

15.10



PYTHON - REVISÃO

Bruno Drumond | Mariana Duarte | Patrícia Alves | Pedro Moreira

Cronograma

- ESTRUTURAS DE CONDIÇÃO
- ESTRUTURAS DE REPETIÇÃO
- LISTAS
- DICIONÁRIOS
- FUNÇÕES
- P.O.O

PYTHON

A LINGUAGEM
EM SI

ESQUEÇA OS
MITOS

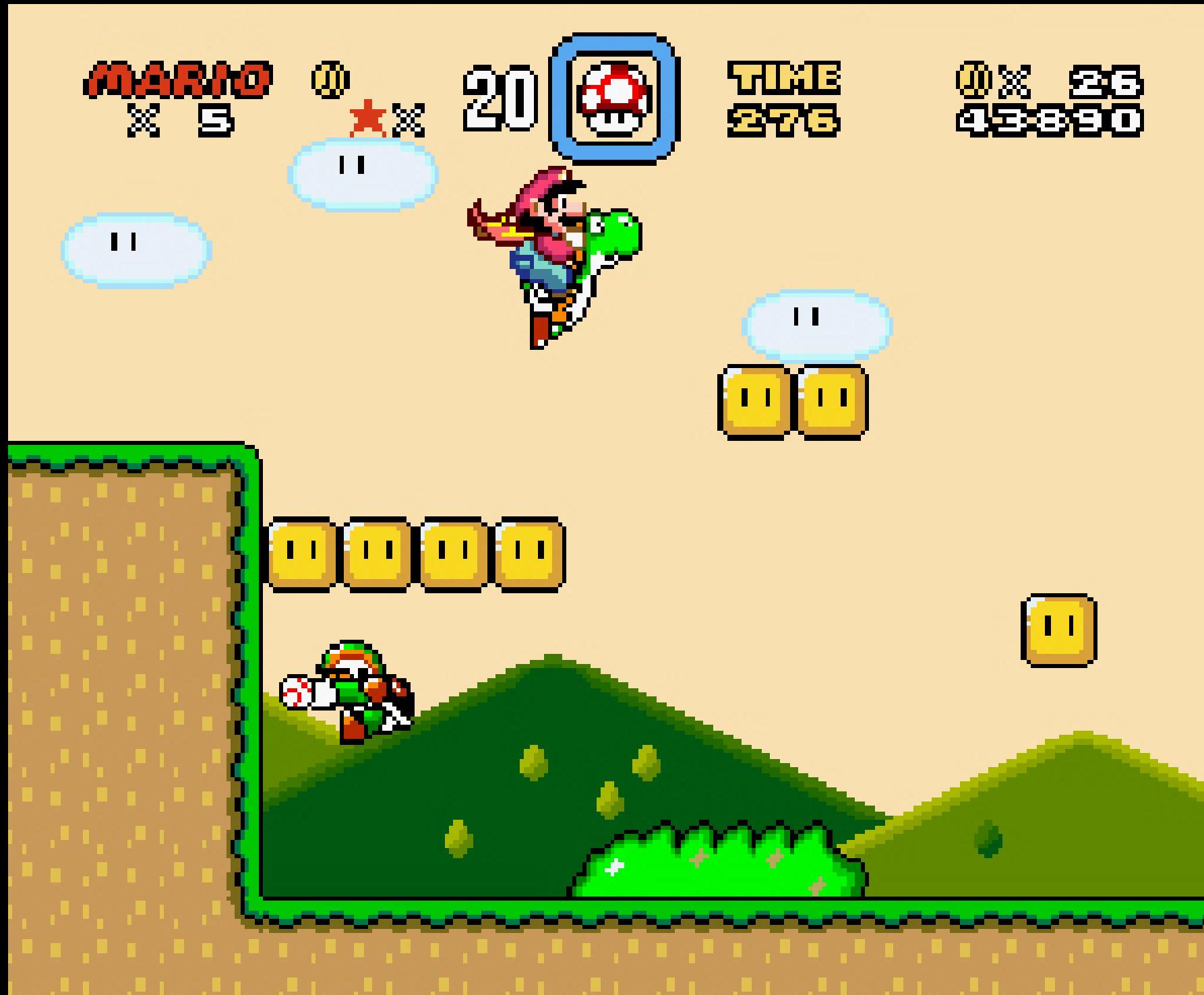
PAPEL ANTES
DO
COMPUTADOR



```
print("Hello Super Mario World")
```

