

# Introdução à Programação de Computadores para Biologia

## Estruturas de controle

Aula 06

<https://ttdorres.github.io/introprog2024/>

# TIPOS DE DADOS EM PERL

## Variáveis

### 1. Escalares (\$):

```
my $variavel_escalar = 1;  
my $cidade = "Sao Paulo";  
my $sequencia = "ATCCTACTGTGCGTCAGGCTAAGCTA";
```

### 2. Arrays, vetores (@):

```
@genes = ("CG7856", "scpr-B", "CG4294", "Sgt", "CG42308");
```

### 3. Hashes, vetores associativos (%):

```
%genes = ("FBgn0033056" => "CG7856",  
          "FBgn0037888" => "scpr-B",  
          "FBgn0034742" => "CG424",  
          "FBgn0032640" => "Sgt")
```

# VETORES (HASHES & ARRAYS)

## PROBLEMAS

Como encontrar valores específicos em um hash? Como acessar e manipular elementos de arrays e hashes?

# ESTRUTURAS DE CONTROLE

## PROBLEMAS

Como encontrar valores específicos em um hash? Como acessar e manipular elementos de arrays e hashes?

Como como tomar decisões e fazer testes em um script?

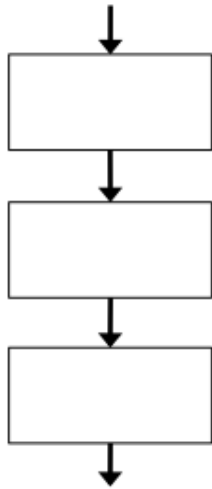
# ESTRUTURAS DE CONTROLE

## PROBLEMAS

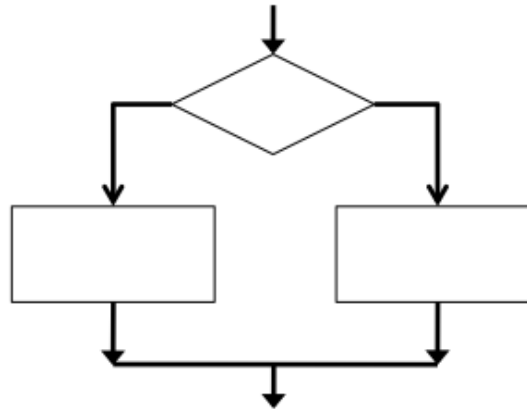
Como encontrar valores específicos em um hash? Como acessar e manipular elementos de arrays e hashes?

Como como tomar decisões e fazer testes em um script?

