

# Introdução à Programação de Computadores para Biologia

## Tipos de dados

Aula 04b

<https://ttdorres.github.io/introprog2021/>

# PERL

## Primeiro Programa

Arquivo [hello.pl](#):

```
#!/usr/bin/perl  
  
# Uhu! Meu primeiro script!  
print "Hello world!\n";
```

No terminal:

```
$ perl hello.pl Sao Paulo  
Hello Sao Paulo!  
$
```

# TIPOS DE DADOS

## Variáveis

### 1. Escalares (\$):

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

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

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

# VARIÁVEIS ESCALARES

## Variáveis

1. Nomes precedidos de "\$":

```
my $cidade = "Sao Paulo";      #correto  
my $ cidade = "Sao Paulo";     #incorreto
```

2.

# VARIÁVEIS ESCALARES

## Variáveis

1. Nomes precedidos de "\$":

```
my $cidade = "Sao Paulo";      #correto
my $ cidade = "Sao Paulo";     #incorreto
```

2. Nomes podem conter uma ou mais letras "A-Z" ou "a-z" incluindo "\_" e depois dela(s) números:

```
my $v = 1;          #correto
my $var = 1;         #correto
my $var1 = 1;        #correto
my $var2 = 2;        #correto
my $var_1 = 1;       #correto
my $var 1 = 1;       #incorreto
my $1var = 1;        #incorreto
my $2var = 2;        #incorreto
my $variavel_escalar_criada_para_armazenar_um_numero_real = 1;
```



# VARIÁVEIS ESCALARES

## Variáveis

3. Variáveis pré-definidas:

```
$_  
$1  
$2  
$^  
$/  
$\n  
$,
```

4.

# VARIÁVEIS ESCALARES

## Variáveis

### 3. Variáveis pré-definidas:

```
$_  
$1  
$2  
$^  
$/  
$\  
$,
```

### 4. Perl diferencia maiúsculas e minúsculas.

```
my $VAR = 1;  
my $VAr = 2;  
my $Var = 3;  
my $var = 4;
```



# VARIÁVEIS ESCALARES

## Atribuição

Operador de atribuição, "="

```
$var = 1; # atribuindo 1 a variavel $var
```

# VARIÁVEIS ESCALARES

## Atribuição

Operador de atribuição, "="

```
$var = 1; #atribuindo 1 a variavel $var  
1 = $var;
```

# VARIÁVEIS ESCALARES

## Atribuição

Operador de atribuição, "="

```
$var = 1;      #atribuindo 1 a variavel $var
```

```
1 = $var;      #incorreto
```

# VARIÁVEIS ESCALARES

## Atribuição

Operador de atribuição, "="

```
$var = 1;      #atribuindo 1 a variavel $var
```

```
1 = $var;      #incorreto
```

```
$greetings = "Hello world!";
```

# VARIÁVEIS ESCALARES

## Atribuição

Operador de atribuição, "="

```
$var = 1;      #atribuindo 1 a variavel $var
```

```
1 = $var;      #incorreto
```

```
$greetings = "Hello world!";
```

```
$location = <STDIN>;
```

# VARIÁVEIS ESCALARES

## hello.pl

1. Abrir o Geany, arquivo [hello.pl](#).
2. Copiar #exemplo01 da página da disciplina.
3. No terminal:

```
perl ~/home/aluno/hello.pl
```

# VARIÁVEIS ESCALARES

## Impressão

Comando *print*

```
#!/usr/bin/perl  
  
# Meu primeiro script  
print "Hello world!\n";
```

# VARIÁVEIS ESCALARES

## Impressão

Comando *print*

```
#!/usr/bin/perl

# Meu primeiro script
print "Hello world!\n";

# Imprimir diretamente o conteúdo da variável
$greetings = "Hello world!";
print $greetings;
```

No terminal:

```
perl ~/home/aluno/hello.pl
```



# VARIÁVEIS ESCALARES

## Impressão

Comando *print*

```
#!/usr/bin/perl

# Meu primeiro script
print "Hello world!\n";

#exemplo02 – COPIAR DA PAGINA
# Imprimir o conteudo da variavel como parte de uma
# sentença maior
```