Estruturas de Condição | Repetição

IF | ELIF | ELSE | WHILE | FOR



SUPER MARIO CLASS

Condição / Repetição

Objetivo escolher um herói e seus movimentos sem quebrar o código

- estrutura de condição:
if / elif / else
- estrutura de repetição
while / for

MARIO
X 5



20



TIME
276

0 X 26
43890

Vamos praticar ?

PRACTICE LEADS TO PERFECTION

EXERCÍCIO- IF

A prefeitura de Cafundó do Judas abriu uma linha de crédito para os funcionários. O valor máximo da prestação não poderá ultrapassar 30% do salário bruto. Fazer um algoritmo onde o usuário informa seu salário bruto e o valor da prestação e o programa exibe na tela se o empréstimo pode ou não ser concedido.

EXERCÍCIO - WHILE | FOR

Em uma competição de ginástica, cada atleta recebe votos de sete jurados. A melhor e a pior nota são eliminadas. A nota do atleta fica sendo a média dos votos restantes. Você deve fazer um programa que receba o nome do ginasta e as notas dos sete jurados alcançadas pelo atleta em sua apresentação. Seu programa deve exibir a média, conforme a descrição acima (retirar o melhor e o pior salto e depois calcular a média com as notas restantes). As notas não são informadas ordenadas.

Listas

```
if ($(window).scrollTop() > header1_initialDistance) {  
    header1.css('padding-top', '' + $(window).scrollTop() - header1_initialPadding);  
}  
} else {  
    header1.css('padding-top', '' + header1_initialPadding);  
}  
  
if ($(window).scrollTop() > header2_initialDistance) {  
    if (list != null) {  
        if (parseInt(header2.css('padding-top'), 10) == header2_initialPadding) {  
            header2.css('padding-top', '' + $(window).scrollTop() - header2_initialPadding);  
        }  
    }  
} else {  
    header2.css('padding-top', '' + header2_initialPadding);  
}
```

item



item2



itens



Logradouro/Nome

Rua Alfredo Backer

Bairro/Distrito

Centro

Localidade/UF

Macaé/RJ

CEP

27910-190





Vamos praticar?

PRACTICE LEADS TO PERFECTION

EXERCÍCIO

Receba números do usuário e armazene em uma lista. Calcule a média desses números. Crie uma segunda lista contendo os números abaixo da média e uma terceira lista com os números acima da média. Exiba todas as três listas.

Dicionários

chaves valores items

```
dic_mundos_mario = {"mundo 1": "Yoshi's Island",
                     "mundo 2": "Donut Plains",
                     "mundo 3": "Vanilla Dome",
                     "mundo 4": "Twins Bridges",
                     "mundo 5": "Forest of Illusion",
                     "mundo 6": "Chocolate Island",
                     "mundo 7": "Valley of Bowser"
                    }
```

Vamos praticar ?



PRACTICE LEADS TO PERFECTION

EXERCÍCIO

Crie um dicionário e receba do usuário os dados para as seguintes: nome, idade, email, profissão, cidade e telefone. Faça um for para exibir cada um desses valores, inclusive as chaves.

header1.Funções

```
def ():
```

**Imagine se você precisasse
desenhar todos esses blocos
amarelos várias vezes, que
mundo cruel seria!**

**Para isso servem as funções,
você poderá invocá-la para
desenhar os blocos para você.**





Vamos praticar?