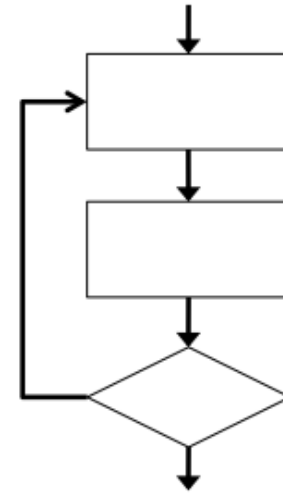
# ESTRUTURAS DE CONTROLE



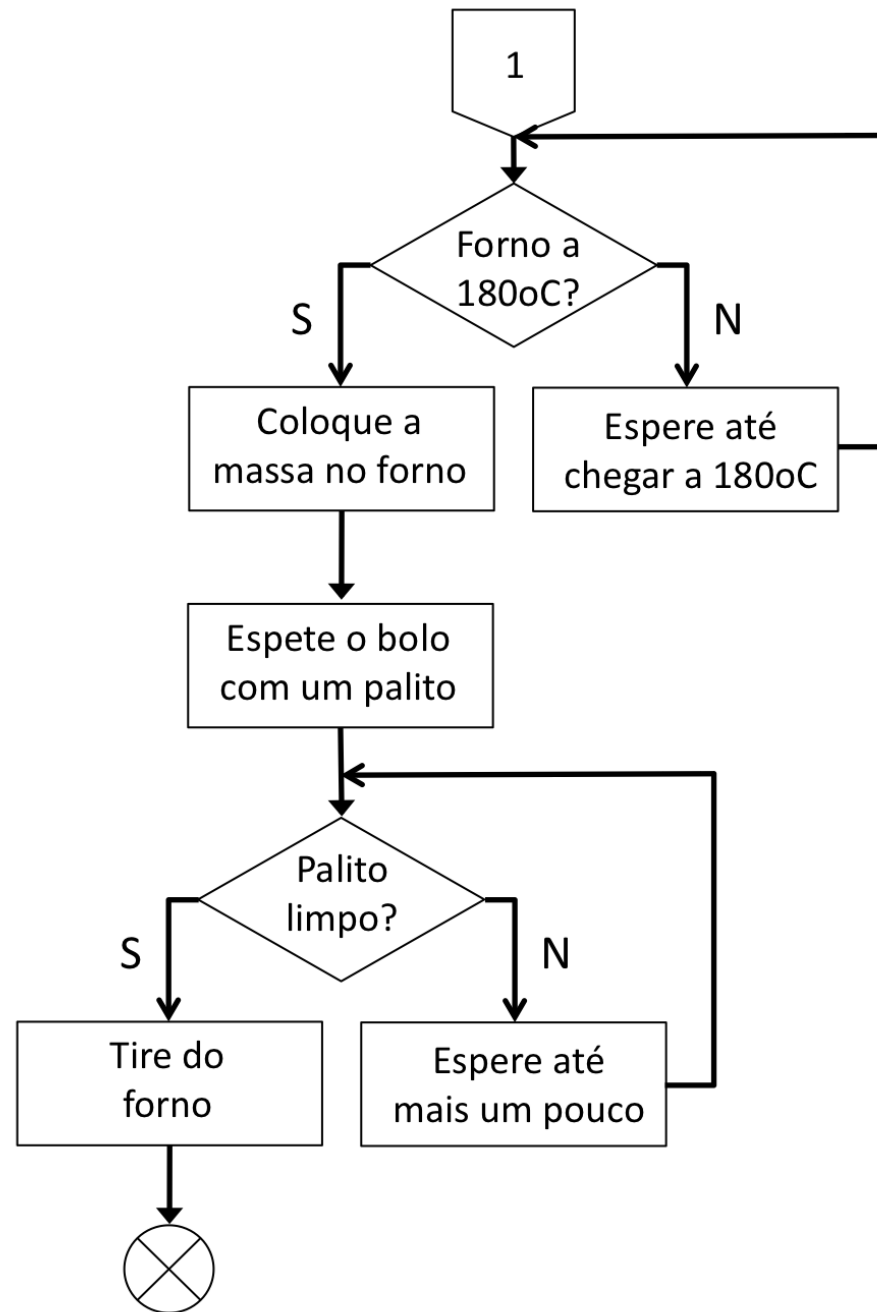
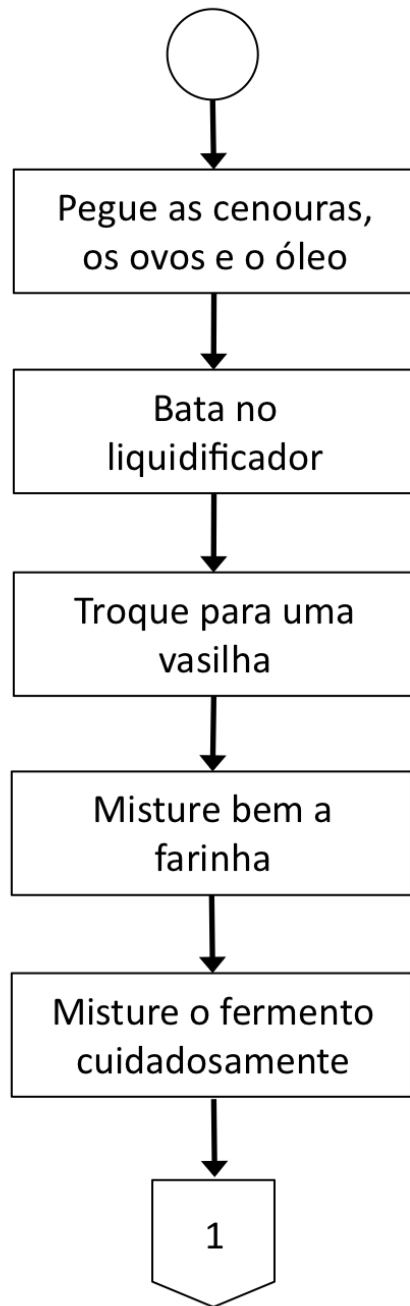
Estrutura de controle  
sequencial



