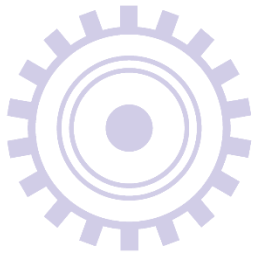
[Ação para condição 2 verdadeira]

Senao

[Ação para condição 2 Falsa]

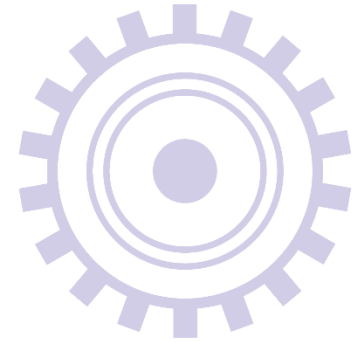
fimse



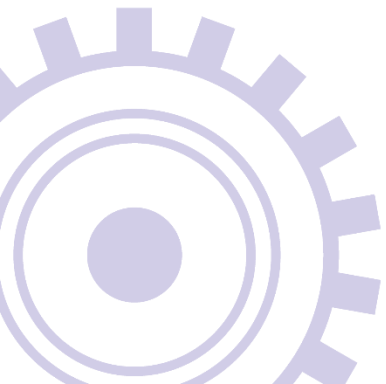


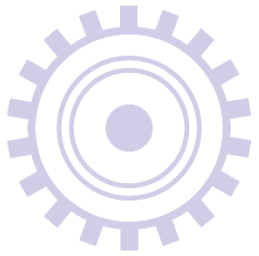
Decisão com operadores lógicos – (E , OU)

Pode ser que você necessite, em algum momento, trabalhar com o relacionamento de duas ou mais condições ao mesmo tempo na mesma instrução se, efetuando desta forma testes múltiplos, ou efetuar a negação de uma condição. Para estes casos é necessário trabalhar com os operadores lógicos. (E , OU)



```
algoritmo "TESTE_LOGICO_E"  
var  
numero:inteiro  
inicio  
    Leia(numero)  
    se (numero >=20) e (numero <=90) entao  
        escreva("O numero esta na faixa de 20 a 90")  
    senao  
        escreva("O numero esta fora da faixa de 20 a  
90")  
    fimse  
finalgoritmo
```





Decisão com operadores lógicos – (E , OU)

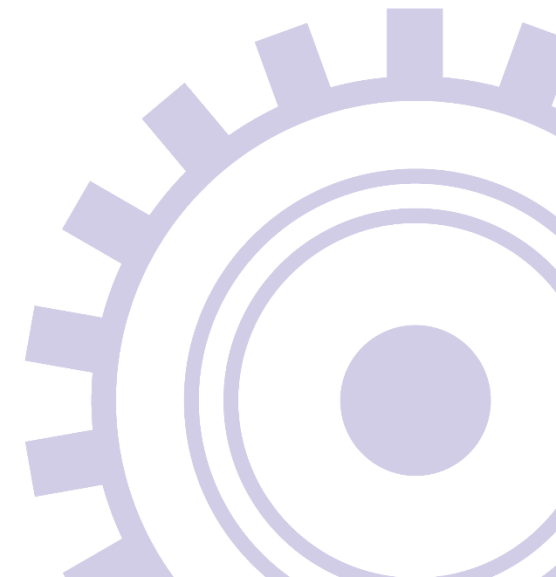
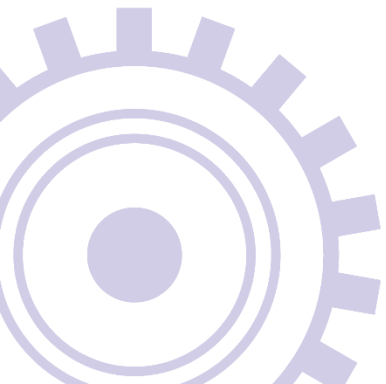
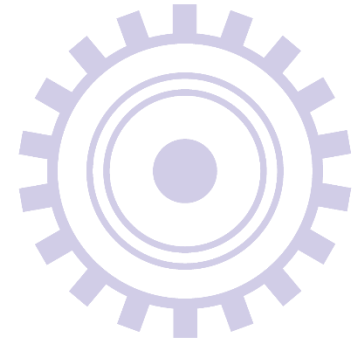
```
algoritmo "TESTE_LOGICO_OU"
```

