



Programação em Python

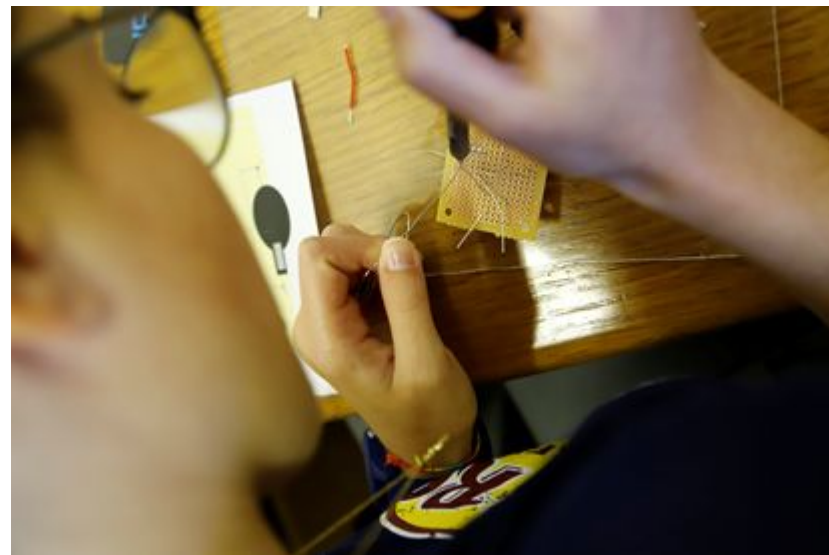
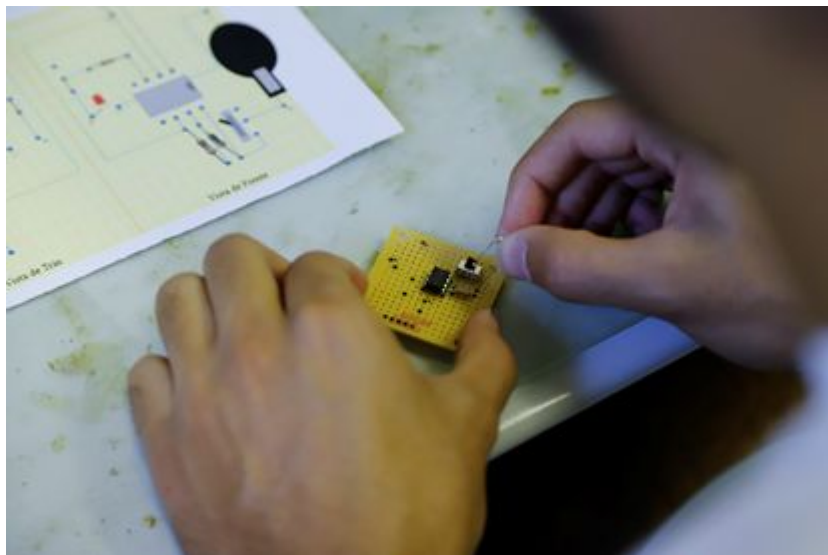
4 Dezembro 2019

Quem somos?

- ❑ Núcleo de alunos de MEEC do IST;
- ❑ Associação sem fins lucrativos;
- ❑ Desde 19 de setembro de 2003.



Receção aos novos alunos



Workshop de construção de um Blinker (em colaboração com a CPMEEC)

O que fazemos?

Workshops



Eventos



NEECathon, apoio nas JEEC, Open Days em sedes de empresas...

O que fazemos?

- ❑ NEECLab;
- ❑ Drive NEEC;
- ❑ Apoio ao Aluno;
- ❑ IST Summer Internships;
- ❑ Contacto empresarial.