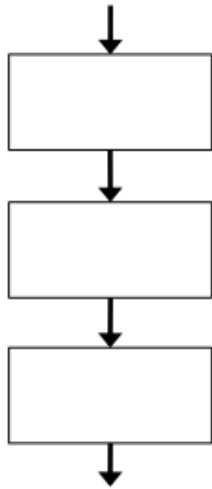
Estrutura de controle  
condicional



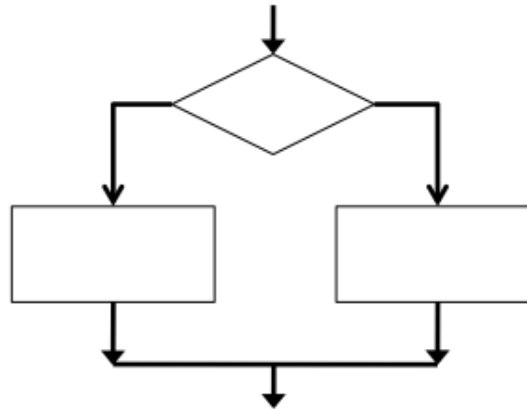
Estrutura de controle  
repetitiva



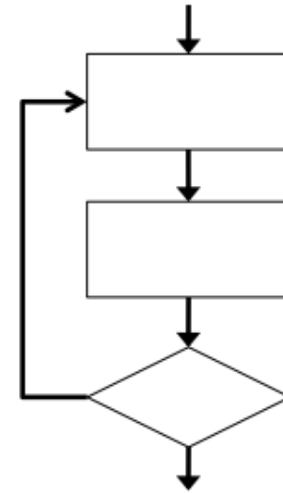
# ESTRUTURAS DE CONTROLE



Estrutura de controle  
sequencial



Estrutura de controle  
condicional



Estrutura de controle  
repetitiva



# ESTRUTURAS DE CONTROLE

- **if** (condição) comando
- **if** (condição) {comando1; comando2; comando3;}
- **if** (condição) comando **else** comando
- **if** (condição) comando **elsif** (condição) comando **else** comando
- **unless** (condição) comando
- **unless** (condição) comando **else** comando
- **unless** (condição) comando **elsif** (condição) comando **else** comando
- Ternary operator, "?":
  - método simplificado do **if** (condição) comando **else** comando
  - (condição) ? comando1 : comando2

# ESTRUTURAS DE CONTROLE

## Condicionais: o que são?

Avaliação de um argumento:  
VERDADEIRO ou FALSO

- Exemplos:
  - O valor "100" é maior que o valor "30". V ou F?
  - A palavra "ATG" é diferente da palavra "ATT". V ou F?
  - A palavra "100" é maior que a palavra "30". V ou F?

# ESTRUTURAS DE CONTROLE

## O que é verdadeiro/falso em Perl?

Não verdadeiro:

- zero
- string vazia
- lista vazia
- undefined (undef)

# ESTRUTURAS DE CONTROLE

## O que é verdadeiro/falso em Perl?

1. No Geany, File > New File.
2. File > Save as...
3. Gravar arquivo como [controle.pl](#)
4. Copiar **exemplo01** da página da disciplina.

# ESTRUTURAS DE CONTROLE

## O que é verdadeiro/falso em Perl?

Script: [controle.pl](#)

Função *defined()*, exemplo01:

```
#!/usr/bin/perl
# script para testar a função defined

# criando variáveis

$scalar1 = 1;
$scalar2 = 0;
$scalar3 = "";
$scalar4;

$temp = defined($scalar1);
print "Resultado do defined = $temp\n\n";

exit;
```

# ESTRUTURAS DE CONTROLE

## O que é verdadeiro/falso em Perl?

Script: [controle.pl](#)

Função *undef()*, exemplo02:

```
# exemplo03
# comandos para testar a função undef

print "Exemplo 02\n";

$temp1 = defined($scalar1); ## TRUE
print "Resultado do defined para variavel \">$temp1 = $temp1";

$temp2 = undef($scalar1); ## FALSE
print "Resultado do defined para variavel \">$temp2 = $temp2";

exit;
```

# ESTRUTURAS DE CONTROLE

## Condicionais: o que são?

Avaliação de um argumento:  
VERDADEIRO ou FALSO

- Exemplos:
  - O valor "100" é maior que o valor "30". V ou F?
  - A palavra "ATG" é diferente da palavra "ATT". V ou F?
  - A palavra "100" é maior que a palavra "30". V ou F?

# COMPARAÇÕES

## Operadores - Números e textos

1. Perl não tem os operadores:  $\neq$ ,  $\leq$ ,  $\geq$ ;
2. O sinal "=" é utilizado para atribuição;

```
#!/usr/bin/perl

# atribuicao

$scalar1 = 1;
$scalar2 = 0;

exit;
```

3. Operadores para números e texto são diferentes.



# COMPARAÇÕES

## Operadores - Números e textos

QUESTÃO	Números	"Strings"
\$a é igual a \$b?	\$a == \$b	\$a eq \$b
\$a é diferente de \$b?	\$a != \$b	\$a ne \$b
\$a é maior que \$b?	\$a > \$b	\$a gt \$b
\$a é maior ou igual a \$b?	\$a >= \$b	\$a ge \$b
\$a é menor que \$b?	\$a < \$b	\$a lt \$b
\$a é menor ou igual a \$b?	\$a <= \$b	\$a le \$b
\$a é diferente de \$b? Nesse caso, \$a é maior ou menor que \$b?	\$a <=> \$b	\$a cmp \$b

# ESTRUTURAS DE CONTROLE

## Operadores - Números e textos

Script: [controle.pl](#), exemplo03:

```
#!/usr/bin/perl

...

# testando operacoes matematicas
# comparacao 1
my $c = 5;
my $e = 5;
my $d = 20;

$temp = $c == $e;
print "A comparacao 1 eh $temp.\n";

exit;
```

# ESTRUTURAS DE CONTROLE

## Operadores - Números e textos

Script: [controle.pl](#), exemplo03:

```
#!/usr/bin/perl

...

# testando operacoes matematicas
# comparacao 1
my $c = 5;
my $e = 5;
my $d = 20;

$temp = $c == $e;
print "A comparacao 1 eh $temp.\n"; ## TRUE

exit;
```