# VARIÁVEIS ESCALARES

## Impressão

Comando *print*

```
#!/usr/bin/perl

# Meu primeiro script
print "Hello world!\n";

#exemplo02
# Imprimir o conteudo da variavel como parte de uma
# sentença maior

print "Eu não suporto mais esse exemplo do $greetings\n";

exit;
```

# VARIÁVEIS ESCALARES

## Na prática

1. No Geany, File > New File.
2. Copiar `#exemplo03` da página da disciplina.
3. File > Save as...
4. Gravar arquivo como [interpolacao.pl](#)

# VARIÁVEIS ESCALARES

## Na prática

Script: [interpolacao.pl](#)

```
#!/usr/bin/perl  
# script para testar interpolacao  
  
# declarando minha variavel constante  
my $greetings = "Hello world!";  
  
# imprimindo  
print "$greetings, mais uma vez!\n";  
  
exit;
```

# VARIÁVEIS ESCALARES

## Na prática

Script: [interpolacao.pl](#)

```
#!/usr/bin/perl  
# script para testar interpolacao  
  
# declarando minha variavel constante  
my $greetings = "Hello world!";  
  
# imprimindo  
print "$greetings, mais uma vez!\n";  
print '$greetings, mais uma vez!\n';  
  
exit;
```

# VARIÁVEIS ESCALARES

## Na prática

Script: [interpolacao.pl](#)

Copiar exemplo04 da página da disciplina

```
# exemplo04

#!/usr/bin/perl
# script para testar interpolacao

# declarando minha variavel constante

my $greetings = "Hello world!";

# imprimindo
print 'O nome da nossa variavel eh $greetings';
print " e o conteudo dela eh $greetings.\n";

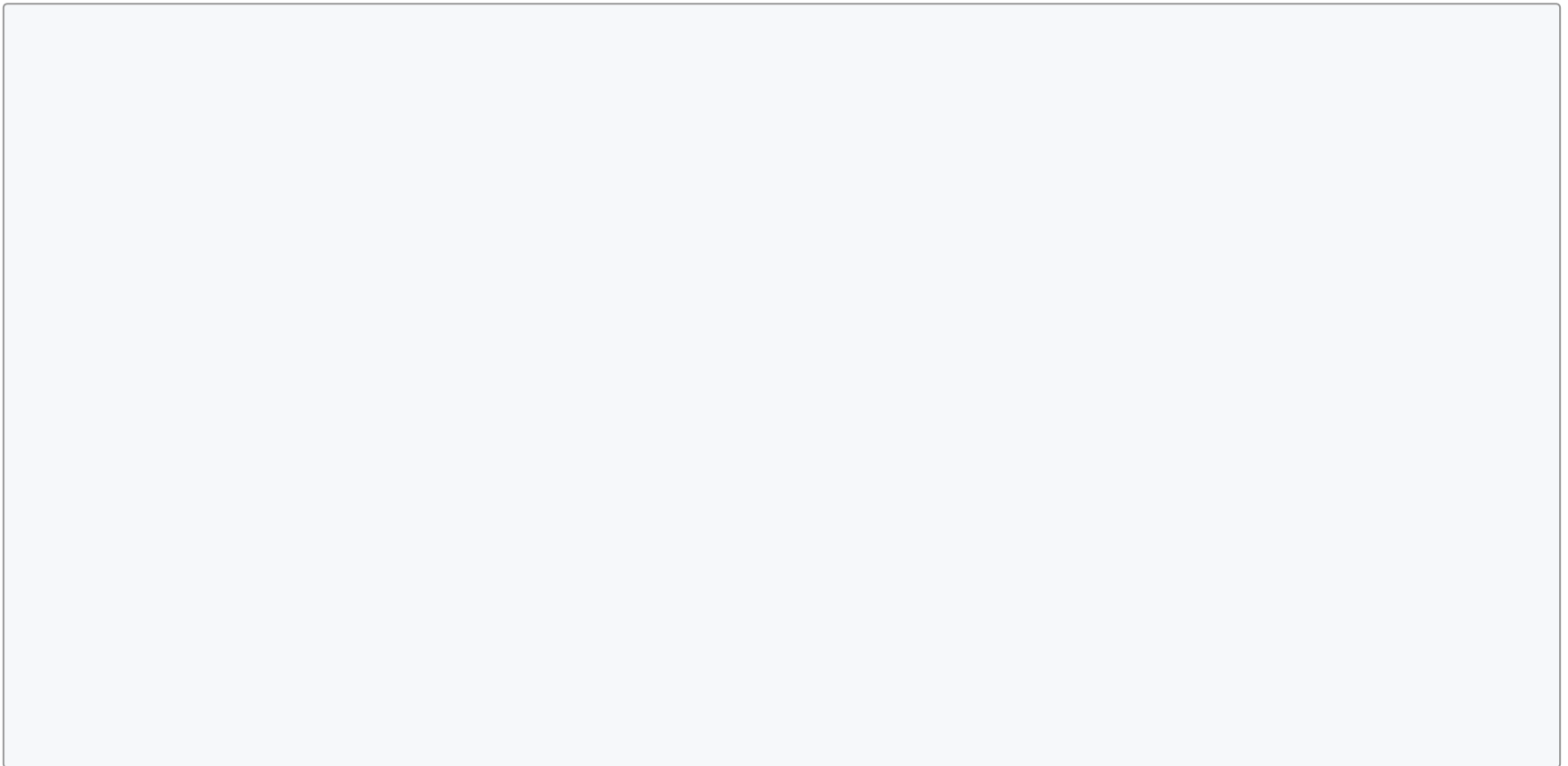
exit;
```