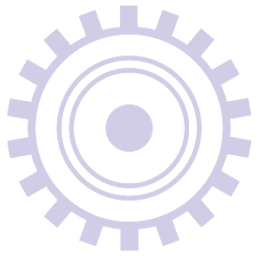
```
var  
usuario:caracter
```

```
inicio
```

```
Leia(usuario)  
se (usuario = "Administrador") ou (usuario =  
"Comum") entao  
    escreva("O seu usuário é válido")  
senao  
    escreva("O usuário não é valido")  
fimse
```

```
finalgoritmo
```





Decisão com operadores lógicos – (E , OU)

```
algoritmo "TESTE_LOGICO_OU"
```

```
var
```

```
usuario:caracter
```

```
inicio
```

```
  Leia(usuario)
```

```
  se (usuario = "Administrador") ou (usuario = "Comum") entao
```

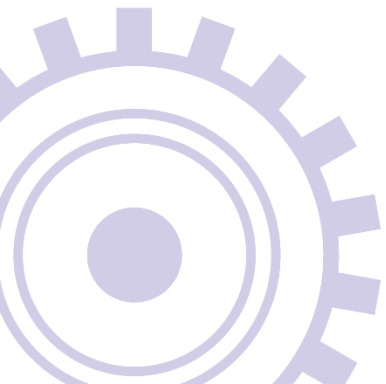
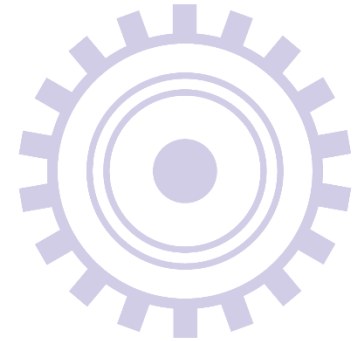
```
    escreva("O seu usuário é válido")
```

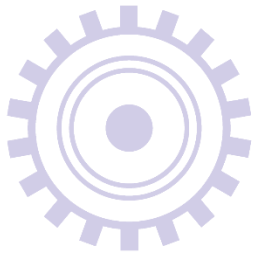
```
  senao
```

```
    escreva("O usuário não é valido")
```

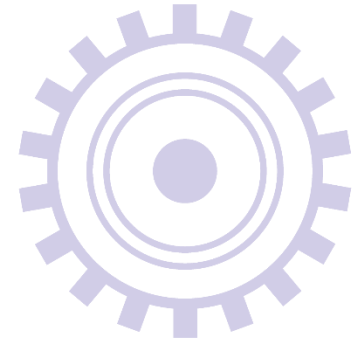
```
  fimse
```

```
fimalgoritmo
```





É hora da prática

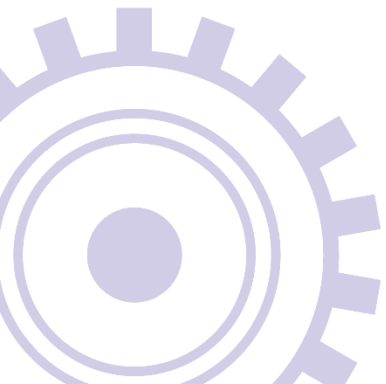


1 - Desenvolver um algoritmo para encontrar a média de duas notas e mostrar se o aluno foi aprovado ou reprovado, sendo que, o critério para a aprovação é ter média superior a 6.

