

# Primeiros Programas

## Aula 1

Diego Padilha Rubert

Faculdade de Computação  
Universidade Federal de Mato Grosso do Sul

Algoritmos e Programação

# Conteúdo da aula

- 1 Primeiro programa
- 2 Próximo programa
- 3 Documentação
- 4 Entrada e saída
- 5 Simulação passo a passo da execução de programas
- 6 Exercícios

- ▶ Nesta disciplina utilizaremos os termos “programa” e “script” de forma intercambiável.
- ▶ Atualmente, há duas versões em uso do Python, a 2.x e a 3.x, aqui utilizaremos versão 3.x. Contudo, 90% é compatível com a versão 2.x.
- ▶ Python 2.x deixou de ser atualizado em 1 de janeiro de 2020

- ▶ Nesta disciplina utilizaremos os termos “programa” e “script” de forma intercambiável.
- ▶ Atualmente, há duas versões em uso do Python, a 2.x e a 3.x, aqui utilizaremos versão 3.x. Contudo, 90% é compatível com a versão 2.x.
- ▶ Python 2.x deixou de ser atualizado em 1 de janeiro de 2020

- ▶ Nesta disciplina utilizaremos os termos “programa” e “script” de forma intercambiável.
- ▶ Atualmente, há duas versões em uso do Python, a 2.x e a 3.x, aqui utilizaremos versão 3.x. Contudo, 90% é compatível com a versão 2.x.
- ▶ Python 2.x deixou de ser atualizado em 1 de janeiro de 2020

# Antes do primeiro programa - O console Python

- ▶ **nosso objetivo é construir programas completos**
- ▶ contudo, o **console** do Python é muito útil para testarmos instruções individuais ou um pequeno conjunto delas
- ▶ no Windows, a instalação completa do Python inclui o ambiente IDLE, que oferece inicialmente um console do Python, mas permite também criarmos programas/scripts completos utilizando a linguagem Python

# Antes do primeiro programa - O console Python

- ▶ nosso objetivo é construir programas completos
- ▶ contudo, o **console** do Python é muito útil para testarmos instruções individuais ou um pequeno conjunto delas
- ▶ no Windows, a instalação completa do Python inclui o ambiente IDLE, que oferece inicialmente um console do Python, mas permite também criarmos programas/scripts completos utilizando a linguagem Python

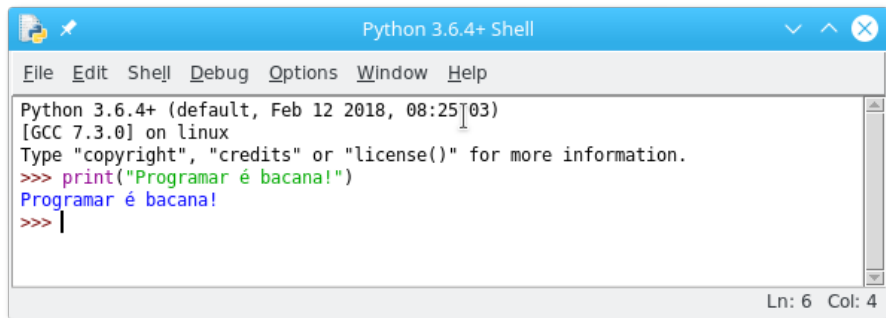
# Antes do primeiro programa - O console Python

- ▶ nosso objetivo é construir programas completos
- ▶ contudo, o **console** do Python é muito útil para testarmos instruções individuais ou um pequeno conjunto delas
- ▶ no Windows, a instalação completa do Python inclui o ambiente IDLE, que oferece inicialmente um console do Python, mas permite também criarmos programas/scripts completos utilizando a linguagem Python



# Antes do primeiro programa - O console Python

Teste no console do Python de uma instrução que irá compor nosso primeiro programa:



```
Python 3.6.4+ Shell
File Edit Shell Debug Options Window Help
Python 3.6.4+ (default, Feb 12 2018, 08:25:03)
[GCC 7.3.0] on linux
Type "copyright", "credits" or "license()" for more information.
>>> print("Programar é bacana!")
Programar é bacana!
>>> |
```

Ln: 6 Col: 4

# Primeiro programa

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

print("Programar é bacana!")