# ESTRUTURAS DE CONTROLE

## Operadores - Números e textos

Script: [controle.pl](#), exemplo03:

```
#!/usr/bin/perl

...

# testando operacoes matematicas
# comparacao 2
my $c = 5;
my $e = 5;
my $d = 20;

$temp = $c == $d;
print "A comparacao 2 eh $temp.\n";

exit;
```

# ESTRUTURAS DE CONTROLE

## Operadores - Números e textos

Script: [controle.pl](#), exemplo02:

```
#!/usr/bin/perl

...

# testando operacoes matematicas
# comparacao 2
my $c = 5;
my $e = 5;
my $d = 20;

$temp = $c == $d;
print "A comparacao 2 eh $temp.\n"; ## FALSE

exit;
```

# ESTRUTURAS DE CONTROLE

## Operadores - Números e textos

Script: [controle.pl](#), exemplo03:

```
#!/usr/bin/perl

...

# testando operacoes matematicas
# comparacao 3
my $c = 5;
my $e = 5;
my $d = 20;

$temp = $c <= $d;
print "A comparacao 3 eh $temp.\n";

exit;
```

# ESTRUTURAS DE CONTROLE

## Operadores - Números e textos

Script: [controle.pl](#), exemplo03:

```
#!/usr/bin/perl

...

# testando operacoes matematicas
# comparacao 3
my $c = 5;
my $e = 5;
my $d = 20;

$temp = $c <= $d;
print "A comparacao 3 eh $temp.\n"; ## TRUE

exit;
```

# ESTRUTURAS DE CONTROLE

## Operadores - Números e textos

Script: [controle.pl](#), exemplo03:

```
#!/usr/bin/perl

...

# testando operacoes matematicas
# comparacao 4
my $c = 5;
my $e = 5;
my $d = 20;

$temp = $c <=> $d;
print "A comparacao 4 eh $temp.\n"; ## -1, 0, 1

exit;
```



# ESTRUTURAS DE CONTROLE

## Operadores - Números e textos

Script: [controle.pl](#), exemplo03:

```
#!/usr/bin/perl

...

# testando operacoes matematicas
# comparacao 5 - contexto de "string"
my $c = 5;
my $e = 5;
my $d = 20;

$temp = $c eq $d;
print "A comparacao 5 eh $temp.\n";

exit;
```

# ESTRUTURAS DE CONTROLE

## Operadores - Números e textos

Script: [controle.pl](#), exemplo03:

```
#!/usr/bin/perl

...

# testando operacoes matematicas
# comparacao 5 - contexto de "string"
my $c = 5;
my $e = 5;
my $d = 20;

$temp = $c eq $d;
print "A comparacao 5 eh $temp.\n"; ## FALSE

exit;
```

# ESTRUTURAS DE CONTROLE

## Operadores - Números e textos

Script: [controle.pl](#), exemplo03:

```
#!/usr/bin/perl

...

# testando operacoes matematicas
# comparacao 6 - contexto de "string"
my $c = 5;
my $e = 5;
my $d = 20;

$temp = $c gt $d;
print "A comparacao 6 eh $temp.\n";

exit;
```

# ESTRUTURAS DE CONTROLE

## Operadores - Números e textos

Script: [controle.pl](#), exemplo03:

```
#!/usr/bin/perl

...

# testando operacoes matematicas
# comparacao 6 - contexto de "string"
my $c = 5;
my $e = 5;
my $d = 20;

$temp = $c gt $d;
print "A comparacao 6 eh $temp.\n"; ## TRUE!!!

exit;
```

# ESTRUTURAS DE CONTROLE

## Operadores - Números e textos

Script: [controle.pl](#), exemplo03:

```
#!/usr/bin/perl

...

# testando operacoes matematicas
# comparacao 7 - contexto de "string"
my $c = 5;
my $e = 5;
my $d = 20;

$temp = $c cmp $d;
print "A comparacao 7 eh $temp.\n"; ## -1, 0, 1

exit;
```

# ESTRUTURAS DE CONTROLE

## Operadores - Números e textos

Script: [controle.pl](#), exemplo03:

```
#!/usr/bin/perl

...

# testando operacoes matematicas
# comparacao 8 - contexto de "string"
# ordem alfabética

$geneA = "FBgn2";
$geneB = "FBgn100";

$temp = $geneA cmp $geneB;
print "A comparacao 8 eh $temp.\n";

exit;
```

# ESTRUTURAS DE CONTROLE

## Operadores - Números e textos

Script: [controle.pl](#), exemplo03:

```
#!/usr/bin/perl

...

# testando operacoes matematicas
# comparacao 9 - contexto de "string"
# ordem alfabética

$geneA = "FBgn002";
$geneB = "FBgn100";

$temp = $geneA cmp $geneB;
print "A comparacao 9 eh $temp.\n";

exit;
```