Exemplo:

Nota1 = 7, Nota2 = 8,

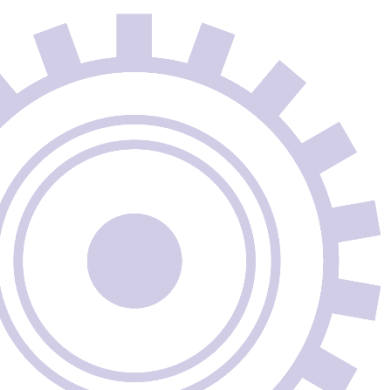
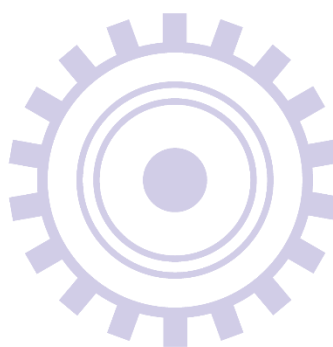
Media = $(7 + 8) / 2$, Media = 7,5 = Aprovado

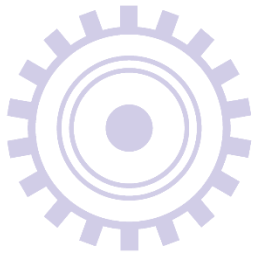




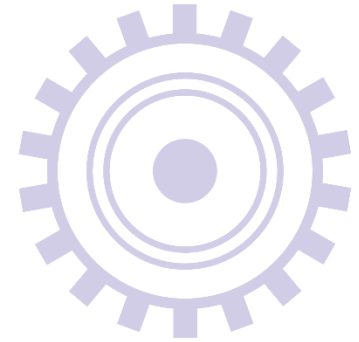
Resolução em VisualG

```
1 Algoritmo "Condicional01"
2
3 // Professor : Flávio Mota da Cruz
4
5 // Desenvolver um algoritmo para encontrar a média
6 // de duas notas e mostrar se o aluno foi aprovado ou reprovado,
7 // sendo que, o critério para a aprovação é ter média superior a 6.
8
9 Var
10     nota1,nota2,media:real
11 Inicio
12
13     Escreva("Digite o valor da Primeira nota: ")
14     leia(nota1)
15     Escreva("Digite o valor da Segunda nota: ")
16     leia(nota2)
17     media <- (nota1 +nota2)/ 2 // Calculando a média
18     se (media > 6) entao // Verificando a condição da média
19         escreva("Aprovado") // Valor se verdadeiro
20     senao
21         escreva("Reprovado") // Valor se Falso
22     fimse // Fim da condição
23
24 Fimalgoritmo
```

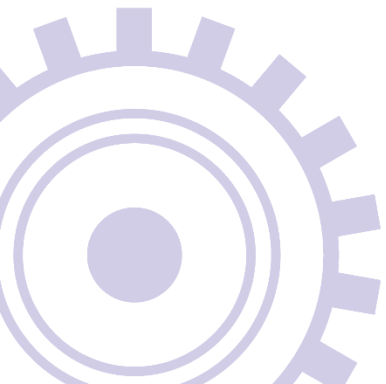




É hora da prática



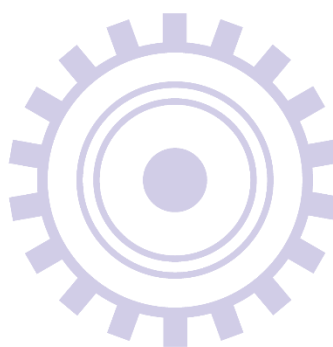
2 – Dado um dia da semana (segunda, terça, quarta, quinta, sexta, sábado e domingo), faça um algoritmo que escreva “Casa” para “Sábado” e “Domingo” e escreva “Trabalho” para os demais dias.