exit(0)
```

## Importante!

- ▶ letras minúsculas e maiúsculas são diferentes na linguagem Python;
- ▶ não há distinção de onde começamos digitar nosso programa: adicionamos linhas em branco em pontos estratégicos para facilitar a leitura do programa;
- ▶ aspas duplas e aspas simples são equivalentes;

# Primeiro programa

## Importante!

- ▶ letras minúsculas e maiúsculas são diferentes na linguagem Python;
- ▶ não há distinção de onde começamos digitar nosso programa: adicionamos linhas em branco em pontos estratégicos para facilitar a leitura do programa;
- ▶ aspas duplas e aspas simples são equivalentes;

## Importante!

- ▶ letras minúsculas e maiúsculas são diferentes na linguagem Python;
- ▶ não há distinção de onde começamos digitar nosso programa: adicionamos linhas em branco em pontos estratégicos para facilitar a leitura do programa;
- ▶ aspas duplas e aspas simples são equivalentes;

# Próximo programa

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

num1 = 25
num2 = 30
soma = num1 + num2
print("A soma de %d e %d é %d" % (num1, num2, soma))

exit(0)
```

- ▶ **declaração de variáveis:** não é necessário definir variáveis, elas passam a existir no momento em que são utilizadas pela primeira vez
- ▶ 3 compartimentos de memória são reservados pelo computador para armazenamento de informações, e a cada um é associado um nome: `num1`, `num2` e `soma`
- ▶ uma variável pode ser alterada a qualquer momento e armazenar qualquer tipo de dado a qualquer momento (a variável não possui um tipo específico, mas os valores armazenados sim!)
- ▶ em geral, na linguagem Python conseguimos armazenar valores tão grandes quanto se queira, isto (o espaço de memória reservado) é gerenciado automaticamente

# Próximo programa

- ▶ **declaração de variáveis:** não é necessário definir variáveis, elas passam a existir no momento em que são utilizadas pela primeira vez
- ▶ 3 compartimentos de memória são reservados pelo computador para armazenamento de informações, e a cada um é associado um nome: `num1`, `num2` e `soma`
- ▶ uma variável pode ser alterada a qualquer momento e armazenar qualquer tipo de dado a qualquer momento (a variável não possui um tipo específico, mas os valores armazenados sim!)
- ▶ em geral, na linguagem Python conseguimos armazenar valores tão grandes quanto se queira, isto (o espaço de memória reservado) é gerenciado automaticamente



- ▶ **declaração de variáveis:** não é necessário definir variáveis, elas passam a existir no momento em que são utilizadas pela primeira vez
- ▶ 3 compartimentos de memória são reservados pelo computador para armazenamento de informações, e a cada um é associado um nome: `num1`, `num2` e `soma`
- ▶ uma variável pode ser alterada a qualquer momento e armazenar qualquer tipo de dado a qualquer momento (a variável não possui um tipo específico, mas os valores armazenados sim!)
- ▶ em geral, na linguagem Python conseguimos armazenar valores tão grandes quanto se queira, isto (o espaço de memória reservado) é gerenciado automaticamente

- ▶ **declaração de variáveis:** não é necessário definir variáveis, elas passam a existir no momento em que são utilizadas pela primeira vez
- ▶ 3 compartimentos de memória são reservados pelo computador para armazenamento de informações, e a cada um é associado um nome: `num1`, `num2` e `soma`
- ▶ uma variável pode ser alterada a qualquer momento e armazenar qualquer tipo de dado a qualquer momento (a variável não possui um tipo específico, mas os valores armazenados sim!)
- ▶ em geral, na linguagem Python conseguimos armazenar valores tão grandes quanto se queira, isto (o espaço de memória reservado) é gerenciado automaticamente

# Próximo programa

- ▶ **atribuição** do valor do tipo inteiro 25 para a variável **num1**
- ▶ **operador de atribuição** da linguagem Python é **=**
- ▶ referente à variável **soma**, ocorre a atribuição do resultado da avaliação da **expressão aritmética** **num1 + num2**
- ▶ o computador consulta o conteúdo das variáveis **num1** e **num2**, realiza a operação de adição com os dois valores obtidos dessas variáveis e atribui o resultado à variável **soma**

