

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
CEARÁ
Campus Tianguá



Estruturas de Controle

davidifce.ti@gmail.com



Dados X Informação

- Qual a diferença?

Existe uma tênue diferença entre dado e informação.

Por exemplo:

data 21 de setembro é apenas um dado.

ao associarmos a essa data um valor como o Dia da Árvore estamos gerando uma informação.

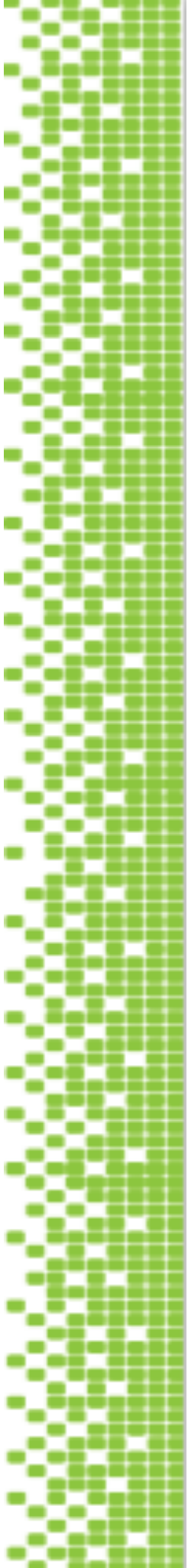


Tipos de Dados

- Destacamos 4 tipos primitivos de dados:
- **Inteiro** : toda e qualquer informação numérica que pertença ao conjunto dos números inteiros relativos (negativa, nula ou positiva).
 - ex.:
 - A escada possui 8 degraus.
 - Meu vizinho tem 15 anos.
 - A quantidade de laranjas compradas foram 10.



Tipos de Dados

- Destacamos 4 tipos primitivos de dados:
 - **Real** : toda e qualquer informação numérica que pertença ao conjunto dos números reais (negativa, nula ou positiva).
 - ex.:
 - Ela tem 1,73 metro de altura.
 - No momento estou pesando 82,5 kg.
 - A média do aluno foi 8,5.
- 

Tipos de Dados

- Destacamos 4 tipos primitivos de dados:
 - **Caracter** : toda e qualquer informação composta de um conjunto de caracteres alfanuméricos: números (0...9), alfabéticos (A...Z, a...z) e especiais(ex.: #, \$, %, & !, @)
 - ex.:
 - Constava na prova: “Use somente caneta!”.
 - No parque tinha placas: “Não pise na grama”.
 - Seu endereço é: “rua Alfa nº 1123”.



Tipos de Dados

- Destacamos 4 tipos primitivos de dados:
- **Lógico** : toda e qualquer informação que pode assumir apenas duas situações(biestável).
 - A porta pode estar aberta ou fechada.
 - A lâmpada pode estar acesa ou apagada.

Lógica e Operadores

- Operadores de Comparação
- Conhecidos como operadores relacionais.
- Permitem que relacionemos dois valores, duas variáveis ou uma variável com um valor.

Operador	Descrição
<	Menor que
>	Maior que
<=	Menor ou igual a
>=	Maior ou igual a
==	Igual a
!=	diferente

Lógica e Operadores

- Operadores Lógicos
- são empregados na construção de expressões lógicas.

Operador	Descrição
E	Conector E
OU	Conector OU
NAO	Operador de Negação

Lógica e Operadores

- Precedência de Operadores Lógicos

Precedência	Operador
1	Negação
2	e
3	ou

Lógica e Operadores

- Precedência de Operadores

Precedência	Operador
1	Parênteses mais internos
2	Operadores aritméticos
3	Operadores relacionais
4	Operadores Lógicos

Estrutura de Seleção

- Permite a escolha de um grupo de ações(bloco) a serem executadas quando determinadas condições, representadas por expressões lógicas ou relacionais, são ou não satisfeitas.
- Seleção Simples
 - Quando precisamos testar uma certa condição antes de executar uma ação, usamos uma seleção simples.

Estrutura de Seleção

É uma expressão lógica!
(Verdadeira / Falsa)

- Seleção Simples:

se <condição> **então**

| *C; //Comando único*

fimse;

Estrutura de Seleção

- Seleção Simples:

se <condição> **então**

{

C1; *//Comando 1*

C2; *//Comando 2*

C3; *//Comando 3*

Cn; *//Comando final do bloco*

}

fimse;

Para executar mais de um comando, utilizamos [início e fim]. Criamos um novo bloco!

Estrutura de Seleção

- Seleção Simples:

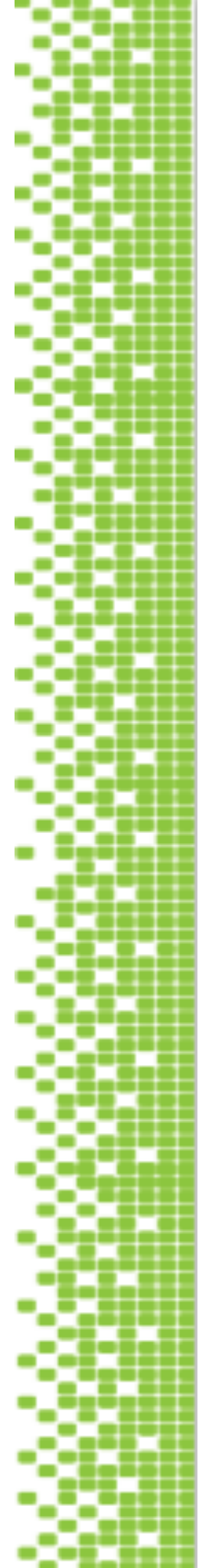
```
1 programa
2 {
3     funcao inicio()
4     {
5         real n1, n2, n3, n4, media
6         leia (n1, n2, n3, n4)
7         media = (n1 + n2 + n3 + n4)/4
8
9         se (media >= 7)
10             escreva("Aprovado")
11     }
12 }
13
14
```

Identifique as etapas de
Entrada - Processamento -
Saída!