# VARIÁVEIS ESCALARES

## Na prática

Script: [interpolacao.pl](#)

E como imprimir "Hello world!" (**COM as aspas**)?



# VARIÁVEIS ESCALARES

## Na prática

Script: [interpolacao.pl](#)

E como imprimir "Hello world!" (COM as aspas)?

```
# exemplo04

#! /usr/bin/perl
# script para testar interpolacao

# declarando minha variavel constante

my $greetings = "Hello world!";

# imprimindo
print 'O nome da nossa variavel eh $greetings';
print " e o conteudo dela eh \"$greetings\".\n";

exit;
```



# VARIÁVEIS ESCALARES

## Na prática

Script: [interpolacao.pl](#)

E como imprimir "Hello world!" (**COM as aspas**)?

Escapando da interpolação:

```
# exemplo04

#!/usr/bin/perl
# script para testar interpolacao

# declarando minha variavel constante

my $greetings = "Hello world!";

# imprimindo
print 'O nome da nossa variavel eh $greetings';
print " e o conteudo dela eh \"$greetings\".\n";

exit;
```

# VARIÁVEIS ESCALARES

## Na prática

Script: [interpolacao.pl](#)

Copiar exemplo05 na página da disciplina

Escapando da interpolação:

```
# exemplo04

#! /usr/bin/perl
# script para testar interpolacao

# declarando minha variavel constante

my $greetings = "Hello world!";

# imprimindo
print 'O nome da nossa variavel eh $greetings';
print " e o conteudo dela eh \"$greetings\".\n";

# exemplo05
print "Ah, se eu ganhasse R$1,00 a cada vez que \"$greetings\" fosse
usado...\n";

exit;
```

# VARIÁVEIS ESCALARES

## Na prática

Script: [interpolacao.pl](#)

Copiar exemplo05 na página da disciplina

Escapando da interpolação:

```
# exemplo04

#! /usr/bin/perl
# script para testar interpolacao

# declarando minha variavel constante

my $greetings = "Hello world!";

# imprimindo
print 'O nome da nossa variavel eh $greetings';
print " e o conteudo dela eh \"$greetings\".\n";

# exemplo05
print "Ah, se eu ganhasse R$1,00 a cada vez que \"$greetings\" fosse us

exit;
```

# VARIÁVEIS ESCALARES

## Na prática

1. No Geany, File > New File.
2. Copiar `#exemplo06` da página da disciplina.
3. File > Save as...
4. Gravar arquivo como [tabela.pl](#)