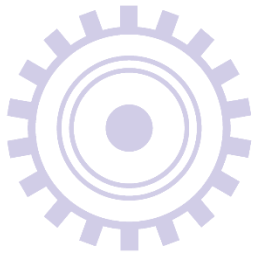




Resolução em VisualG

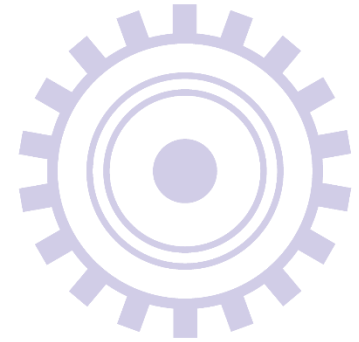
```
1 Algoritmo "Condicional01"  
2  
3 // Professor : Flávio Mota da Cruz  
4  
5 // Sendo dado um dia da semana (Segunda, Terça, Quarta,  
6 // Quinta, Sexta, Sábado e Domingo), faça um algoritmo  
7 // que escreva "Casa" para "Sábado" e "Domingo" e escreva  
8 // "Trabalho" para os demais dias.  
9  
10 Var  
11     diaSemana: caracter  
12 Inicio  
13  
14     Escreva("Digite o dia da semana: ")  
15     leia(diaSemana)  
16  
17     se ((diaSemana = "Sabado") ou (diaSemana = "Domingo")) entao  
18         escreva("Casa") // Valor se verdadeiro  
19     senao  
20         escreva("Trabalho") // Valor se Falso  
21     fimse // Fim da condição  
22  
23 Fimalgoritmo
```



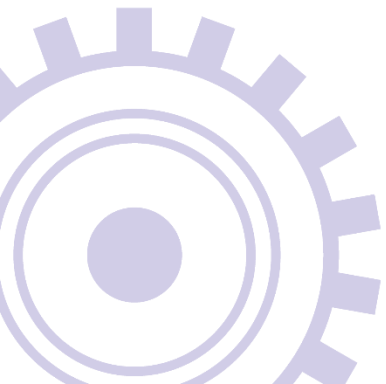


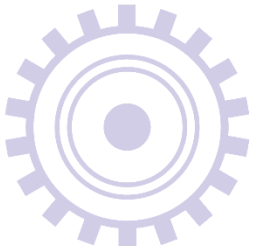
Comando de seleção múltipla (Escolha caso)

O VisuAlg implementa (com certas variações) o comando switch case das linguagens modernas como C#, Java, Javascript entre outras. A sintaxe é a seguinte:



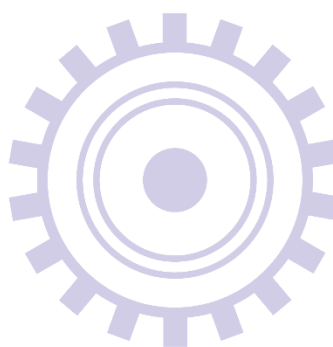
```
escolha <expressão-de-seleção>  
  caso <exp11>, <exp12>, ..., <exp1n>  
    <seqüência-de-comandos-1>  
  caso <exp21>, <exp22>, ..., <exp2n>  
    <seqüência-de-comandos-2>  
  ...  
  outrocaso  
    <seqüência-de-comandos-extra>  
fimescolha
```

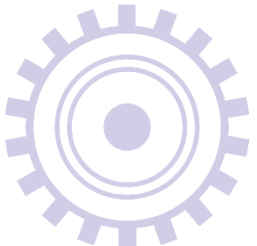




```
1 Algoritmo "Meses"
2 // Autor : Flávio Mota
3 var
4   mes : inteiro
5 inicio
6   escreva("Entre com um valor de 01 a 12:")
7   leia(mes)
8   escolha mes
9   caso 1
10      escreval("Janeiro")
11   caso 2
12      escreval("Fevereiro")
13   caso 3
14      escreval("Março")
15   caso 4
16      escreval("Abril")
17   caso 5
18      escreval("Maio")
19   caso 6
20      escreval("Junho")
21   // aqui, os outros meses .....
22   outrocaso
23      escreval("Mês inválido.")
24   fimescolha
25 fimalgoritmo
```

Nesse exemplo, o valor mostrado pelo algoritmo depende do valor digitado para a variável mês: caso 1 o sistema mostra “janeiro”, caso 2 o sistema mostra fevereiro e assim por diante.





```
algoritmo "SistemaDeNotas"
```

```
var  
nota:real
```