# Próximo programa

- ▶ **atribuição** do valor do tipo inteiro 25 para a variável `num1`
- ▶ **operador de atribuição** da linguagem Python é `=`
- ▶ referente à variável `soma`, ocorre a atribuição do resultado da avaliação da **expressão aritmética** `num1 + num2`
- ▶ o computador consulta o conteúdo das variáveis `num1` e `num2`, realiza a operação de adição com os dois valores obtidos dessas variáveis e atribui o resultado à variável `soma`

# Próximo programa

- ▶ **atribuição** do valor do tipo inteiro 25 para a variável `num1`
- ▶ **operador de atribuição** da linguagem Python é `=`
- ▶ referente à variável `soma`, ocorre a atribuição do resultado da avaliação da **expressão aritmética** `num1 + num2`
- ▶ o computador consulta o conteúdo das variáveis `num1` e `num2`, realiza a operação de adição com os dois valores obtidos dessas variáveis e atribui o resultado à variável `soma`

# Próximo programa

- ▶ **atribuição** do valor do tipo inteiro 25 para a variável **num1**
- ▶ **operador de atribuição** da linguagem Python é **=**
- ▶ referente à variável **soma**, ocorre a atribuição do resultado da avaliação da **expressão aritmética** **num1 + num2**
- ▶ o computador consulta o conteúdo das variáveis **num1** e **num2**, realiza a operação de adição com os dois valores obtidos dessas variáveis e atribui o resultado à variável **soma**

# Próximo programa

- ▶ fazemos uma **chamada à função** `print` com um único argumento

- ▶ **único argumento??** observe o exemplo:

```
print("A soma dos 2 números %d e %d é %d" % (num1, num2, soma))
```

- ▶ o texto entre aspas é uma **cadeia de caracteres de formatação**, contendo não apenas caracteres a serem impressos na saída, mas símbolos especiais, iniciados com `%`, conhecidos como **conversores de tipo**
- ▶ os conversores de tipo são substituídos pelos valores das variáveis correspondentes, e apenas o texto final é passado à função `print`, ou seja, a função está recebendo um único argumento
- ▶ mais adiante veremos outros tipos de valores e seus respectivos conversores (utilizamos `%d` apenas para valores inteiros)

# Próximo programa

- ▶ fazemos uma **chamada à função** `print` com um único argumento
- ▶ **único argumento??** observe o exemplo:

```
print("A soma dos 2 números %d e %d é %d" % (num1, num2, soma))
```

- ▶ o texto entre aspas é uma **cadeia de caracteres de formatação**, contendo não apenas caracteres a serem impressos na saída, mas símbolos especiais, iniciados com `%`, conhecidos como **conversores de tipo**
- ▶ os conversores de tipo são substituídos pelos valores das variáveis correspondentes, e apenas o texto final é passado à função `print`, ou seja, a função está recebendo um único argumento
- ▶ mais adiante veremos outros tipos de valores e seus respectivos conversores (utilizamos `%d` apenas para valores inteiros)



# Próximo programa

- ▶ fazemos uma **chamada à função** `print` com um único argumento
- ▶ **único argumento??** observe o exemplo:

```
print("A soma dos 2 números %d e %d é %d" % (num1, num2, soma))
```

- ▶ o texto entre aspas é uma **cadeia de caracteres de formatação**, contendo não apenas caracteres a serem impressos na saída, mas símbolos especiais, iniciados com `%`, conhecidos como **conversores de tipo**
- ▶ os conversores de tipo são substituídos pelos valores das variáveis correspondentes, e apenas o texto final é passado à função `print`, ou seja, a função está recebendo um único argumento
- ▶ mais adiante veremos outros tipos de valores e seus respectivos conversores (utilizamos `%d` apenas para valores inteiros)

# Próximo programa

- ▶ fazemos uma **chamada à função** `print` com um único argumento
- ▶ **único argumento??** observe o exemplo:

```
print("A soma dos 2 números %d e %d é %d" % (num1, num2, soma))
```