# VARIÁVEIS ESCALARES

## Na prática

Script: [tabela.pl](#)

Exemplo 06:

```
#!/usr/bin/perl
# script para criar uma tabela

# titulo e header
print "Notas da disciplina de Introducao a Programacao\n\n";
print "No USP\tNome\tNota\n";

# imprimindo
print "0001\tMaricotinha\t9,8\n";
print "0002\tJoazinho\t2,0\n";
print "0003\tJujubinha\t9,0\n";
print "0004\tJuquinha\t3,5\n";
print "0005\tMariazinha\t9,5\n";
print "0006\tPedrinho\t2,8\n";

exit;
```

# VARIÁVEIS ESCALARES

## Caracteres especiais para formatação

CARACTER	FUNÇÃO
\n	Newline
\t	Tabulação (tab)
\u ou \U	Força letras maiúsculas (uppercase) para o primeiro (\u) ou para todos os próximos caracteres (\U)
\l ou \L	Força letras minúsculas (lowercase) para o primeiro (\l) ou para todos os próximos caracteres (\L)
\E	Delimita o final do \U ou \L

# VARIÁVEIS ESCALARES

## Na prática

Script: [tabela.pl](#)

Copiar #exemplo07 da página da disciplina:

```
print "0003\tJujubinha\t9,0\n";
print "0004\tJuquinha\t3,5\n";
print "0005\tMariazinha\t9,5\n";
print "0006\tPedrinho\t2,8\n";

# continuacao do script para criar uma tabela

# alunas aprovadas
$aluna01 = "Maricotinha";
$aluna02 = "Jujubinha";
$aluna03 = "Mariazinha";

print "\nParabens as alunas $aluna01, $aluna02 e $aluna03!\n\n";

exit;
```

# VARIÁVEIS ESCALARES

## Na prática

Script: [tabela.pl](#)

Copiar #exemplo07 da página da disciplina:

```
print "0003\tJujubinha\t9,0\n";
print "0004\tJuquinha\t3,5\n";
print "0005\tMariazinha\t9,5\n";
print "0006\tPedrinho\t2,8\n";

# continuacao do script para criar uma tabela

# alunas aprovadas
$aluna01 = "Maricotinha";
$aluna02 = "Jujubinha";
$aluna03 = "Mariazinha";

print "\nParabens as alunas \U$aluna01, $aluna02 e $aluna03!\n\n";

exit;
```



# VARIÁVEIS ESCALARES

## Na prática

Script: [tabela.pl](#)

Copiar #exemplo07 da página da disciplina:

```
print "0003\tJujubinha\t9,0\n";
print "0004\tJuquinha\t3,5\n";
print "0005\tMariazinha\t9,5\n";
print "0006\tPedrinho\t2,8\n";

# continuacao do script para criar uma tabela

# alunas aprovadas
$aluna01 = "Maricotinha";
$aluna02 = "Jujubinha";
$aluna03 = "Mariazinha";

print "\n\UParabens as alunas $aluna01, $aluna02 e $aluna03!\n\n";

exit;
```

# VARIÁVEIS ESCALARES

## Na prática

Script: [tabela.pl](#)

Copiar #exemplo07 da página da disciplina:

```
print "0003\tJujubinha\t9,0\n";
print "0004\tJuquinha\t3,5\n";
print "0005\tMariazinha\t9,5\n";
print "0006\tPedrinho\t2,8\n";

# continuacao do script para criar uma tabela

# alunas aprovadas
$aluna01 = "Maricotinha";
$aluna02 = "Jujubinha";
$aluna03 = "Mariazinha";

print "\n\UParabens as alunas\E $aluna01, $aluna02 e $aluna03!\n\n";

exit;
```

# VARIÁVEIS ESCALARES

## Na prática

1. No Geany, File > New File.
2. Copiar `#exemplo08` da página da disciplina.
3. File > Save as...
4. Gravar arquivo como [boanoite.pl](#)