# ESTRUTURAS DE CONTROLE

## Condicionais

### Problema:

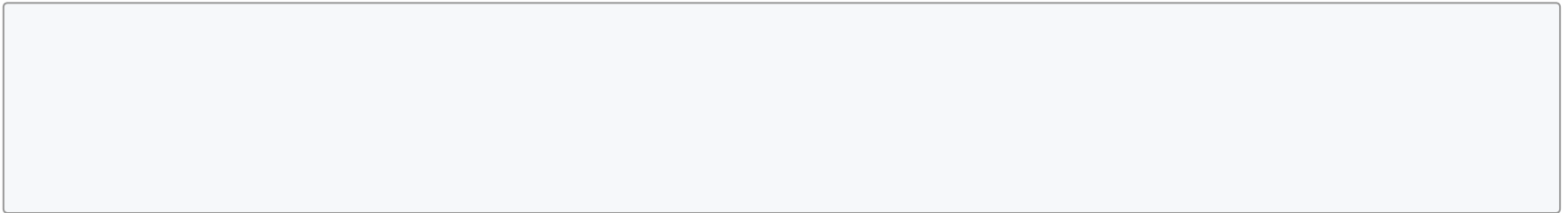
Dizer se um aluno foi aprovado somente SE a nota final dele for maior que 5.



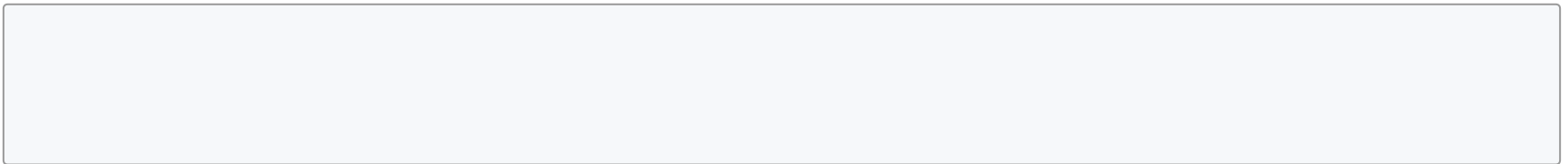
# ESTRUTURAS DE CONTROLE

## Condicionais

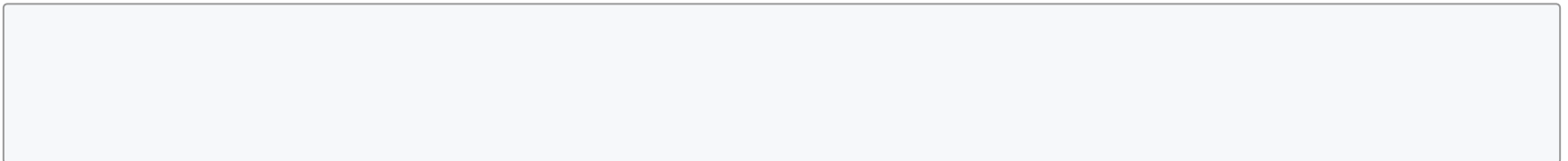
### 1. *if*

A large, empty rectangular box with a light blue background and a thin black border, intended for a code example of an if statement.

### 2. *else e elsif*

A large, empty rectangular box with a light blue background and a thin black border, intended for a code example of an else or elsif statement.

### 3. *boolean*

A large, empty rectangular box with a light blue background and a thin black border, intended for a code example of a boolean statement.

# ESTRUTURAS DE CONTROLE

## Condicionais

### 1. *if*

```
if (expressao) {  
    bloco de comandos  
}
```

### 2. *else e elsif*

### 3. *boolean*

# CONDICIONAIS

if, boolean, else

notas.pl

```
#!/usr/bin/perl

($E, $M, $H, $nota_final);

($E, $M, $H) = @ARGV;

$nota_final = ((2*$E)+(3*$M)+(5*$H))/10;

if ($nota_final >= 5) {
    print "Aluno aprovado\n";
} else {
    print "Aluno reprovado\n";
}

exit;
```

# ESTRUTURAS DE CONTROLE

## Condicionais

### Problema:

Calcular o comprimento de uma sequência somente SE a qualidade média dela for maior que 30.

# CONDICIONAIS

## if, else, boolean

1. No Geany, File > New File.
2. File > Save as...
3. Gravar arquivo como [condicionais.pl](#)
4. Copiar **exemplo04** da página da disciplina e colar no script [condicionais.pl](#).

