

# Algoritmos e Lógica de Programação

80 horas // 4 h/semana

---

## ***Estrutura Sequencial***

### **Aula 04**

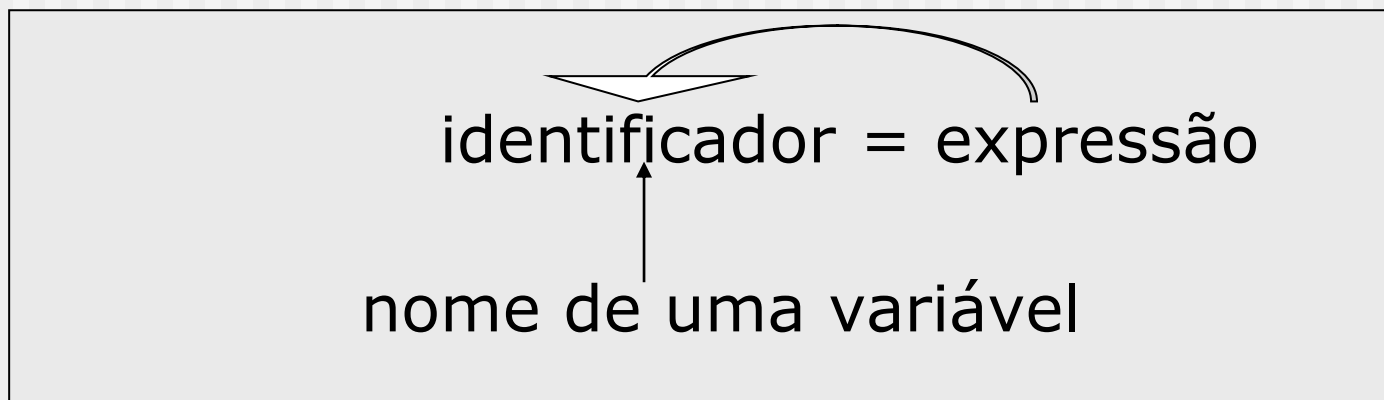
Prof. Piva

# Expressões Aritméticas...

---

- operandos: constantes e/ou variáveis numéricas;
- operadores: adição(+), subtração(-), multiplicação(\*) e divisão(/).
- resultado de uma expressão aritmética: armazenado em uma variável para uso posterior;
- guardar o resultado de uma expressão aritmética significa um comando de atribuição.

# Comando de Atribuição...


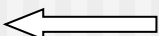
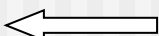


**Significado:** o resultado da expressão será armazenado no identificador

**Observação:** não se trata de uma igualdade matemática. É a atribuição de um valor para um identificador. O identificador por sua vez, não é um número e nem uma constante.

# Comandos de Atribuição...

Seja a seguinte sequência de comandos de atribuição:

$A = 5$		à variável A é atribuído o valor 5
$B = 3$		à variável B é atribuído o valor 3
$A = B$		à variável A é atribuído o valor 3

**Observação:** para que a atribuição ( $A = B$ ) não resulte num “erro”, é necessário que a variável B tenha valor, pois é seu valor que será atribuído à variável A.

# Comando de Atribuição...

identificador  $\leftarrow$  expressão

ou

identificador  $:=$  expressão

## **análise do comando:**

- *identificador* é uma variável e sempre aparecerá do lado esquerdo da atribuição;
- a seta ( $\leftarrow$ ) símbolos: menor ( $<$ ) e subtração ( $-$ ) juntos sem espaços ou ( $:=$ ) símbolos: dois-pontos ( $:$ ) e igual ( $=$ ) juntos sem espaço

# Comandos de Entrada...

---

Formato do comando - maioria das linguagens é composto dos elementos:

- **nome do comando;**
- um ou um conjunto de **identificadores** onde cada um deles receberá um valor que virá de alguma fonte de dados, podendo ser de um teclado (valores digitados), de um arquivo, de um formulário preenchido.

# Comandos de Entrada...

---

## Exemplo:

**leia (A, B, C)**

- No exemplo, o comando (Leia) fará a leitura de três dados, o primeiro será atribuído à variável A, o segundo à variável B e o terceiro à variável C.

# Comandos de Saída...

---

Formato do comando - maioria das linguagens é composto dos elementos:

- **nome do comando;**
- uma lista de identificadores e/ou constantes que se deseja imprimir.