- ▶ o texto entre aspas é uma **cadeia de caracteres de formatação**, contendo não apenas caracteres a serem impressos na saída, mas símbolos especiais, iniciados com `%`, conhecidos como **conversores de tipo**
- ▶ os conversores de tipo são substituídos pelos valores das variáveis correspondentes, e apenas o texto final é passado à função `print`, ou seja, a função está recebendo um único argumento
- ▶ mais adiante veremos outros tipos de valores e seus respectivos conversores (utilizamos `%d` apenas para valores inteiros)

# Próximo programa

- ▶ fazemos uma **chamada à função** `print` com um único argumento
- ▶ **único argumento??** observe o exemplo:

```
print("A soma dos 2 números %d e %d é %d" % (num1, num2, soma))
```

- ▶ o texto entre aspas é uma **cadeia de caracteres de formatação**, contendo não apenas caracteres a serem impressos na saída, mas símbolos especiais, iniciados com `%`, conhecidos como **conversores de tipo**
- ▶ os conversores de tipo são substituídos pelos valores das variáveis correspondentes, e apenas o texto final é passado à função `print`, ou seja, a função está recebendo um único argumento
- ▶ mais adiante veremos outros tipos de valores e seus respectivos conversores (utilizamos `%d` apenas para valores inteiros)

# Função `print`

- ▶ a função `print` também pode receber como argumento uma única variável:

```
print (num1)
```

Segundo o professor Paulo Feofiloff, uma boa documentação de um programa significa:

- ▶ inserir comentários apropriados no código de modo a explicar *o que* cada uma das funções que compõem o programa faz;
- ▶ a documentação de uma função é um pequeno manual que dá instruções precisas e completas sobre o uso da função;
- ▶ uma boa documentação não se preocupa em explicar *como* uma função faz o que faz, mas sim *o que* ela faz de fato, informando quais são os valores de entrada da função, quais são os valores de saída, quais as relações que esses valores que entram e saem da função e as transformações pela função realizadas.

Segundo o professor Paulo Feofiloff, uma boa documentação de um programa significa:

- ▶ inserir comentários apropriados no código de modo a explicar o *que* cada uma das funções que compõem o programa faz;
- ▶ a documentação de uma função é um pequeno manual que dá instruções precisas e completas sobre o uso da função;
- ▶ uma boa documentação não se preocupa em explicar *como* uma função faz o que faz, mas sim *o que* ela faz de fato, informando quais são os valores de entrada da função, quais são os valores de saída, quais as relações que esses valores que entram e saem da função e as transformações pela função realizadas.

Segundo o professor Paulo Feofiloff, uma boa documentação de um programa significa:

- ▶ inserir comentários apropriados no código de modo a explicar *o que* cada uma das funções que compõem o programa faz;
- ▶ a documentação de uma função é um pequeno manual que dá instruções precisas e completas sobre o uso da função;
- ▶ uma boa documentação não se preocupa em explicar *como* uma função faz o que faz, mas sim *o que* ela faz de fato, informando quais são os valores de entrada da função, quais são os valores de saída, quais as relações que esses valores que entram e saem da função e as transformações pela função realizadas.

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

# Este programa faz a adição de dois números inteiros
# fixos e mostra o resultado da operação na saída.

num1 = 25
num2 = 30
soma = num1 + num2
print("A soma de %d e %d é %d"% (num1, num2, soma))

exit(0)
```



- ▶ Já sabemos como escrever algo na tela (**saída**)
- ▶ Contudo, é muito comum queremos obter dados do usuário através do teclado (**entrada**)

- ▶ Já sabemos como escrever algo na tela (**saída**)
- ▶ Contudo, é muito comum queremos obter dados do usuário através do teclado (**entrada**)

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

# Recebe (lê do teclado) dois números inteiros e imprime sua soma

num1 = input("Informe um número: ")
num2 = input("Informe outro número: ")
soma = num1 + num2

print("A soma de %d mais %d é %d" % (num1, num2, soma))

exit(0)
```

```
num1 = input("Informe um número: ")
```

- ▶ a **função de leitura** `input`, que recebe opcionalmente uma mensagem ao usuário, lê e devolve um **texto** (*string*, do tipo *str*)
- ▶ podemos converter o texto para um outro tipo apropriado:  

```
num1 = int(num1)
```
- ▶ podemos também fazer isso diretamente antes de armazenar algo na variável (p. ex. 

```
num1 = int(input())
```