# CONDICIONAIS

if

**condicionais.pl**, exemplo 04

```
# exemplo04
#! /usr/bin/perl
# script para testar estruturas de controle

$seq = "ATGCGCGTAGCTCGTAGCTAGCTAGCTAGCT";
$avg_qual = 40;

if ($avg_qual > 30) {
    $seq_length = length($seq);
}

print "O tamanho da sequencia eh $seq_length.\n";

exit;
```

# CONDICIONAIS

if

**condicionais.pl**, exemplo 04

```
# exemplo04
#! /usr/bin/perl
# script para testar estruturas de controle

$seq = "ATGCGCGTAGCTCGTAGCTAGCTAGCTAGCT";
$avg_qual = 40;

if ($avg_qual > 40) {

    $seq_length = length($seq);

}

print "O tamanho da sequencia eh $seq_length.\n";

exit;
```

# CONDICIONAIS

if

**condicionais.pl**, exemplo 04

```
# exemplo04
#! /usr/bin/perl
# script para testar estruturas de controle

$seq = "ATGCGCGTAGCTCGTAGCTAGCTAGCTAGCT";
$avg_qual = 40;

if ($avg_qual >= 40) {
    $seq_length = length($seq);
}

print "O tamanho da sequencia eh $seq_length.\n";

exit;
```



# CONDICIONAIS

if

**condicionais.pl**, exemplo 05

```
# exemplo05
# comando die()

$nota_do_aluno = 4.5;
$nota_de_corte = 5.0;

if ($nota_do_aluno < $nota_de_corte) {
    die "Aluno reprovado\n";
}

exit;
```

# CONDICIONAIS

if

**condicionais.pl**, exemplo 05

```
# exemplo05
# comando die()

$nota_do_aluno = 4.5;
$nota_de_corte = 5.0;

if ($nota_do_aluno < $nota_de_corte) {
    die "Aluno reprovado\n";
}

print "Imprimindo qualquer coisa depois de die()\n";

exit;
```

# CONDICIONAIS

if

**condicionais.pl**, exemplo 05

```
# exemplo05
# comando die()

$nota_do_aluno = 5.5;
$nota_de_corte = 5.0;

if ($nota_do_aluno < $nota_de_corte) {
    die "Aluno reprovado\n";
}

print "Imprimindo qualquer coisa depois de die()\n";

exit;
```

# CONDICIONAIS

**if**

**condicionais.pl**, exemplo 06

```
# exemplo06  
# comando die()  
  
@array_vazio = ();  
  
$temp = pop(@array_vazio);  
  
if (! $temp) {  
    die "pop() nao deve ser usado em um array vazio\n";  
}  
  
exit;
```

# CONDICIONAIS

## Comando *unless*

[condicionais.pl](#), exemplo 06

```
# exemplo06
# comando die()

@array_vazio = ();

$temp = pop(@array_vazio);

unless ($temp) {
    die "pop() nao deve ser usado em um array vazio\n";
}

exit;
```

# CONDICIONAIS

## Comando *unless*

[condicionais.pl](#), exemplo 06

```
# exemplo06
# comando die()

@array_vazio = (1,2,3);

$temp = pop(@array_vazio);

unless ($temp) {
    die "pop() nao deve ser usado em um array vazio\n";
}

exit;
```

# CONDICIONAIS

**if**

## **Problema:**

Determinar se um elemento está em um hash como chave e usar a informação (o valor) associada a essa chave.

# CONDICIONAIS

if

**condicionais.pl**, exemplo 07

```
# exemplo07
# verificando hashes

%presenca = ("Cascao"    => 85,
             "Cebolinha" => 65,
             "Magali"    => 95,
             "Monica"    => 80);

$estudante = "Cebolinha";

if ($presenca{$estudante}) {
    print "$estudante participou de $presenca{$estudante}% das aulas."
}

exit;
```



# CONDICIONAIS

lembrar de "inativar" o comando die()!

**condicionais.pl**, exemplo 07

```
# exemplo07
# verificando hashes

%presenca = ("Cascao"    => 85,
             "Cebolinha" => 65,
             "Magali"    => 95,
             "Monica"    => 80);

$estudante = "Cebolinha";

if ($presenca{$estudante}) {
    print "$estudante participou de $presenca{$estudante}% das aulas.
}

exit;
```

# CONDICIONAIS

lembrar de "inativar" o comando die()!

[condicionais.pl](#), exemplo 07

```
# exemplo07
# verificando hashes

%presenca = ("Cascao"    => 85,
             "Cebolinha" => 65,
             "Magali"    => 95,
             "Monica"    => 80);

$estudante = "Cebolinha";

if ($presenca{$estudante} < 75) {
    print "$estudante reprovado por faltas. \n";
}

exit;
```

# CONDICIONAIS

**if**

## **Problema:**

Determinar se a presença é maior que 75% e imprimir "aprovado", caso contrário, imprimir "reprovado" (realizar ações diferentes para valores verdadeiros e falsos).