É uma **linguagem de programação de alto nível** criada por Guido van Rossum em 1991;

Algumas características:

- ❑ Simples, elegante e minimalista;
- ❑ Extensível (C,Java,Fortran,.Net);
- ❑ Orientado a objetos;
- ❑ Gratuito e Open Source;
- ❑ Inúmeras bibliotecas;



Variáveis

Existem vários tipos de **variáveis**, por exemplo:

- ❑ Números;
- ❑ Listas;
- ❑ Strings;
- ❑ Tuples;
- ❑ Dicionários;



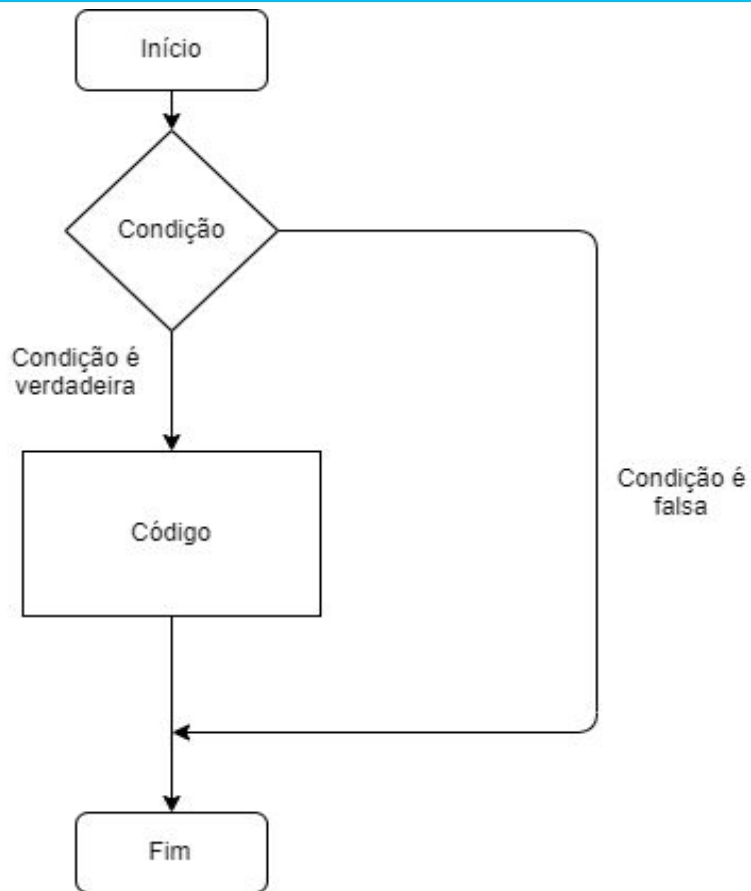
Podemos pensar nelas como caixas onde se guardam dados.

Condições

❏ **If:**

❏ **Elif:**

❏ **Else:**



Condições

```
1  a = 100
2  b = 10
3
4  if a > b:
5      print("a é maior que b")
6  else:
7      print("a é menor que b")
8
9  print("Fim do programa")
```