PRACTICE LEADS TO PERFECTION

EXERCÍCIO - FUNÇÕES

Crie uma função que recebe o salário bruto de um empregado e retorne o salário líquido. Sobre o salário bruto são feitos os seguintes descontos:

INSS: 9%

Vale Transporte: 6%

Imposto de renda: varia de acordo com a seguinte tabela

Até R\$ 1.903,98: isento

De R\$ 1.903,99 até R\$ 2.826,65: 7,5%

De R\$ 2.826,66 até R\$ 3.751,05: 15%

De R\$ 3.751,06 até R\$ 4.664,68: 22,5%

Acima de R\$ 4.664,68: 27,5%

Crie um programa que recebe do usuário o salário bruto, chame a função acima e exiba o salário líquido.

```
if ($(window).scrollTop() > header1_initialDistance) {  
    if (parseInt(header1.css('padding-top'), 10) < header1_initialPadding) {  
        header1.css('padding-top', '' + $(window).scrollTop() -  
    }  
} else {  
    header1.css('padding-top', '' + header1_initialPadding + 'px');  
}  
  
if ($(window).scrollTop() > header2_initialDistance) {  
    if (parseInt(header2.css('padding-top'), 10) < header2_initialPadding) {  
        header2.css('padding-top', '' + $(window).scrollTop() -  
    }  
} else {  
    header2.css('padding-top', '' + header2_initialPadding + 'px');
```

CONCEITOS

CLASSES

As classes são como um Molde de criação, esse molde possui características e funcionalidades

OBJETOS

Os objetos são cada criação de uma variável cujo tipo é uma classe, no qual chamamos de instânciação

MÉTODOS

Funções (ações) da classe

ATRIBUTOS

Variáveis (características) da classe

Classe Caneta:

Atributos:

cor

marca

ponta

Metodos:

tampar()

destampar()

escrever()

rabiscar()





Classe Personagem:

Atributos:

cor

nome

moedas = 0

Metodos:

andar()

pular()

pegar_moedas()

pegar_item()

Classe Personagem:

#Atributos do Herói

nome = ''

cor = ''

moedas = 0

#Ações do Herói

def pegar_item(self,item):

def pegar_moeda(self):

def andar():