# ESTRUTURAS DE CONTROLE

## Condicionais

### 1. *if*

```
if (expressao) {  
    bloco de comandos  
}
```

### 2. *else e elsif*

```
if (expressao) {  
    bloco de comandos  
} else {  
    bloco de comandos  
}
```

### 3. *boolean*

# CONDICIONAIS

*if*(condição) { comandos } *else* { comandos }

**condicionais.pl**, exemplo 07

```
# exemplo07
# verificando hashes

%presenca = ("Cascao"    => 85,
             "Cebolinha" => 65,
             "Magali"    => 95,
             "Monica"    => 80);

$estudante = "Cebolinha";

if ($presenca{$estudante} < 75) {
    print "$estudante reprovado por faltas.\n";
} else {
    print "$estudante aprovado.\n";
}

exit;
```

# CONDICIONAIS

*if*(condição) { comandos } *else* { comandos }

[condicionais.pl](#), exemplo 07

```
# exemplo07
# verificando hashes

%presenca = ("Cascao"    => 85,
             "Cebolinha" => 65,
             "Magali"    => 95,
             "Monica"    => 80);

$estudante = "Cascao";

if ($presenca{$estudante} < 75) {
    print "$estudante reprovado por faltas.\n";
} else {
    print "$estudante aprovado.\n";
}

exit;
```

# CONDICIONAIS

## Operator conditional ?

### condicionais.pl, exemplo 07

```
# exemplo07
# (condição) ? resultado1 : resultado2

%presenca = ("Cascao"      => 85,
             "Cebolinha"   => 65,
             "Magali"      => 95,
             "Monica"      => 80);

$estudante = "Cebolinha";

$situacao = ($presenca{$estudante} < 75) ? "reprovado" : "aprovado";

print "$estudante foi $situacao.\n";

exit;
```

# CONDICIONAIS

## Operator conditional ?

### condicionais.pl, exemplo 07

```
# exemplo07
# (condição) ? resultado1 : resultado2

%presenca = ("Cascao"    => 85,
             "Cebolinha" => 65,
             "Magali"    => 95,
             "Monica"    => 80);

$estudante = "Monica";

$situacao = ($presenca{$estudante} < 75) ? "reprovado" : "aprovado";

print "$estudante foi $situacao.\n";

exit;
```



# CONDICIONAIS

## if, elsif, else

1. No Geany, File > New File.
2. File > Save as...
3. Gravar arquivo como [notas.pl](#)
4. Criar um script para testar se a nota é maior ou menor que a nota de corte e imprimir "Aprovado" ou "Reprovado".

# CONDICIONAIS

## if e else

### notas.pl

```
#!/usr/bin/perl
# outro exemplo com if() e else()

print "Notas:\n";

$nota_do_aluno = 10.0;
$nota_de_corte = 5.0;

if (.....) {
    comando;
} .....

exit;
```

# CONDICIONAIS

## if e else

### notas.pl

```
#!/usr/bin/perl
# outro exemplo com if() e else()

print "Notas:\n";

$nota_do_aluno = 10.0;
$nota_de_corte = 5.0;

if ($nota_do_aluno < $nota_de_corte) {
    print "Aluno reprovado\n";
} else {
    print "Aluno aprovado\n";
}

exit;
```

# CONDICIONAIS

**if, else e elsif**

**E se houver mais que duas alternativas?**

**Exemplo:**

Conceitos na Pós-graduação

nota < 5.0, reprovado

nota <= 7.0, conceito C

nota <= 8.5, conceito B

nota > 8.5, conceito A

# CONDICIONAIS

## else e elsif

[notas.pl](#), exemplo 08

```
# nota < 5,      reprovado
# nota <= 7,     C
# nota <= 8.5,  B
# nota > 8.5,   A
# varias condicoes diferentes

if ($nota_do_aluno < $nota_de_corte) {
    print "Aluno reprovado!\n";
} else {
    if ($nota_do_aluno <= 7) {
        print "C, regular!\n";
    }
}
```

```
if ($nota_do_aluno < $nota_de_corte) {  
    print "Aluno reprovado!\n";  
} else {  
    if ($nota_do_aluno <= 7) {  
        print "C, regular!\n";  
    } else {  
        if ($nota_do_aluno <= 8.5) {  
            print "B, bom!\n";  
        } else {  
            print "A, excelente!\n";  
        }  
    }  
}  
  
exit;
```

# else e elsif

## notas.pl, exemplo 08

```
if ($nota_do_aluno < $nota_de_corte) {  
    print "Aluno reprovado!\n";  
}  
elsif ($nota_do_aluno <= 7) {  
    print "C, regular!\n";  
}  
elsif ($nota_do_aluno <= 8.5) {  
    print "B, bom!\n";  
}  
else {  
    print "A, excelente!\n";  
}  
  
exit;
```

# CONDICIONAIS

**boolean**

**Exemplo:**

Determinar se um mesmo elemento está em dois hashes diferentes e usar a informação desses hashes.



# CONDICIONAIS

## boolean

1. No Geany, File > New File.
2. File > Save as...
3. Gravar arquivo como [boolean.pl](#)
4. Copiar **exemplo09** da página da disciplina e colar no script [boolean.pl](#).