# VARIÁVEIS ESCALARES

## Na prática

Script: [boanoite.pl](#)

```
#!/usr/bin/perl

# Perguntar o nome do usuário
print "Qual seu nome?\n";

# Entrada do usuário
my $nome = <STDIN>;

# Cumprimentar o usuário
print "Boa noite, $nome!\n";

exit;
```

# VARIÁVEIS ESCALARES

## Na prática

Script: [boanoite.pl](#)

No terminal:

```
Darwin:Introprog Tatiana$ ./boanoite.pl
Qual seu nome?
Tatiana
Boa noite, Tatiana
!
Darwin:Introprog Tatiana$
```

# VARIÁVEIS ESCALARES

## Comando *chomp*

Remove o último caracter se ele for um *newline*

Script: [boanoite.pl](#)

```
#!/usr/bin/perl

# Perguntar o nome do usuário
print "Qual seu nome?\n";

# Entrada do usuário
my $nome = <STDIN>;

# Remover newline
chomp($nome);

# Cumprimentar o usuário
print "Boa noite, $nome!\n";
```

# VARIÁVEIS ESCALARES

## Na prática

Script: [boanoite.pl](#)

No terminal:

```
Darwin:Introprog Tatiana$ ./boanoite.pl
Qual seu nome?
Tatiana
Boa noite, Tatiana!
Darwin:Introprog Tatiana$
```

# **VARIÁVEIS ESCALARES**

**Valores de Escalares**



# VARIÁVEIS ESCALARES

## Valores de Escalares

PERL tem dois tipos básicos de escalares:

### 1. Números:

```
$y=1;           # inteiro positivo
$z=-5;          # inteiro negativo
$x = 3.14;      # real em ponto flutuante
$w = 2.75E-6;   # real em notação científica
$t = 0377;      # octal
$u = 0xffff;    # hexadecimal
```

### 2. Strings:

```
$string1 = "Oi, eu sou uma string!";    # string
$string2 = 'Oi, eu tb sou uma string!';  # string
$string3 = "ATCGATCGATCGATTGGATC";      # string
```

# VARIÁVEIS ESCALARES

## Valores

1. No Geany, File > New File.
2. File > Save as...
3. Gravar arquivo como [escalares.pl](#)
4. Criar um script que produza a seguinte saída

# VARIÁVEIS ESCALARES

## Valores

Script: [escalares.pl](#)

Output:

```
$y = 1  
$z = -5  
$x = 3.14  
$w = 2.75e-06  
$t = 255  
$u = 65535
```

```
Oi, eu sou uma string!  
Oi, eu tb sou uma string  
ATCGATCGATCGATCGATTGGATC
```

\$t = atribuir um valor octal = 0377

\$u = atribuir um valor hexadecimal = 0xffff

# VARIÁVEIS ESCALARES

## Valores

Script: [escalares.pl](#)

Output:

```
$y = 1  
$z = -5  
$x = 3.14  
$w = 2.75e-06  
$t = 255  
$u = 65535
```

```
Oi, eu sou uma string!  
Oi, eu tb sou uma string  
ATCGATCGATCGATCGATTGGATC
```

\$t = atribuir um valor octal = 0377

\$u = atribuir um valor hexadecimal = 0xffff

# VARIÁVEIS ESCALARES

## Valores

Script: [escalares.pl](#); exemplo09 da página

```
#!/usr/bin/perl

# atribuindo valores as variaveis
$y = 1;          # inteiro positivo
$z = -5;         # inteiro negativo
$x = 3.14;       # real em ponto flutuante
$w = 2.75e-6;    # real em notação científica
$t = 0377;       # octal
$u = 0xffff;     # hexadecimal

$string1 = "Oi, eu sou uma string!";    # string
$string2 = 'Oi, eu tb sou uma string';  # string
$string3 = "ATCGATCGATCGATCGATTGGATC"; # string
```