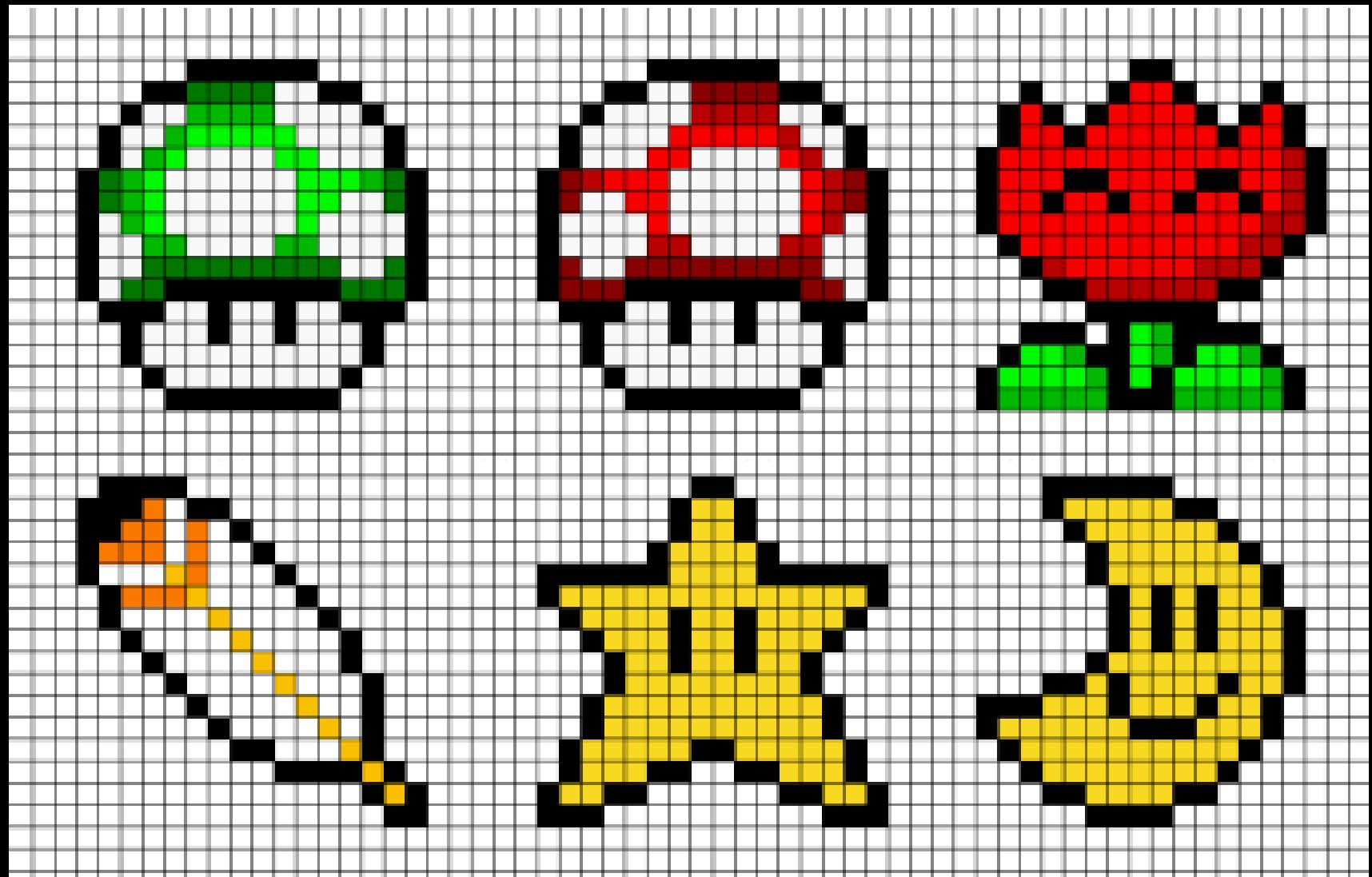
def pular ():

#Instanciando um Objeto

p = Personagem()

p.nome("Mario")

p.cor("Vermelho")



CONSTRUTOR

Função-metodo executada automaticamente na criação do objeto, é declarado com `def __init__`

SELF

É definido automaticamente para referenciar o objeto recém-criado que precisa ser inicializado

HERANÇA

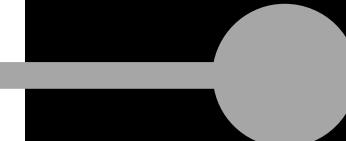
Quando uma classe herda as propriedades (métodos e atributos) de outra classe.

POLIMORFISMO

Permite sobrescrever determinados métodos quando temos herança.

ENCAPSULAMENTO

São utilizados para proteção dos atributos ou métodos de uma classe





Classe ContaBancaria:

Construtor:

def

__init__(self,conta,titular,saldo):

Metodos:

sacar()

depositar()

extrato()

**Classe ContaPoupança(
ContaBancaria):**

**Classe ContaCorrente(
ContaBancaria):**



Vamos praticar?

PRACTICE LEADS TO PERFECTION

EXERCÍCIO 1

Crie uma classe denominada Pessoa, com os atributos nome, idade e email. Seu programa principal deve criar 2 objetos do tipo pessoa e preenche-los com dados do usuário. Exiba todos os dados na tela.

EXERCÍCIO 2

Crie uma classe Produto com os atributos nome, preço e estoque. Seu programa principal deve ter uma lista do tipo produto. Crie um menu com opções para: inserir, excluir, procurar e sair.

EXERCÍCIO 3

Crie uma classe chamada Triangulo (com base, altura e um método area) e uma classe chamada Círculo (com raio e um método area). Faça o usuário preencher uma lista com 5 Triângulos e outra com 5 círculo. Exiba na tela o triângulo e o círculo com maior área