# CONDICIONAIS

if

[boolean.pl](#), exemplo 09

```
# exemplo09
# comparando dois hashes

%coord_start = ("Gene1" => "100000",
                "Gene2" => "550000");
%coord_end   = ("Gene1" => "102000",
                "Gene2" => "556000");

$gene_alvo = "Gene1";

if ($coord_start{$gene_alvo}) {
    if ($coord_end{$gene_alvo}) {
        print "As coordenadas do $gene_alvo sao
              $coord_start{$gene_alvo}..$coord_end{$gene_alvo}. \n";
    }
}

exit;
```

# CONDICIONAIS

if

[boolean.pl](#), exemplo 09

```
# exemplo09
# comparando dois hashes

%coord_start = ("Gene1" => "100000",
                "Gene2" => "550000");
%coord_end   = ("Gene2" => "556000");

$gene_alvo = "Gene1";

if ($coord_start{$gene_alvo}) {
    if ($coord_end{$gene_alvo}) {
        print "As coordenadas do $gene_alvo sao
               $coord_start{$gene_alvo}..$coord_end{$gene_alvo}. \n";
    }
}

exit;
```

# ESTRUTURAS DE CONTROLE

## Condicionais

### 1. *if*

```
if (expressao) {  
    bloco de comandos  
}
```

### 2. *else e elsif*

```
if (expressao) {  
    bloco de comandos  
} else {  
    bloco de comandos  
}
```

### 3. *boolean*

```
(expressao) && (expressao)  
(expressao) || (expressao)
```

# CONDICIONAIS

## Operadores booleanos

A	B	<i>A and B</i>	<i>A or B</i>
TRUE	TRUE		
FALSE	FALSE		
TRUE	FALSE		
FALSE	TRUE		

# CONDICIONAIS

## Operadores booleanos

A	B	<i>A and B</i>	<i>A or B</i>
TRUE	TRUE	TRUE	
FALSE	FALSE		
TRUE	FALSE		
FALSE	TRUE		

# CONDICIONAIS

## Operadores booleanos

A	B	<i>A and B</i>	<i>A or B</i>
TRUE	TRUE	TRUE	TRUE
FALSE	FALSE		
TRUE	FALSE		
FALSE	TRUE		

# CONDICIONAIS

## Operadores booleanos

A	B	<i>A and B</i>	<i>A or B</i>
TRUE	TRUE	TRUE	TRUE
FALSE	FALSE	FALSE	
TRUE	FALSE		
FALSE	TRUE		



# CONDICIONAIS

## Operadores booleanos

A	B	<i>A and B</i>	<i>A or B</i>
TRUE	TRUE	TRUE	TRUE
FALSE	FALSE	FALSE	FALSE
TRUE	FALSE		
FALSE	TRUE		

# CONDICIONAIS

## Operadores booleanos

A	B	<i>A and B</i>	<i>A or B</i>
TRUE	TRUE	TRUE	TRUE
FALSE	FALSE	FALSE	FALSE
TRUE	FALSE	FALSE	TRUE
FALSE	TRUE		

# CONDICIONAIS

## Operadores booleanos

A	B	<i>A and B</i>	<i>A or B</i>
TRUE	TRUE	TRUE	TRUE
FALSE	FALSE	FALSE	FALSE
TRUE	FALSE	FALSE	TRUE
FALSE	TRUE	FALSE	TRUE

# CONDICIONAIS

## Operadores booleanos

### boolean.pl, exemplo 10

```
# exemplo10
# operadores booleanos &&, ||

%coord_start = ("Gene1" => "100000",
                "Gene2" => "550000");
%coord_end   = ("Gene1" => "102000",
                "Gene2" => "556000");

$gene_alvo = "Gene1";

if (($coord_start{$gene_alvo}) && ($coord_end{$gene_alvo})) {
    print "As coordenadas do $gene_alvo sao
    $coord_start{$gene_alvo}..$coord_end{$gene_alvo}. \n";
}

exit;
```

# CONDICIONAIS

## Operadores booleanos

### boolean.pl, exemplo 10

```
# exemplo10
# operadores booleanos &&, ||

%coord_start = ("Gene1" => "100000",
                "Gene2" => "550000");
%coord_end    = ("Gene2" => "556000");

$gene_alvo = "Gene1";

if (($coord_start{$gene_alvo}) && ($coord_end{$gene_alvo})) {
    print "As coordenadas do $gene_alvo sao
    $coord_start{$gene_alvo}..$coord_end{$gene_alvo}. \n";
}

exit;
```

# CONDICIONAIS

## Operadores booleanos

### boolean.pl, exemplo 10

```
# exemplo10
# operadores booleanos &&, ||

%coord_start = ("Gene1" => "100000",
                "Gene2" => "550000");
%coord_end    = ("Gene2" => "556000");

$gene_alvo = "Gene1";

if (($coord_start{$gene_alvo}) || ($coord_end{$gene_alvo})) {
    print "As coordenadas do $gene_alvo sao
    $coord_start{$gene_alvo}..$coord_end{$gene_alvo}. \n";
}

exit;
```

# ESTRUTURAS DE CONTROLE

## Condicionais

### 1. *if*

```
if (expressao) {  
    bloco de comandos  
}
```

### 2. *else e elsif*

```
if (expressao) {  
    bloco de comandos  
} else {  
    bloco de comandos  
}
```

### 3. *boolean*

```
(expressao) && (expressao)  
(expressao) || (expressao)
```

