

Fundamentos de Programação Variáveis

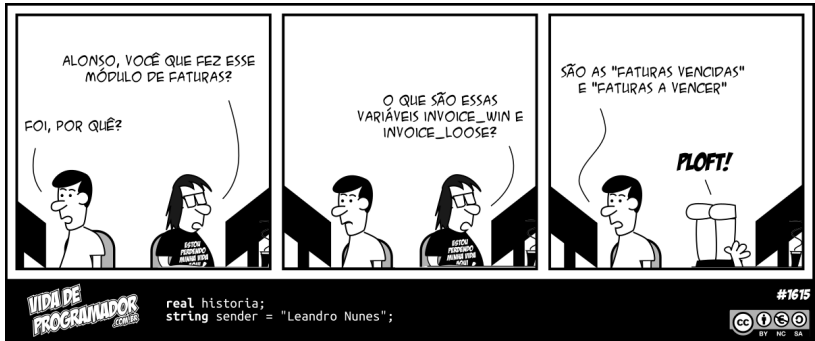
Dainf - UTFPR

Profa. Leyza Baldo Dorini
Prof. Bogdan Tomoyuki Nassu

Variáveis

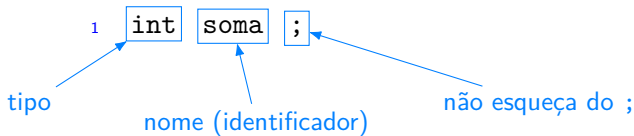
- São posições de memória capazes de reter e representar valores, os quais podem ser manipulados pelo programa.
- Só existem em tempo de execução.
- São acessadas via um nome (identificador) e possuem um tipo, o qual determina o que pode ser armazenado.

Declaração de variáveis



Declarando uma variável

Na linguagem C, antes de armazenar um valor, é preciso **declarar** a variável. A sintaxe para tal é:



Declarando uma variável: o tipo

São diversos os **tipos de variáveis** na linguagem C. Por enquanto, iremos utilizar somente dois

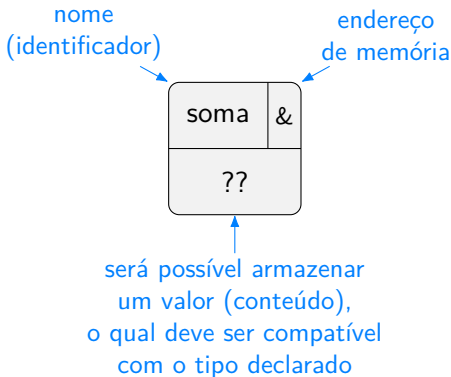
- `int`: para armazenar valores inteiros.
- `float`: para armazenar valores reais.

Atenção: o tipo precisa vir antes do nome da variável!

```
soma float;
```

Declarando uma variável: o identificador

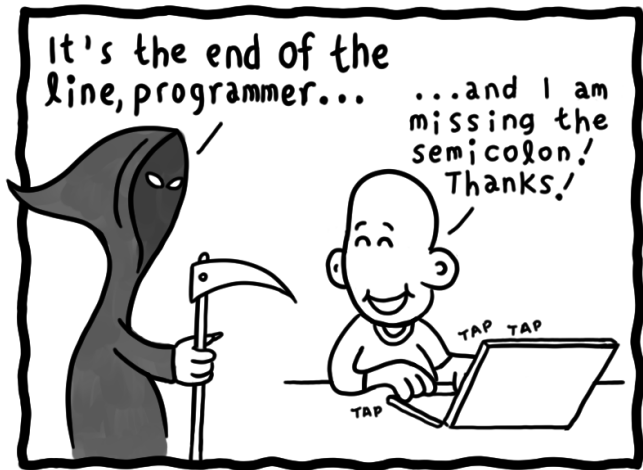
Com a declaração, um espaço de memória é **alocado**¹, o qual será acessado por meio do **identificador** da variável.



¹Alocar: *destinar (verbas, recursos etc.) a um fim específico.*

Declarando uma variável: o ponto e vírgula

Não se esqueça de finalizar a linha com o símbolo de ponto e vírgula! Caso contrário, você terá um erro de compilação!



Daniel Stori {turnoff.us}

Exemplo

```
1  int main()  
2  {  
3      int idade;  
4      int matricula;  
5      float peso, altura;  
6      float imc;  
7  
8      return 0;  
9  }
```

variáveis do mesmo
tipo podem ser
declaradas
separadamente...