# Comandos de Saída...

---

## Exemplo:

**escreva** (A, B, C, 20, "Nome", 'x')

No exemplo, o comando (Escreva) fará a impressão de:

- conteúdo das variáveis: A, B e C;
- constante: 20;
- cadeia-de-caracteres: Nome;
- caractere: x.

# Comandos de Saída...

---

Além do comando **escreva**, existe o **escreval**:

**escreval** (*<lista-de-expressões>*)

## **análise do comando:**

- O comando **escreval** muda de linha após terminar de executar o comando de saída. E, o **escreva** o cursor permanece na posição em que estava ao terminar a execução do comando;
- *lista-de-expressões*: as expressões são analisadas e os seus resultados são impressos na saída padrão. Devem estar entre parênteses e separadas por vírgula.

# Exemplo 1

---

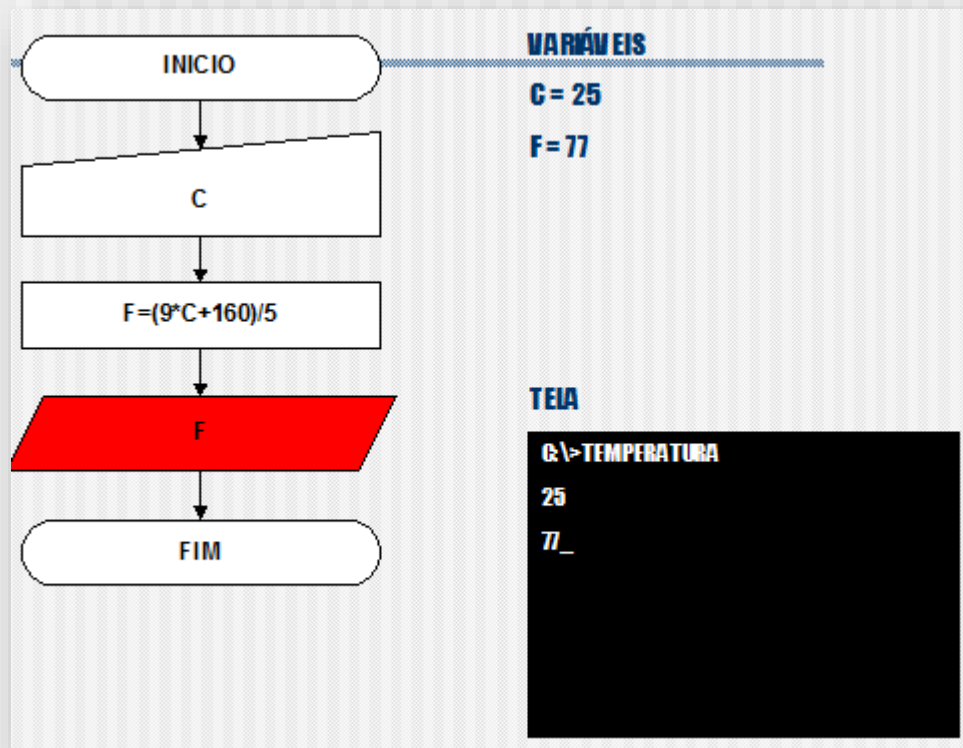
**Calcular a área de um quadrilátero, com dois lados iguais. Para isso faz a leitura de dois números inteiros, representando os lados e imprime os valores lidos e a área calculada.**

**base, altura, area** representam os nomes das variáveis – podem ser do tipo inteiro ou do tipo real.

```
leia(base, altura)  
area = base * altura  
escreva( base, altura, area)
```