Condições

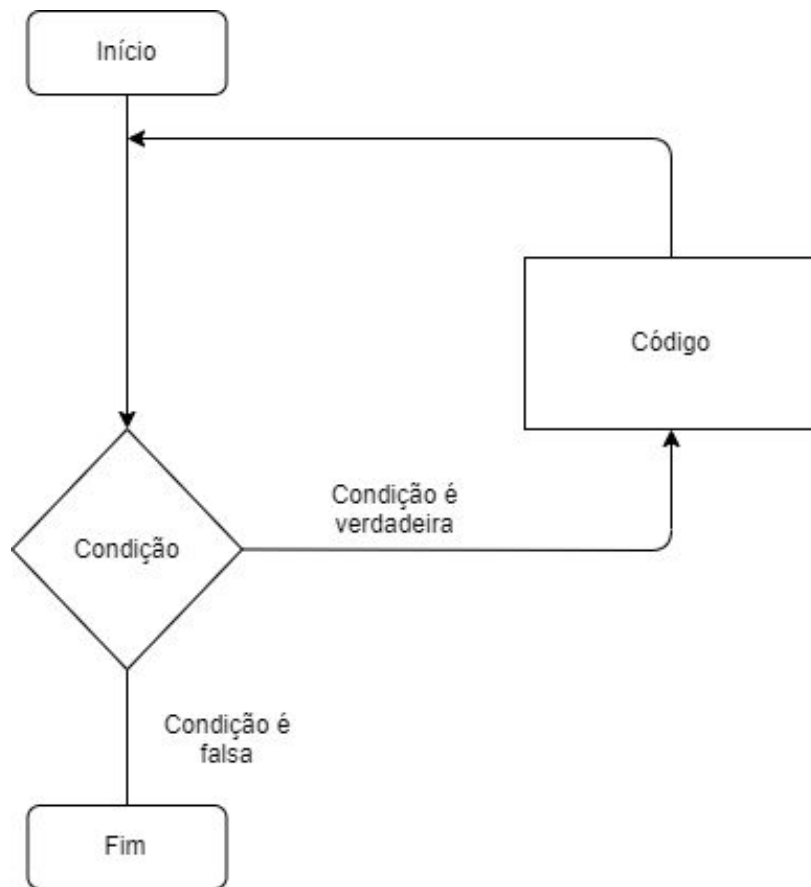
```
1  a = 100
2  b = 10
3
4  if a > b:
5      print("a é maior que b")
6
7
8
9
10
11 print("Fim do programa")
```

```
C:\Users\simao\Desktop\NEEC\Testes>py Condicoes.py
a é maior que b
Fim do programa
```

Ciclos

■ While

■ For __ in __:



Ciclos

```
1 input_do_utilizador = str()
2
3 while input_do_utilizador != 'sair' :
4     input_do_utilizador = input("Escreva sair para abandonar o programa: ")
5     print(input_do_utilizador)
6
7 print("Programa abandonado")
```

```
Escrever sair no terminal e dar enter: 1
1
Escrever sair no terminal e dar enter: 2
2
Escrever sair no terminal e dar enter: sair
sair
```

Ciclos - for

```
1 neec_membros = ["Alex", "Antunes", "Coelho", "Adrian", "Maria", "Gabriel"]
2 neec_eventos = ["Neecathon", "WS Python", "WS Matlab", "WS Git"]
3
4 for x in neec_membros:
5     print(x)
6
7 print('')
8
9 for y in range(2):
10     print(neec_eventos[y])
11
12 print("Fim do programa")
```

```
C:\Users\simao\Desktop\NEEC\Testes>py Ciclos.py
Alex
Antunes
Coelho
Adrian
Maria
Gabriel

Neecathon
WS Python
Fim do programa
```


Estrutura do Jogo

1. **Inicializações** de variáveis;
2. **Jogadas** e respectiva **verificação**;
3. Fim do jogo;
4. **Parte Gráfica**/Criação do tabuleiro.


Estrutura do Jogo - Verificações

Há que verificar se a jogada é válida e se a partir da última jogada se fez 4 em linha.