...ou de uma única vez,
separadas por vírgula

Observe que variáveis do mesmo tipo podem ser declaradas juntas (os identificadores são separados por vírgula).

Em C ANSI, as variáveis são sempre declaradas após um {, mas versões mais recentes permitem fazer em outros pontos do código.

Checkpoint

Faça um programa que declare quatro variáveis do tipo float e duas do tipo int.

```
1 float peso, altura,  
2     imc;  
3 int idade;  
4 int codigo;  
5 float salario;
```

Checkpoint

Faça um programa que declare quatro variáveis do tipo float e duas do tipo int.

variáveis do mesmo

tipo podem ser
declaradas juntas...

1 float peso, altura,
2 imc;

3 int idade;

...ou separadamente

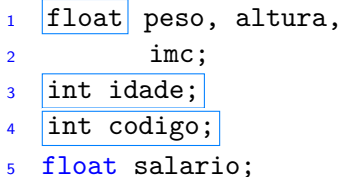
4 int codigo;

5 float salario;

Checkpoint

Faça um programa que declare quatro variáveis do tipo float e duas do tipo int.

você pode usar essa estratégia para a clareza do código, separando as variáveis em linhas diferentes (note que o ; está só no final)

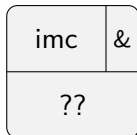
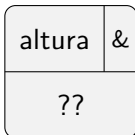
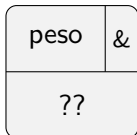
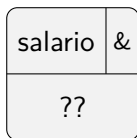
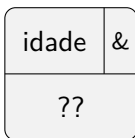
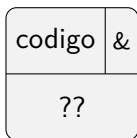


```
1 float peso, altura,  
2     imc;  
3 int idade;  
4 int codigo;  
5 float salario;
```

The diagram shows five lines of code. The keyword 'float' on line 1 is enclosed in a blue box. The keyword 'int' on line 3 is enclosed in a blue box. The keyword 'int' on line 4 is enclosed in a blue box. The keyword 'float' on line 5 is enclosed in a blue box. A blue arrow points from the text 'você pode usar essa estratégia...' to the 'float' box on line 1.

Declaração de variáveis

Após as declarações anteriores, teríamos 6 variáveis na memória:



Observe que a memória foi alocada, mas ainda tem conteúdo desconhecido! O próximo passo é atribuir valores para as variáveis!

Atribuição

Atribuir um valor a uma variável significa “copiar” tal valor para o endereço de memória alocado! Em C, a atribuição é feita da seguinte forma:

`<variável> = <valor>;`

O conteúdo de `<valor>` pode ser, por exemplo:

- Uma constante (um número);
- Uma outra variável (neste caso, o valor da variável à direita do símbolo de atribuição é **copiado** para a variável à esquerda);
 - Cuidado: as duas variáveis não ficam “conectadas”! Se o valor de uma mudar, o valor da outra **não** muda junto!
- Uma expressão (usando variáveis e/ou escalares).

Atribuição

Lembre-se: o operador de atribuição é o sinal de igual (=).

À esquerda do operador de atribuição deve existir o nome de **somente uma variável**.

=

À direita, deve haver um valor ou uma **expressão** cujo valor será calculado e armazenado na variável.

Note que este = não é como o = da matemática: o destino fica **sempre** à esquerda, e o valor a armazenar fica **sempre** à direita!

~~0.5 = peso;~~

Atenção! Lembre-se de **sempre** atribuir um valor a uma variável antes de usá-la! Esquecer disso é um erro muito comum!


OBS.: A atribuição pode ocorrer junto com a declaração!

Exemplos

```
1 float peso, altura,  
2     imc;  
3 int idade = 20;  
4 int codigo;  
5 float salario;  
6  
7 peso = 62.5;  
8 altura = 1.65;  
9  
10 codigo = idade;  
11  
12 imc = peso / (altura * altura);  
13  
14 salario = 1050.0;
```

Atenção: o valor atribuído
deve ser compatível com o
tipo da variável

atribuição de um
valor float




Exemplos

```
1 float peso, altura,
2     imc;                                atribuição de conteúdo
3 int idade = 20;                         no momento da declaração
4 int codigo;
5 float salario;
6
7 peso = 62.5;
8 altura = 1.65;
9
10 codigo = idade;
11
12 imc = peso / (altura * altura);
13
14 salario = 1050.0;
```