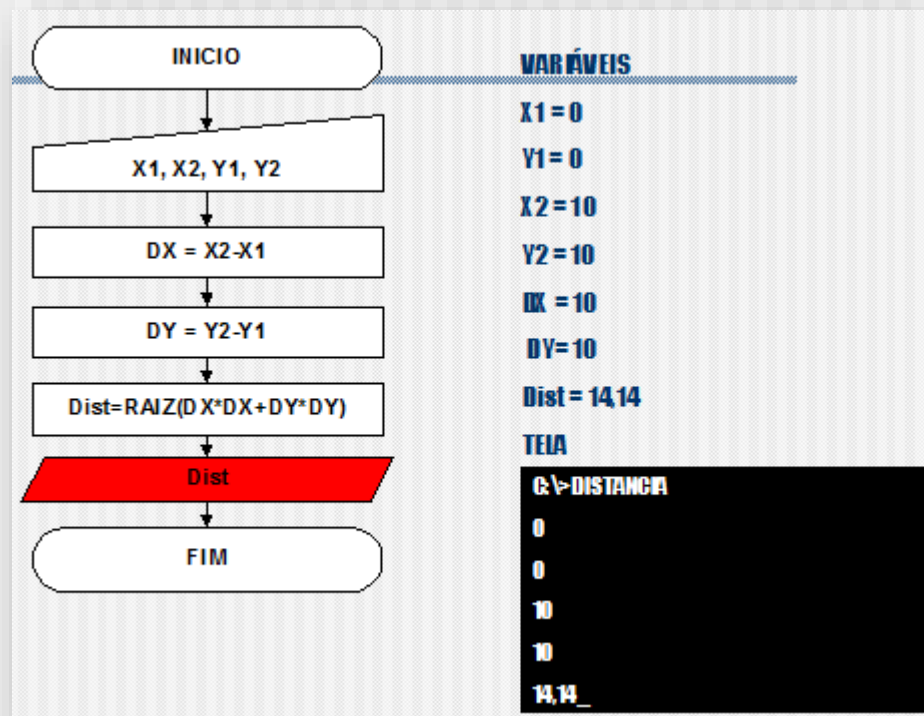
# Exemplo 2

**Ler a temperatura em °C e apresentar em °F.  
Dados :  $F = (9 \cdot C + 160) / 5$ .**

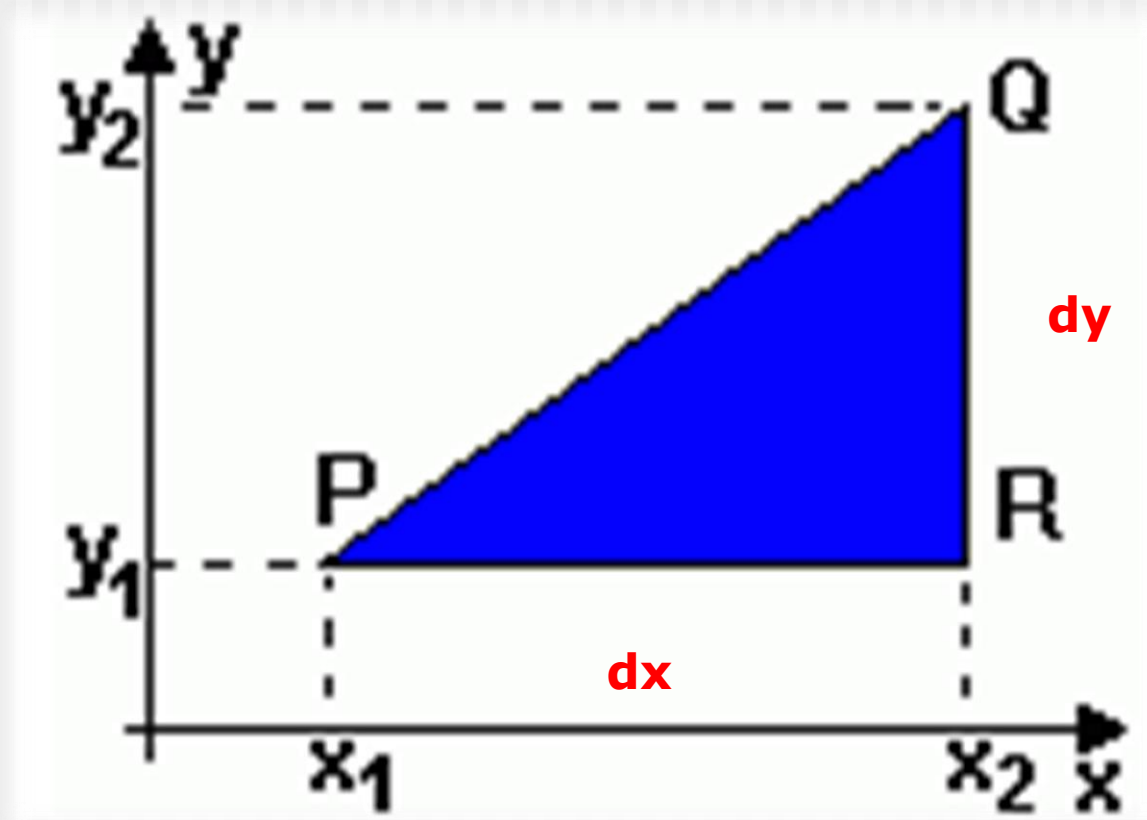


# Exemplo 3

## CALCULAR A DISTÂNCIA ENTRE 2 PONTOS.



# Exemplo 3



# VAMOS PARA A PRÁTICA ?!!!

---



# **Python: Comandos Básicos**



# Primeiros passos em programação

- Até agora só vimos como computar algumas expressões simples
  - Expressões são escritas e computadas imediatamente
  - Variáveis podem ser usadas para valores temporários
- Um programa típico entretanto usa vários tipos de construções tais como:
  - Comandos condicionais
  - Comandos de repetição
  - Definição e uso de procedimentos (subprogramas)
  - Definição e uso de classes e objetos (programação OO)

# Primeiros passos em programação

```
>>> # Série de Fibonacci
```

Comentário

```
... a,b=0,1
```

Atribuição dupla

```
>>> while b < 10:
```

Comando de repetição

```
...     print (b)
```

```
...     a,b=b,a+b
```

Expressão booleana

```
...
```

```
1
```

Indentação

Bloco do  
comando de repetição

```
1
```

```
2
```

```
3
```

```
5
```

```
8
```

Resultado

# Programas armazenados

- À medida que os programas vão se tornando mais complicados, é mais interessante guardá-los em arquivos e executá-los quando necessário
- Arquivo fibo.py (use um editor de textos como o do IDLE):
  - # Série de Fibonacci:
  - a, b = 0, 1
  - while b < 10:
    - print (b)
    - a, b = b, a+b

# Formas de Executar um Programa

- Digite *python fibo.py* no seu shell, ou
- Clique no ícone do arquivo, ou
- De dentro do editor IDLE, selecione *Run Module (F5)*, ou