# VARIÁVEIS ESCALARES

## Valores

Script: [escalares.pl](#); exemplo09 da página

```
#continuacao

# imprimindo
print "\$y \= $y\n";
print "\$z \= $z\n";
print "\$x \= $x\n";
print "\$w \= $w\n";
print "\$t \= $t\n";
print "\$u \= $u\n\n";

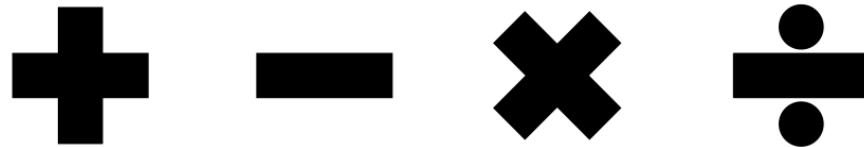
print "$string1\n$string2\n$string3\n\n";

exit;
```

# VARIÁVEIS ESCALARES

## Operações com números

1. Os "operadores da escola" estão disponíveis:



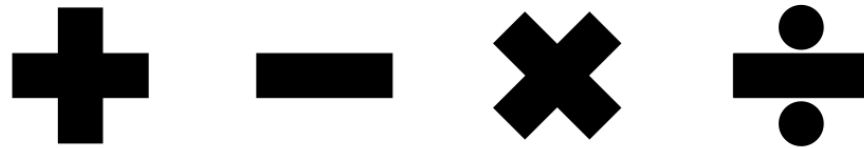
2. Precedência: praticamente igual "da escola"



# VARIÁVEIS ESCALARES

## Operações com números

1. Os "operadores da escola" estão disponíveis:



2. Precedência: praticamente igual "da escola"



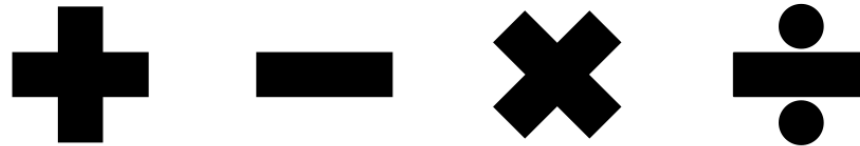
$$2+3*4$$



# VARIÁVEIS ESCALARES

## Operações com números

1. Os "operadores da escola" estão disponíveis:



2. Precedência: praticamente igual "da escola"



**Sempre use parênteses!!!**

# VARIÁVEIS ESCALARES

## Operações com números

CARACTER	FUNÇÃO
+	Adição
=	Atribuição
+=	Atribuição após soma
-=	Atribuição após subtração
++	Auto-acréscimo
--	Auto-decrécimo
/	Divisão
%	Módulo (Resto da divisão)
*	Multiplicação
**	Potenciação (Exponenciação)
sqrt()	Raiz quadrada
-	Subtração

# VARIÁVEIS ESCALARES

## Operações com números

1. No Geany, File > New File.
2. File > Save as...
3. Copiar #exemplo10 na página da disciplina.
4. Gravar arquivo como [operacoes.pl](#).

# VARIÁVEIS ESCALARES

## Operações com números

Script: [operacoes.pl](#)

```
#!/usr/bin/perl
# script para testar operacoes matematicas

# testando
$a = 1;
print "Atribuicao\:           \b$a \b= $a\b\n";

#++$a;
#print "Auto\b-acrescimo\:           \b$a \b= $a\b\n";

#--$a;
#print "Auto\b-decrescimo\:           \b$a \b= $a\b\n";

#$b = 3 + 1;
#print "Soma\:           \b$b \b= $b\b\n";

#$c = $a + $b;
#print "Soma\:           \b$c \b= $c\b\n";
```

# VARIÁVEIS ESCALARES

## Operações com números

Script: [operacoes.pl](#)

Output:

```
Atribuicao:          $a = 1
Auto-acrescimo:     $a = 2
Auto-decrescimo:    $a = 1
Soma:               $b = 4
Soma:               $c = 5
Multiplicacao:      $d = 20
Divisao:            $e = 5
Raiz quadrada:      $f = 2
Equacao:            $g = 25
Modulo:             $h = 1
Modulo:             $i = 0
Potenciacao:        $j = 25
Adicao e atribuicao:  $j = 30
Subtracao e atribuicao: $j = 25
```