)

```
num1 = input("Informe um número: ")
```

- ▶ a **função de leitura** `input`, que recebe opcionalmente uma mensagem ao usuário, lê e devolve um **texto** (*string*, do tipo *str*)
- ▶ podemos converter o texto para um outro tipo apropriado:

```
num1 = int(num1)
```

- ▶ podemos também fazer isso diretamente antes de armazenar algo na variável (p. ex. `num1 = int(input())`)

```
num1 = input("Informe um número: ")
```

- ▶ a **função de leitura** `input`, que recebe opcionalmente uma mensagem ao usuário, lê e devolve um **texto** (*string*, do tipo *str*)
- ▶ podemos converter o texto para um outro tipo apropriado:  

```
num1 = int(num1)
```
- ▶ podemos também fazer isso diretamente antes de armazenar algo na variável (p. ex. 

```
num1 = int(input())
```

)

```
num1 = int(input("Informe um número: "))
```

- ▶ o valor devolvido deve ser armazenado em alguma variável

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

# Recebe (lê do teclado) dois números inteiros e imprime sua soma

num1 = int(input("Informe um número: "))
num2 = int(input("Informe outro número: "))
soma = num1 + num2

print("A soma de %d mais %d é %d" % (num1, num2, soma))

exit(0)
```



# Simulação passo a passo da execução de programas

```
01  #!/usr/bin/env python
02  # -*- coding: utf-8 -*-
03
04  num1 = int(input("Informe um número: "))
05  num2 = int(input("Informe outro número: "))
06  produto = num1 * num2
07  print("O produto de %d por %d é %d" % (num1, num2, produto))
08
09  exit(0)
```

1. Escreva um programa que mostre a mensagem "Alô Mundo" na tela.
2. Escreva um programa que recebe um número digitado pelo usuário e mostra a seguinte mensagem: O número informado foi [número].
3. Escreva um programa que recebe dois números digitados pelo usuário e imprime a soma desses dois números.
4. Escreva um programa que recebe as 4 notas bimestrais de um aluno e mostra a média dessas notas.
5. Escreva um programa que recebe o raio de um círculo, calcula e mostra o valor da área desse círculo.

1. Escreva um programa que mostre a mensagem "Alô Mundo" na tela.
2. Escreva um programa que recebe um número digitado pelo usuário e mostra a seguinte mensagem: O número informado foi [número].
3. Escreva um programa que recebe dois números digitados pelo usuário e imprime a soma desses dois números.
4. Escreva um programa que recebe as 4 notas bimestrais de um aluno e mostra a média dessas notas.
5. Escreva um programa que recebe o raio de um círculo, calcula e mostra o valor da área desse círculo.

1. Escreva um programa que mostre a mensagem "Alô Mundo" na tela.
2. Escreva um programa que recebe um número digitado pelo usuário e mostra a seguinte mensagem: O número informado foi [número].
3. Escreva um programa que recebe dois números digitados pelo usuário e imprime a soma desses dois números.
4. Escreva um programa que recebe as 4 notas bimestrais de um aluno e mostra a média dessas notas.
5. Escreva um programa que recebe o raio de um círculo, calcula e mostra o valor da área desse círculo.

1. Escreva um programa que mostre a mensagem "Alô Mundo" na tela.
2. Escreva um programa que recebe um número digitado pelo usuário e mostra a seguinte mensagem: O número informado foi [número].
3. Escreva um programa que recebe dois números digitados pelo usuário e imprime a soma desses dois números.
4. Escreva um programa que recebe as 4 notas bimestrais de um aluno e mostra a média dessas notas.
5. Escreva um programa que recebe o raio de um círculo, calcula e mostra o valor da área desse círculo.

1. Escreva um programa que mostre a mensagem "Alô Mundo" na tela.
2. Escreva um programa que recebe um número digitado pelo usuário e mostra a seguinte mensagem: O número informado foi [número].
3. Escreva um programa que recebe dois números digitados pelo usuário e imprime a soma desses dois números.
4. Escreva um programa que recebe as 4 notas bimestrais de um aluno e mostra a média dessas notas.
5. Escreva um programa que recebe o raio de um círculo, calcula e mostra o valor da área desse círculo.

6. Qual é a saída esperada do programa a seguir?

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

print("Alô! ")
print()
print("Tem\ndalguém\ndai?")
print("\n")
print("?")

exit(0)
```

7. Escreva um programa que subtraia 14 de 73 e mostre o resultado na saída padrão com uma mensagem apropriada. Faça a simulação passo a passo da execução do programa.
8. Escreva um programa que leia três números inteiros  $a$ ,  $b$  e  $c$ , calcule  $a * b + c$  e mostre o resultado na saída padrão para o(a) usuário(a). Faça a simulação passo a passo da execução do programa.