- De dentro do interpretador *python*:

```
>>> execfile ("fibo.py")
```

```
Entre com um numero 5
```

```
1 1 2 3
```

```
>>>
```

# print

- Forma geral: **print *expr,expr,...***
- Os valores das expressões são escritos um após o outro sem pular de linha:  

```
>>> print ("1.001 ao quadrado é ",1.001**2)
1.001 ao quadrado é  1.002001
```
- Se o comando terminar com vírgula, o próximo print escreverá na mesma linha. Por exemplo:  

```
>>> a, b = 0, 1
>>> while b < 1000:
...     print (b, end="")
...     a, b = b, a+b
...
1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987
```

# input

- O programa que computa elementos da série de Fibonacci termina quando atinge um elemento com valor superior a uma constante
- Podemos tornar o programa mais flexível se ao usuário for permitido estipular o valor máximo
- O comando `input` permite perguntar ao usuário um valor (normalmente é atribuído a uma variável)
  - Formato: **`input(pergunta)`**
  - onde *pergunta* é uma string opcional que será exibida para indicar o valor que se espera (i.e., *prompt*)
- Exemplo:

```
>>> a = input("Entre com um numero: ")
Entre com um numero: 19
>>> print (a)
19
```



Usuário digita o número

# Input

- O comando input espera que se digite algo, e retorna uma string, sem nenhum tipo de interpretação (raw\_input do python 2).
  - O resultado é simplesmente uma string com o texto digitado
- Ex.:

```
>>> nome = input ("Entre seu nome: ")
Entre seu nome: Dilermando Piva Jr
>>> print (nome)
Dilermando Piva Jr
>>> nome
'Dilermando Piva Jr'
```

# Palavras reservadas

---

- São palavras que têm um significado especial para a linguagem.