Exemplos

```
1 float peso, altura,  
2     imc;  
3 int idade = 20;  
4 int codigo;  
5 float salario;  
6  
7 peso = 62.5;  
8 altura = 1.65;  
9  
10 codigo = idade;  
11  
12 imc = peso / (altura * altura);  
13  
14 salario = 1050.0;
```


variável `codigo` recebe
uma cópia do conteúdo
da variável `idade`



Exemplos

```
1 float peso, altura,  
2     imc;  
3 int idade = 20;  
4 int codigo;  
5 float salario;  
6  
7 peso = 62.5;  
8 altura = 1.65;  
9  
10 codigo = idade;  
11  
12 imc = peso / (altura * altura);  
13  
14 salario = 1050.0;
```

atribuição do resultado
de uma expressão



Atribuição

Após as atribuições anteriores, as 6 variáveis na memória teriam valores válidos:

codigo	&
20	

idade	&
20	

salario	&
1050.0	

peso	&
62.5	

altura	&
1.65	

imc	&
22.95684..	

Lembre-se: é essencial que os valores atribuídos sejam **compatíveis** com os tipos das variáveis!

Atribuições substituem os valores

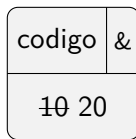
Quando atribuímos um valor a uma variável, o seu conteúdo é substituído. Por exemplo, qual o valor da variável `codigo` ao final do seguinte programa?

```
1 int main(){  
2  
3     int codigo;  
4  
5     codigo = 10;  
6     codigo = 20;  
7  
8     return 0;  
9 }
```

Atribuições substituem os valores

Quando atribuímos um valor a uma variável, o seu conteúdo é substituído. Por exemplo, qual o valor da variável `codigo` ao final do seguinte programa?

```
1 int main(){  
2  
3     int codigo;  
4  
5     codigo = 10;  
6     codigo = 20;  
7  
8     return 0;  
9 }
```



O valor 10 (atribuído com o comando da linha 5) foi substituído por 20 (linha 6). Portanto, caso o valor inicial (10) seja necessário, uma cópia deve ser armazenada em **outra** variável.

Exercícios de fixação

Exercício de fixação 1

Faça um programa que declare três variáveis do tipo float e duas do tipo int. Depois, atribua valores a todas.

Um exemplo (você pode fazer outras variações):

```
1  int main(){
2
3      float peso = 90, altura = 1.8,
4          taxa;
5      int idade = 20;
6      int codigo;
7
8      taxa = peso / altura;
9
10     codigo = 1001;
11
12     return 0;
13 }
```

Exercício de fixação 1

Faça um programa que declare três variáveis do tipo float e duas do tipo int. Depois, atribua valores a todas.

Um exemplo (você pode fazer outras variações):

```
1  int main(){
2
3      float peso = 90, altura = 1.8,
4          taxa;
5      int idade = 20;
6      int codigo;
7
8      taxa = peso / altura;
9
10     codigo = 1001;
11
12     return 0;
13 }
```


Exercício de fixação 2

O código abaixo deveria trocar o conteúdo de duas variáveis, mas ambas terminam com o mesmo valor! O que está errado?

```
1 int main(){
2     int primeiro= 10,
3         segundo = 20;
4
5     primeiro = segundo;
6     segundo = primeiro;
7     return 0;
8 }
9
```



comando da linha 2 declarou
e inicializou (com o valor 10)
a variável chamada `primeiro`



comando da linha 3 declarou
e inicializou (com o valor 20)
a variável chamada `segundo`

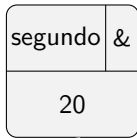
Exercício de fixação 2

O código abaixo deveria trocar o conteúdo de duas variáveis, mas ambas terminam com o mesmo valor! O que está errado?

```
1 int main(){  
2     int primeiro= 10,  
3         segundo = 20;  
4  
5     primeiro = segundo;  
6     segundo = primeiro;  
7     return 0;  
8 }  
9
```



comando da linha 2 declarou
e inicializou (com o valor 10)
a variável chamada primeiro

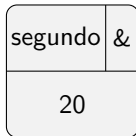
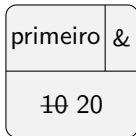


comando da linha 3 declarou
e inicializou (com o valor 20)
a variável chamada segundo

Exercício de fixação 2

O código abaixo deveria trocar o conteúdo de duas variáveis, mas ambas terminam com o mesmo valor! O que está errado?

```
1 int main(){
2     int primeiro= 10,
3         segundo = 20;
4
5     primeiro = segundo;
6     segundo = primeiro;
7     return 0;
8 }
9
```