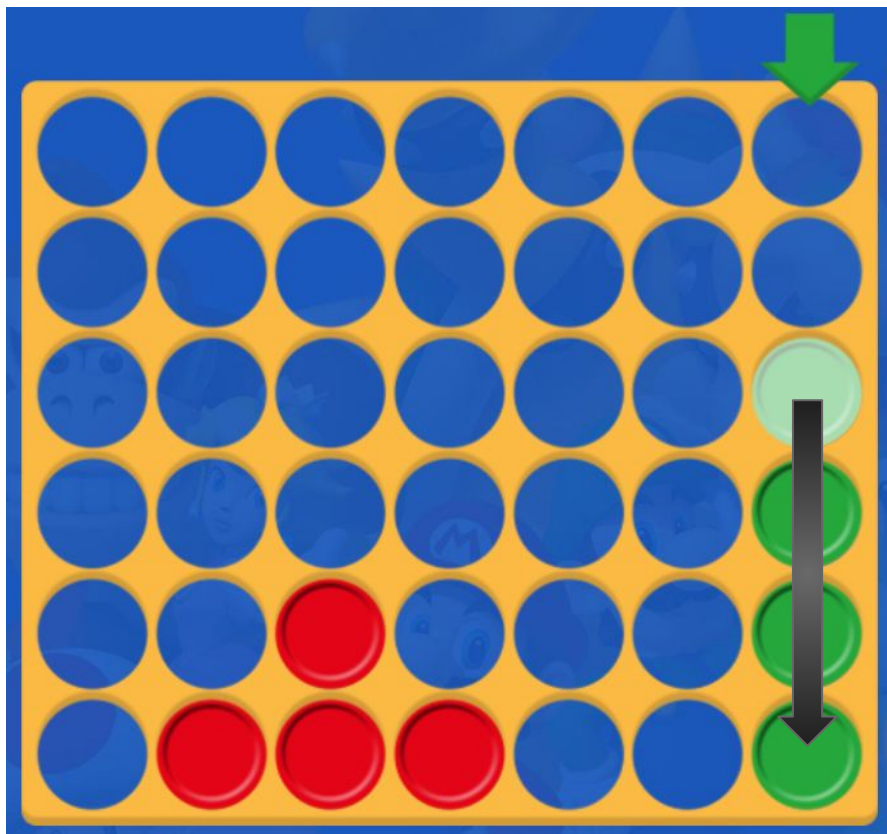
Jogada válida, há que verificar se:

- ☐ Coluna está cheia
- ☐ O espaço está ocupado por uma peça adversária

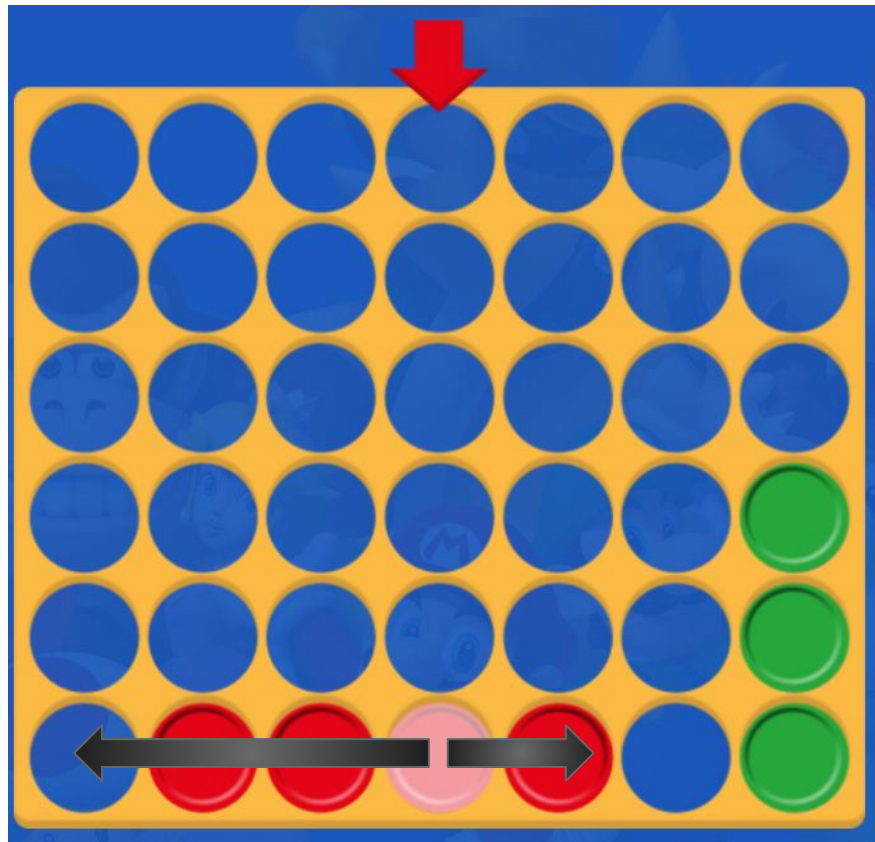
4 em linha, a partir da última peça colocada vamos verificar:

- ☐ 4 direções 
- ☐ Para cada direção, vai-se ver os dois sentidos e somar cada peça desse jogador

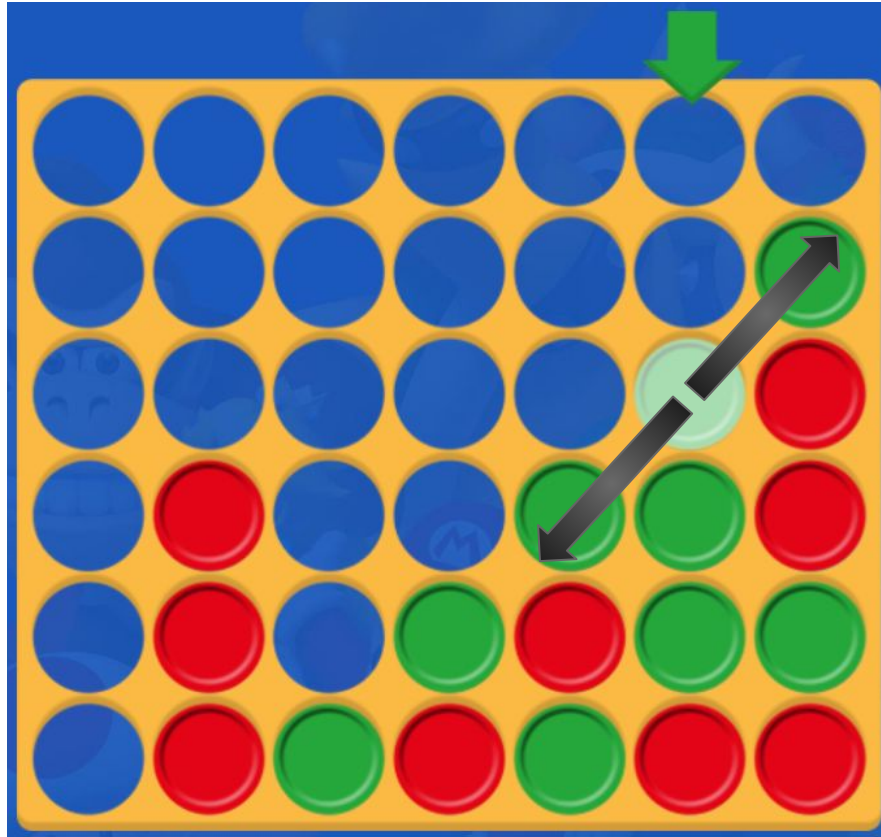
Estrutura do Jogo - Verificações



Estrutura do Jogo - Verificações



Estrutura do Jogo - Verificações



Estrutura do Jogo - Parte Gráfica

Há que criar o tabuleiro e ir atualizando-o consoante o decorrer do jogo.

O que vai acontecer aqui é que vamos sobrepondo imagens, de forma a substituir aquilo que vemos com algo atualizado.

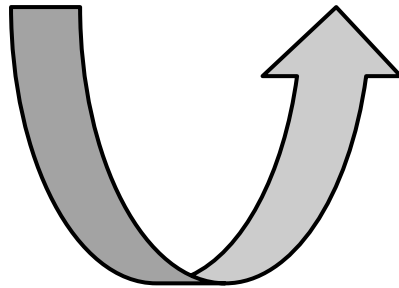


Para desenhar o tabuleiro temos que fazer algumas continhas, porque o ecrã está dividido em píxeis.