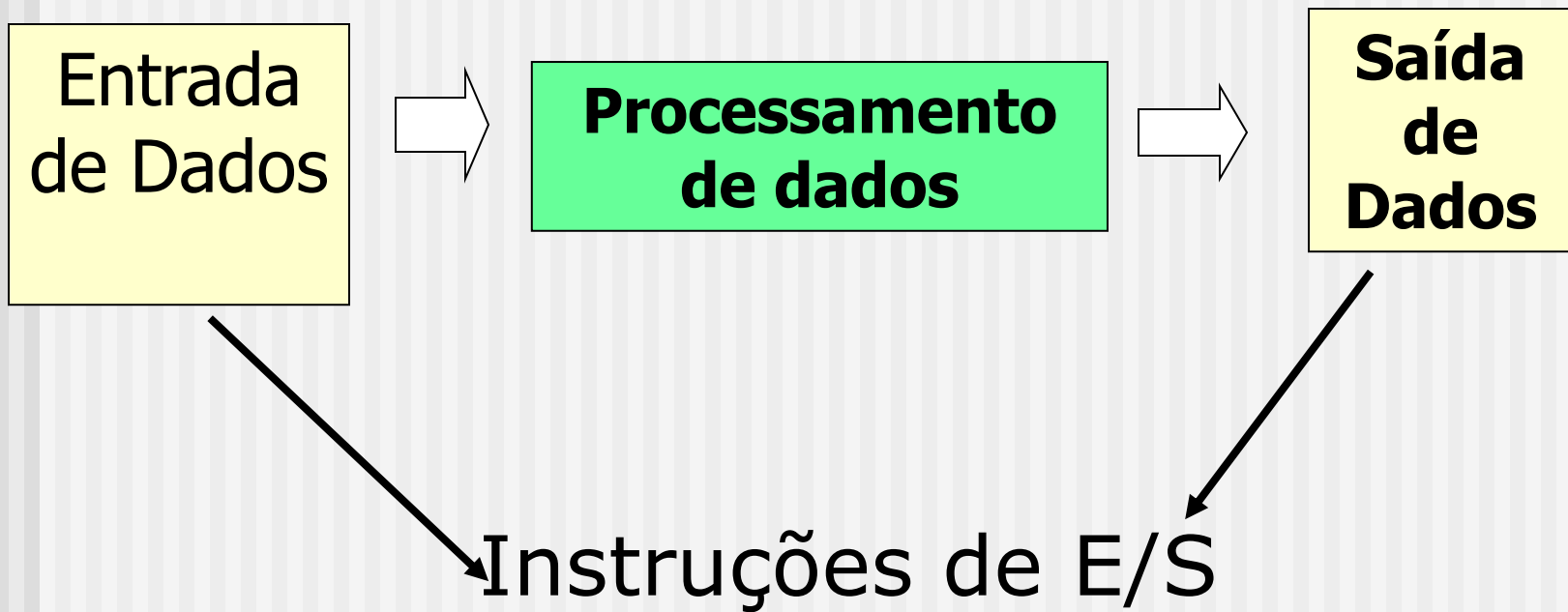
and	as	assert	break
class	continue	def	del
elif	else	except	False
finally	for	from	global
if	import	in	is
lambda	None	nonlocal	not
or	pass	raise	return
True	try	while	with
yield			

- Python entende tais palavras apenas da forma como estão escritas



# Comandos básicos

---



# Instruções de E/S – input()

---

- Leitura de dados tipados via teclado
- `input ("string de controle")`

Exemplo:

```
input("entre com um no.")
```

# Comandos de E/S – print( )

---

- Apresentação de dados no monitor
- **print("string de controle", lista de argumentos)**

Exemplo:

```
print("Olá... Seja bem-vindo(a)")
```

# Caracteres Especiais

---

`\n` nova linha

`\r` enter

`\t` tabulação (tab)

`\b` retrocesso

`\"` aspas

`\\` barra

# Programando – Exercício 0

---

- Faça um programa para ler e escrever na tela o seu nome e a sua idade.

```
nome = input("Entre com o seu nome: ")  
idade = input("Entre com sua idade: ")  
print("Nome: ", nome, " Idade: ", idade)
```

# Operadores Aritméticos

---

Operador	Ação
+	Adição
*	Multiplicação
/	Divisão
%	Resto da divisão inteira
-	Subtração (unário)

# Operadores Relacionais e Lógicos

Operador	Ação
>	Maior que
>=	Maior ou igual que
<	Menor que
<=	Menor ou igual que
==	Igual a
!=	Diferente de
and	Condição "E"
or	Condição "OU"
not	Não

# Exemplo 1

---

- Cálculo da área de um quadrilátero.

```
base = float(input("Entre com a base: "))  
altura = float(input("Entre com a altura: "))  
area = (base * altura)  
print("A área do quadrilátero é: ", area)
```