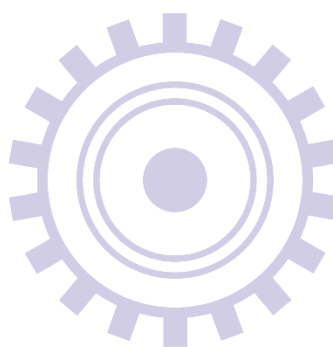
```
inicio
```

```
    escreva("Digite a nota do aluno (0 a 100): ")  
    leia(nota)
```

```
    escolha(nota)
```

```
        caso >= 90  
            escreva("Conceito: A")  
        caso >= 80  
            escreva("Conceito: B")  
        caso >= 70  
            escreva("Conceito: C")  
        caso >= 60  
            escreva("Conceito: D")  
        caso < 60  
            escreva("Conceito: F")  
    outrocaso  
        escreva("Nota inválida!")
```


```
    fimescolha  
fimalgoritmo
```

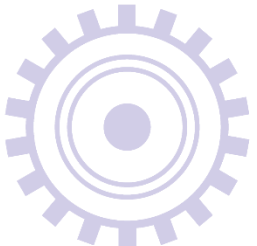


Este algoritmo é um sistema de notas que recebe a nota de um aluno (um número entre 0 e 100) e exibe o conceito correspondente de acordo com a seguinte escala:

- A: 90 a 100
- B: 80 a 89
- C: 70 a 79
- D: 60 a 69
- F: abaixo de 60

Se a nota estiver fora do intervalo de 0 a 100, ele exibe a mensagem "Nota inválida!".





```
algoritmo "CalculadoraSimples"
```

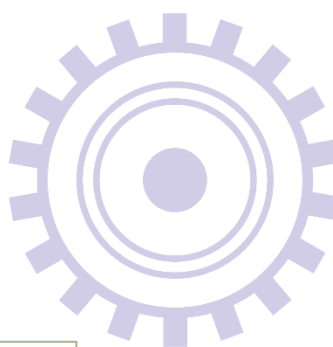
```
var
```

```
numero1,numero2,resultado:real  
operacao:caractere
```

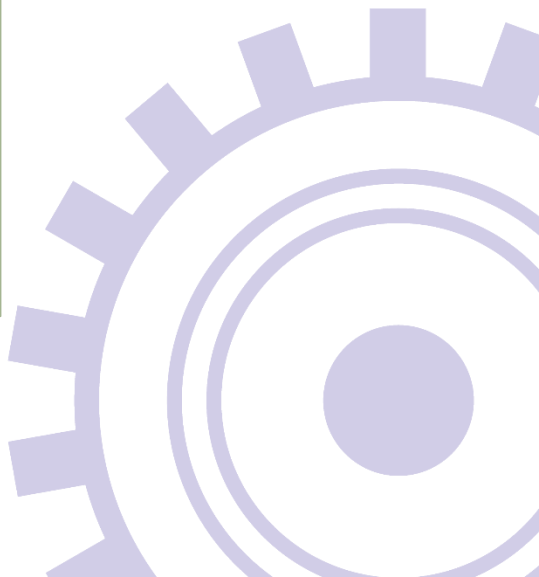
```
inicio
```

```
    escreva("Digite o primeiro número: ")  
    leia(numero1)  
    escreva("Digite o segundo número: ")  
    leia(numero2)  
    escreva("Escolha a operação: (+, -, *, /): ")  
    leia(operacao)
```

```
    escolha (operacao)  
    caso "+"  
        resultado <- numero1 + numero2  
        escreva("Resultado: ", resultado)  
    caso "-"  
        resultado <- numero1 - numero2  
        escreva("Resultado: ", resultado)  
    caso "*"  
        resultado <- numero1 * numero2  
        escreva("Resultado: ", resultado)  
    caso "/"  
        se numero2 != 0 entao  
            resultado <- numero1 / numero2  
            escreva("Resultado: ", resultado)  
        senao  
            escreva("Erro: Divisão por zero!")  
        fimse  
    outrocaso  
        escreva("Operação inválida!")  
    fimescolha  
fimalgoritmo
```



O algoritmo "CalculadoraSimples" é um programa que realiza operações matemáticas básicas (adição, subtração, multiplicação e divisão) entre dois números inseridos pelo usuário. Ele utiliza a estrutura "escolha caso" para determinar qual operação será realizada com base na entrada do usuário.



RECODE

Institucional



/rederecode



/recoderede