Estrutura do Jogo - Parte Gráfica

00						06
	11					
		22				
			33			
				44		
					55	
60						66

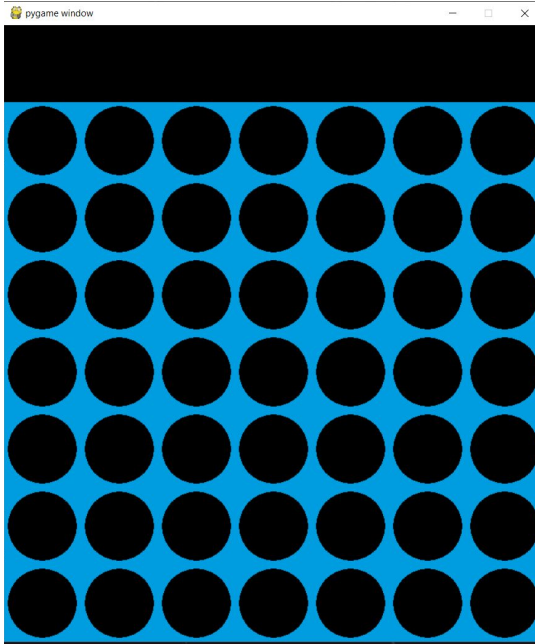
Matriz correspondente
ao **Tabuleiro analisado**
(escondido)



Matriz correspondente ao
Tabuleiro visualizado

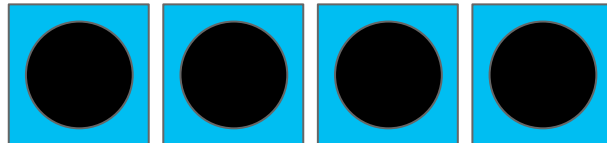
60						66
					55	
				44		
			33			
		22				
	11					
00						06

Estrutura do Jogo - Parte Gráfica

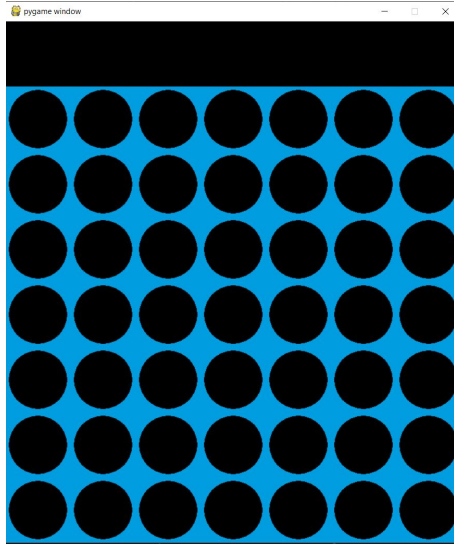


- ❑ Conseguimos desenhar formas como quadrados e círculos.
- ❑ Desenharam-se vários quadrados azuis e desenhamos círculos pretos por cima a simular espaços vazios.

Juntando vários temos o nosso tabuleiro feito!

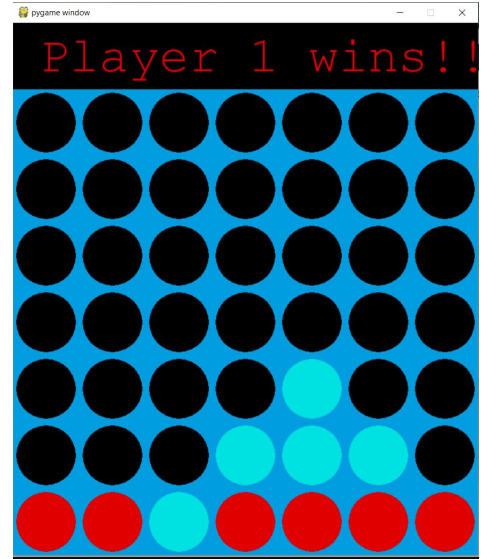


Estrutura do Jogo - Parte Gráfica



A barra de cima foi reservada para conter informações adicionais.

Tendo o primeiro tabuleiro vazio, basta agora desenhar por cima as peças de cada jogada (círculos de cores diferentes).



4 em Linha - Source Code

❏ necist.org/ws/python/4emlinha_incompleto.py

- ❑ Está atento ao nosso **Facebook**.
- ❑ Também nos podes ver no **Linkedin**.
- ❑ Podes visitar-nos no nosso **site** que te vai pôr a par de todas as novidades.
- ❑ Já estamos no **Instagram**.



Em cada ícone está a hiperligação para o respetivo site.

 necist.org/ws/python/4emlinha.py