# Exemplo 2

---

- Temperatura -> °C para °F

```
celsius = float(input("Entre com a temperatura em C:"))  
fare = (9 * celsius + 160) / 5  
print("A temperatura em F = ", fare)
```

# Exemplo 3

## ■ Distância entre dois pontos

```
from math import pow, sqrt

x1 = float(input("Entre com x1: "))
y1 = float(input("Entre com y1: "))
x2 = float(input("Entre com x2: "))
y2 = float(input("Entre com y2: "))
dx = x2 - x1
dy = y2 - y1
d = sqrt(pow(dx,2)+pow(dy,2))

print("Distância = ", d)
```

# EXERCÍCIOS

---

# Exercício 1

---

Declare corretamente as variáveis com base nas seguintes atribuições:

$A \leftarrow \text{exp}(3, 2) \leq 6$

$B \leftarrow 12 \text{ div } 2$

$C \leftarrow "3"$

$D \leftarrow \text{exp}(3, 2) + \text{raizq}(16)$

$X \leftarrow \text{"Bom-dia!"}$

# Exercício 2

---

Escreva um algoritmo que receba o nome do usuário e lhe deseje "Bom-dia!".

# Exercício 3

---

Faça um algoritmo que receba o ano de nascimento de uma pessoa e o ano atual e mostre:

- a) A idade dessa pessoa em anos;
- b) A idade dessa pessoa em meses;
- c) A idade dessa pessoa em dias;
- d) A idade dessa pessoa em semanas.

# Exercício 4

---

Faça um algoritmo que receba a data de nascimento de uma pessoa e a data atual e mostre sua idade em anos, meses, semanas e dias

# Exercício 5

---

Faça um algoritmo que receba o salário de um funcionário, calcule e mostre o novo salário, sabendo-se que este sofreu um aumento de 25%.



# Exercício 6

---

Faça um algoritmo que receba o salário de um funcionário e o percentual de aumento, calcule e mostre o valor do aumento e o novo salário.

# Exercício 7

---

- Faça um algoritmo que receba o valor de um depósito e o valor da taxa de juros, calcule e mostre o valor do rendimento e o valor total depois do rendimento.

# Exercício 8

---

Faça um algoritmo que calcule e mostre a área de um triângulo. Sabe-se que  $\text{Área} = (\text{base} * \text{altura}) / 2$

# Exercício 9

---

Faça um algoritmo que calcule e mostre a área de um círculo. Sabe-se que

$$\text{Area} = \pi * R^2$$

$$\pi = 3,1415$$

# Exercício 10

---

Faça um algoritmo que receba um número positivo e maior que zero, calcule e mostre:

- a) O número digitado ao quadrado
- b) O número digitado ao cubo
- c) A raiz quadrada do número digitado

Observação:

$\text{Exp}(x,y)$  – Calcula a potência de  $x$  elevado a  $y$

$\text{Raizq}(x)$  – Calcula a raiz quadrada de  $x$

# Exercício 11

---

Sabe-se que:

- 1 pé = 12 polegadas
- 1 jarda = 3 pés
- 1 milha = 1760 jardas

Faça um algoritmo que receba uma medida em pés, faça as conversões a seguir e mostre os resultados:

- a) Polegadas
- b) Jardas
- c) milhas

# Exercício 12

---

Cada degrau de uma escada tem  $X$ cm de altura. Faça um algoritmo que receba a altura de cada degrau em cm e a altura que o usuário deseja alcançar subindo a escada (em metros). Faça as conversões, calcule e mostre quantos degraus o usuário deverá subir para atingir seu objetivo.

Obs: não se preocupe com a altura do usuário!

# Exercício 13

---

Sabe-se que o quilowatt de energia custa um oitavo do salário mínimo. Faça um algoritmo que receba o valor do salário mínimo e a quantidade de quilowatts consumida por uma residência. Calcule e mostre:

- a) O valor, em reais, de cada quilowatt
- b) O valor, em reais, a ser pago por essa residência
- c) O valor, em reais, a ser pago com desconto de 15%



# Exercício 14

---

Faça um algoritmo que receba uma hora formada por hora e minutos (um número real), calcule e mostre a hora digitada apenas em minutos. Lembre-se de que:

- Para quatro e meia deve-se digitar: 4,30
- Os minutos vão de 0 a 60!!