7. Escreva um programa que subtraia 14 de 73 e mostre o resultado na saída padrão com uma mensagem apropriada. Faça a simulação passo a passo da execução do programa.
8. Escreva um programa que leia três números inteiros  $a$ ,  $b$  e  $c$ , calcule  $a * b + c$  e mostre o resultado na saída padrão para o(a) usuário(a). Faça a simulação passo a passo da execução do programa.

9. Escreva um programa que leia um número inteiro e mostre o seu quadrado e seu cubo. Por exemplo, se o número de entrada é 3, a saída deve ser 9 e 27. Faça a simulação passo a passo da execução do programa.
10. Escreva um programa que leia três números inteiros e mostre como resultado a soma desses três números e também a multiplicação desses três números. Faça a simulação passo a passo da execução do programa.
11. Escreva um programa que leia um número inteiro e mostre o resultado do quociente da divisão desse número por 2 e por 3. Faça a simulação passo a passo da execução do programa.

9. Escreva um programa que leia um número inteiro e mostre o seu quadrado e seu cubo. Por exemplo, se o número de entrada é 3, a saída deve ser 9 e 27. Faça a simulação passo a passo da execução do programa.
10. Escreva um programa que leia três números inteiros e mostre como resultado a soma desses três números e também a multiplicação desses três números. Faça a simulação passo a passo da execução do programa.
11. Escreva um programa que leia um número inteiro e mostre o resultado do quociente da divisão desse número por 2 e por 3. Faça a simulação passo a passo da execução do programa.

9. Escreva um programa que leia um número inteiro e mostre o seu quadrado e seu cubo. Por exemplo, se o número de entrada é 3, a saída deve ser 9 e 27. Faça a simulação passo a passo da execução do programa.
10. Escreva um programa que leia três números inteiros e mostre como resultado a soma desses três números e também a multiplicação desses três números. Faça a simulação passo a passo da execução do programa.
11. Escreva um programa que leia um número inteiro e mostre o resultado do quociente da divisão desse número por 2 e por 3. Faça a simulação passo a passo da execução do programa.

12. Escreva um programa que pergunte quanto você ganha por hora e o número de horas trabalhadas por mês. Seu programa deve calcular e mostrar o total do seu salário no referido mês.
13. Escreva um programa que recebe uma temperatura em graus Celsius, transforma e mostra a temperatura em graus Fahrenheit.
14. Escreva um programa que recebe uma temperatura em graus Fahrenheit, transforma e mostra a temperatura em graus Celsius.

12. Escreva um programa que pergunte quanto você ganha por hora e o número de horas trabalhadas por mês. Seu programa deve calcular e mostrar o total do seu salário no referido mês.
13. Escreva um programa que recebe uma temperatura em graus Celsius, transforma e mostra a temperatura em graus Fahrenheit.
14. Escreva um programa que recebe uma temperatura em graus Fahrenheit, transforma e mostra a temperatura em graus Celsius.

12. Escreva um programa que pergunte quanto você ganha por hora e o número de horas trabalhadas por mês. Seu programa deve calcular e mostrar o total do seu salário no referido mês.
13. Escreva um programa que recebe uma temperatura em graus Celsius, transforma e mostra a temperatura em graus Fahrenheit.
14. Escreva um programa que recebe uma temperatura em graus Fahrenheit, transforma e mostra a temperatura em graus Celsius.