Explique o
funcionamento do
algoritmo



Estrutura de Seleção

- Seleção Composta:
 - Quando tivermos situações em que duas alternativas dependem de uma mesma condição, uma sendo a condição verdadeira e a outra a falsa.
- 

Estrutura de Seleção

- Seleção Composta:

É uma expressão lógica!
(Verdadeira / Falsa)

se <condição> **então**

C; //Comando(s) Se Verdadeiro

senão

C; //Comando(S) Se Falso

fimse;

Estrutura de Seleção

- Seleção Composta:

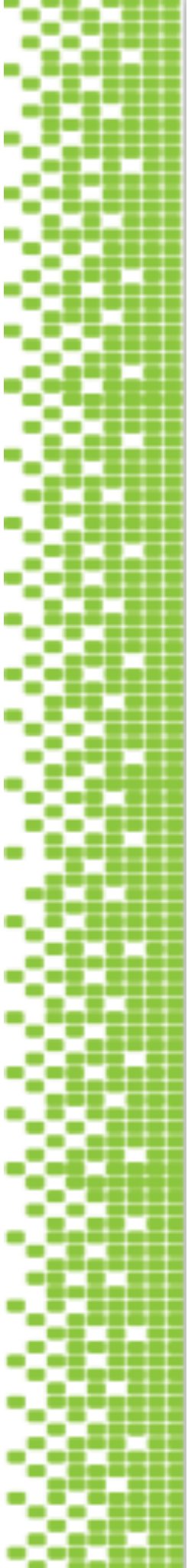
```
1 programa
2 {
3     funcao inicio()
4     {
5         real n1, n2, n3, n4, media
6         leia (n1, n2, n3, n4)
7
8         media = (n1 + n2 + n3 + n4)/4
9
10
11         se (media >= 7)
12             escreva("Aprovado")
13         senao
14             escreva("Reprovado")
15     }
16 }
```

Identifique as etapas de
Entrada - Processamento -
Saída!

Explique o
funcionamento do
algoritmo



Estrutura de Seleção

- Seleção Encadeada:
 - Quando, devido à necessidade de processamento, agruparmos várias seleções (simples e/ou compostas).
 - Problema da definição do tipo de triângulo!
- 

Estrutura de Seleção

- Seleção de Múltipla Escolha
 - Quando um conjunto de valores discretos precisa ser testado e ações diferentes são associadas a esses valores, estamos diante de uma seleção encadeada homogênea do tipo **se-senão-se**.
 - Podemos substituir as várias ocorrências de **se-senão-se**.
 - Analisa apenas um operador!

Estrutura de Seleção

- Seleção de Múltipla

escolha (X){

caso V1: C1

pare

caso V2: C2

pare

caso V3: C3

pare

caso Vn: Cn

pare

caso contrário: C

} **fimescolha**;

Operador a ser testado!

1. Casos em que o valor de X irá variar!
2. Para cada um, um comando deve ser executado!
3. Se nenhum caso ocorrer, será executado o “Caso contrário”!

Estrutura de Seleção

- EXERCÍCIOS

1. Quando se cria uma conta corrente no banco o usuário só pode sacar se seu saldo for maior ou igual ao valor desejado. Crie um algoritmo que o usuário informa o valor que deseja sacar e então o sistema indica se o saque poderá ser realizado emitindo uma mensagem (inicialmente o saldo do usuário é de 500) e o novo valor de saldo (saldo anterior - valor sacado). Caso a conta apresente saldo insuficiente (valor sacado > saldo) então uma mensagem de alerta será enviada. (considere como saldo inicial para a conta o valor 500) [Verifique em seu algoritmo os dois teste: 1 - para o saque no valor 250 e 2 - para o saque no valor 600]

Estrutura de Seleção

- EXERCÍCIOS

2. Você tem dois números, crie um algoritmo em javascript que sempre mostre esses números ordenados, primeiro o menor depois o maior. Ex.: caso o usuário indique os números 9 e 5 a resposta seria 5 e 9 (ordenado); caso o usuário indique os números 5 e 6 a resposta seria 5 e 6 (ordenados do menor para o maior).

3. Faça um algoritmo em javascript que receba do usuário 3 valores e informe qual é o menor e qual é o maior deles. Ex.: Se os números forem 4, 45, 88 a resposta seria: o maior 88 e o menor 4.

Estrutura de Seleção

- EXERCÍCIOS

4. Em diversas empresas é comum tratar os usuários/clientes de formas diferentes, de acordo com a faixa etária, no YouTube, por exemplo, tem restrições para menores de idade. Crie um algoritmo em javascript que recebe do usuário sua idade e mostra a sua faixa etária idoso (acima de 60 anos), adulto (de 18 a 60 anos) e menor de idade (de 0 a 17 anos).