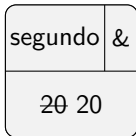


comando da linha 5 substituiu
o conteúdo inicial (10) por uma cópia
do conteúdo da variável segundo (20)

Exercício de fixação 2

O código abaixo deveria trocar o conteúdo de duas variáveis, mas ambas terminam com o mesmo valor! O que está errado?

```
1  int main(){
2      int primeiro= 10,
3          segundo = 20;
4
5      primeiro = segundo;
6      segundo = primeiro;
7      return 0;
8  }
9
```



comando da linha 6 atribui
o conteúdo da variável `primeiro`,
que neste momento já é 20

Exercício de fixação 2

Para fazer a troca, precisamos de uma variável auxiliar! Lembra do exercício dos copos de suco?

```
1  int main()
2  {
3      int primeiro, segundo, auxiliar;
4
5      primeiro = 10;
6      segundo = 20;
7
8      auxiliar = primeiro;
9      primeiro = segundo;
10     segundo = auxiliar;
11
12     return 0;
13 }
14
```

Nomes de variáveis



Nome de variáveis em C

Em C, um nome de variável:

- ❶ **Deve** começar com uma letra (maiúscula ou minúscula) ou underline () (ou seja, nunca pode começar com um número).
- ❷ Pode conter letras maiúsculas, minúsculas, números e o símbolo de *underline*.
- ❸ Evite iniciar nomes de variáveis com letras maiúsculas (convenção).

Exercício: qual das seguintes opções contém apenas nomes de variáveis válidos?

() aux, 2id, hora_inicial, num, iDaD&

() aux, id2, hora_inicial, _num, iDaDe

() au*, id2, hora_inicial, num, iDaDe

() aux, -id2, hora/inicial, num, iDaDe

Nome de variáveis em C

Em C, um nome de variável:

- ① **Deve** começar com uma letra (maiúscula ou minúscula) ou underline () (ou seja, nunca pode começar com um número).
- ② Pode conter letras maiúsculas, minúsculas, números e o símbolo de *underline*.
- ③ Evite iniciar nomes de variáveis com letras maiúsculas (convenção).

Exercício: qual das seguintes opções contém apenas nomes de variáveis válidos?

- () aux, 2id, hora_inicial, num, iDaD&
- () aux, id2, hora_inicial, _num, iDaDe
- () au*, id2, hora_inicial, num, iDaDe
- () aux, -id2, hora/inicial, num, iDaDe

Nome de variáveis em C

Em C, um nome de variável:

- ① **Deve** começar com uma letra (maiúscula ou minúscula) ou underline (_) (ou seja, nunca pode começar com um número).
- ② Pode conter letras maiúsculas, minúsculas, números e o símbolo de *underline*.
- ③ Evite iniciar nomes de variáveis com letras maiúsculas (convenção).

Exercício: qual das seguintes opções contém apenas nomes de variáveis válidos?

- () aux, **2id**, hora_inicial, num, **iDaD&**
- (X) aux, id2, hora_inicial, _num, iDaDe
- () **au***, id2, hora_inicial, num, iDaDe
- () aux, **-id2**, **hora/inicial**, num, iDaDe

Escolha nomes significativos

Usar **nomes significativos** para as variáveis é uma boa prática de programação. Lembre-se que a clareza do seu programa é importante! Por exemplo, qual informação estaria armazenada nas variáveis abaixo?



uma altura



impossível saber
(pode ser idade,
código, dia do mês...)

Outras observações

- Escolha nomes significativos, mas não muito longos.
- Use letras maiúsculas ou underline ('_') para juntar palavras. Por exemplo, `diaInicio` ou `hora_inicio`. Use o que você preferir, mas seja consistente (ou seja, padronize o uso em seu programa).
- A linguagem C faz distinção entre letras maiúsculas e minúsculas, ou seja, `soma` e `somA` seriam variáveis diferentes (mas você **não vai** criar variáveis com nomes tão parecidos, vai?).
- Toda linguagem de programação tem algumas palavras reservadas, que você não pode usar como nomes de variáveis (por exemplo, você não pode criar uma variável chamada `int`!).



Variáveis e próximos passos

Neste tópico, estudamos variáveis (sua definição, declaração e inicialização). É importante que você faça todos os exercícios de fixação propostos (e compreenda a solução)!