15. Escreva um programa que pergunte quanto você ganha por hora e o número de horas trabalhadas por mês. Seu programa deve calcular e mostrar o total do seu salário no referido mês.
16. Tendo como dado de entrada a altura (h) de uma pessoa, escreva um programa que mostre o peso recomendado para essa pessoa utilizando as seguintes fórmulas:
  - ▶ Homens:  $(72.7 * h) - 58$
  - ▶ Mulheres:  $(62.1 * h) - 44.7$
17. Escreva um programa que recebe o tamanho de um arquivo para download (em MB) e a velocidade de um link de internet (em Mbps), calcula e informa o tempo aproximado de download do arquivo usando este link (em minutos).



15. Escreva um programa que pergunte quanto você ganha por hora e o número de horas trabalhadas por mês. Seu programa deve calcular e mostrar o total do seu salário no referido mês.
16. Tendo como dado de entrada a altura (h) de uma pessoa, escreva um programa que mostre o peso recomendado para essa pessoa utilizando as seguintes fórmulas:
  - ▶ Homens:  $(72.7 * h) - 58$
  - ▶ Mulheres:  $(62.1 * h) - 44.7$
17. Escreva um programa que recebe o tamanho de um arquivo para download (em MB) e a velocidade de um link de internet (em Mbps), calcula e informa o tempo aproximado de download do arquivo usando este link (em minutos).

15. Escreva um programa que pergunte quanto você ganha por hora e o número de horas trabalhadas por mês. Seu programa deve calcular e mostrar o total do seu salário no referido mês.
16. Tendo como dado de entrada a altura (h) de uma pessoa, escreva um programa que mostre o peso recomendado para essa pessoa utilizando as seguintes fórmulas:
  - ▶ Homens:  $(72.7 * h) - 58$
  - ▶ Mulheres:  $(62.1 * h) - 44.7$
17. Escreva um programa que recebe o tamanho de um arquivo para download (em MB) e a velocidade de um link de internet (em Mbps), calcula e informa o tempo aproximado de download do arquivo usando este link (em minutos).

18. (Extensão do exercício 15) Faça um programa que pergunta quanto você ganha por hora e o número de horas trabalhadas no mês. Calcule e mostre o total do seu salário no referido mês, sabendo-se que são descontados 11% para o imposto de renda, 8% para o INSS e 5% para o sindicato. Seu programa deve mostrar para o usuário as seguintes informações:
- ▶ Salário bruto;
  - ▶ Quanto pagou ao INSS;
  - ▶ Quando pagou ao sindicato;
  - ▶ Salário líquido.
19. Faça um programa para uma loja de tintas. O programa deverá pedir o tamanho em metros quadrados da área a ser pintada. Considere que a cobertura da tinta é de 1 litro para cada 3 metros quadrados e que a tinta é vendida em latas de 18 litros, que custam R\$ 80,00. Informe ao usuário a quantidades de latas de tinta a serem compradas e o preço total.

18. (Extensão do exercício 15) Faça um programa que pergunta quanto você ganha por hora e o número de horas trabalhadas no mês. Calcule e mostre o total do seu salário no referido mês, sabendo-se que são descontados 11% para o imposto de renda, 8% para o INSS e 5% para o sindicato. Seu programa deve mostrar para o usuário as seguintes informações:
- ▶ Salário bruto;
  - ▶ Quanto pagou ao INSS;
  - ▶ Quando pagou ao sindicato;
  - ▶ Salário líquido.
19. Faça um programa para uma loja de tintas. O programa deverá pedir o tamanho em metros quadrados da área a ser pintada. Considere que a cobertura da tinta é de 1 litro para cada 3 metros quadrados e que a tinta é vendida em latas de 18 litros, que custam R\$ 80,00. Informe ao usuário a quantidades de latas de tinta a serem compradas e o preço total.

20. (Extensão do exercício 19) Faça um programa para uma loja de tintas. O programa deverá pedir o tamanho em metros quadrados da área a ser pintada. Considere que a cobertura da tinta é de 1 litro para cada 3 metros quadrados e que a tinta é vendida em latas de 18 litros (que custam R\$ 80,00) ou em galões de 3,6 litros (que custam R\$ 25,00). Informe ao usuário as quantidades de tinta a serem compradas e os respectivos preços em 3 situações:
- ▶ Comprar apenas latas de 18 litros;
  - ▶ Comprar apenas galões de 3,6 litros;
  - ▶ Misturar latas e galões, de forma que o preço seja o menor. Acrescente 10% de folga e sempre arredonde os valores para cima, isto é, considere latas cheias.

21. Uma pessoa resolver fazer uma aplicação em uma poupança programada. Para calcular seu rendimento, ela deverá fornecer o valor constante da aplicação mensal, a taxa e o número de meses. A fórmula usada para o cálculo do valor acumulado é a seguinte:

$$\text{Valor acumulado} = \frac{P \times (1+i)^n - 1}{i}$$

Onde  $i$  = taxa,  $P$  = aplicação mensal e  $n$  é o número de meses.

22. Faça um programa que após a entrada de uma determinada distância entre dois pontos (em Km) e uma determinada velocidade (Km/h), diga qual o tempo médio que levará para chegar a esse local e qual a velocidade em metros/segundos.

21. Uma pessoa resolver fazer uma aplicação em uma poupança programada. Para calcular seu rendimento, ela deverá fornecer o valor constante da aplicação mensal, a taxa e o número de meses. A fórmula usada para o cálculo do valor acumulado é a seguinte:

$$\text{Valor acumulado} = \frac{P \times (1+i)^n - 1}{i}$$

Onde  $i$  = taxa,  $P$  = aplicação mensal e  $n$  é o número de meses.

22. Faça um programa que após a entrada de uma determinada distância entre dois pontos (em Km) e uma determinada velocidade (Km/h), diga qual o tempo médio que levará para chegar a esse local e qual a velocidade em metros/segundos.

23. O custo de um carro novo ao consumidor é a soma do custo de fábrica com a percentagem do distribuidor e dos impostos (aplicados, primeiro os impostos sobre o custo de fábrica, e depois a percentagem do distribuidor sobre o resultado). Supondo que a percentagem do distribuidor seja de 28% e os impostos 45%, escreva um programa que leia o custo de fábrica de um carro e informa o custo final ao consumidor.
24. Elabore um algoritmo e leia do teclado uma quantidade de segundos e transforme esse tempo em dias, horas e minutos.



23. O custo de um carro novo ao consumidor é a soma do custo de fábrica com a percentagem do distribuidor e dos impostos (aplicados, primeiro os impostos sobre o custo de fábrica, e depois a percentagem do distribuidor sobre o resultado). Supondo que a percentagem do distribuidor seja de 28% e os impostos 45%, escreva um programa que leia o custo de fábrica de um carro e informa o custo final ao consumidor.
24. Elabore um algoritmo e leia do teclado uma quantidade de segundos e transforme esse tempo em dias, horas e minutos.

25. Um hotel deseja fazer uma promoção especial de final de semana, concedendo um desconto de 25% na diária. Sendo informados, através do teclado, o número de apartamentos do hotel e o valor da diária por apartamento para o final de semana completo, elabore um programa para calcular:
- ▶ Valor promocional da diária;
  - ▶ Valor total a ser arrecadado caso a ocupação neste final de semana atinja 100%;
  - ▶ Valor total a ser arrecadado caso a ocupação neste final de semana atinja 70%;
  - ▶ Valor que o hotel deixará de arrecadar em virtude da promoção, caso a ocupação atinja 100%.
26. Elabore um algoritmo para efetuar o cálculo da quantidade de combustível gasto em uma viagem, utilizando-se um automóvel que faz 12 Kms por litro. Para obter o cálculo, o usuário deverá fornecer o tempo gasto e a velocidade média durante a viagem.

25. Um hotel deseja fazer uma promoção especial de final de semana, concedendo um desconto de 25% na diária. Sendo informados, através do teclado, o número de apartamentos do hotel e o valor da diária por apartamento para o final de semana completo, elabore um programa para calcular:
- ▶ Valor promocional da diária;
  - ▶ Valor total a ser arrecadado caso a ocupação neste final de semana atinja 100%;
  - ▶ Valor total a ser arrecadado caso a ocupação neste final de semana atinja 70%;
  - ▶ Valor que o hotel deixará de arrecadar em virtude da promoção, caso a ocupação atinja 100%.
26. Elabore um algoritmo para efetuar o cálculo da quantidade de combustível gasto em uma viagem, utilizando-se um automóvel que faz 12 Kms por litro. Para obter o cálculo, o usuário deverá fornecer o tempo gasto e a velocidade média durante a viagem.