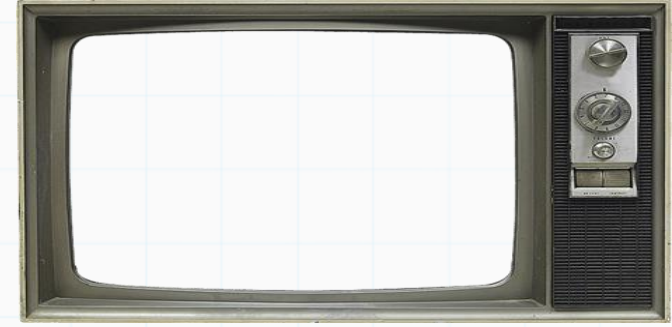


Programação De Computadores

Professor : Yuri Frota

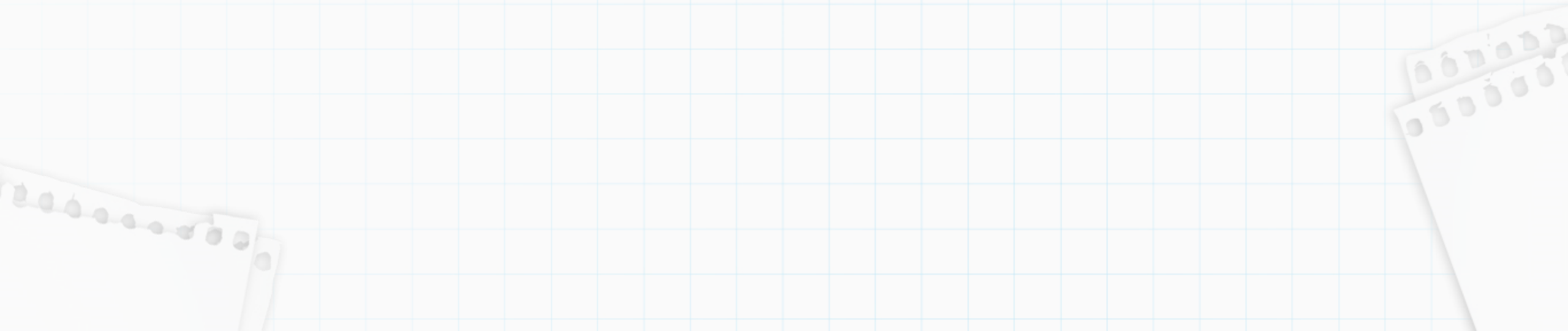
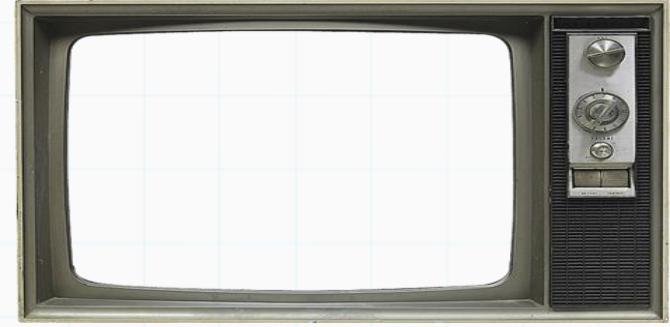
www.ic.uff.br/~yuri/prog.html

yuri@ic.uff.br



Laços Aninhados

Na literatura da língua inglesa, o termo utilizado é “*nested loops*” (laços aninhados), nada mais é do que laços encaixados, um dentro do outro.

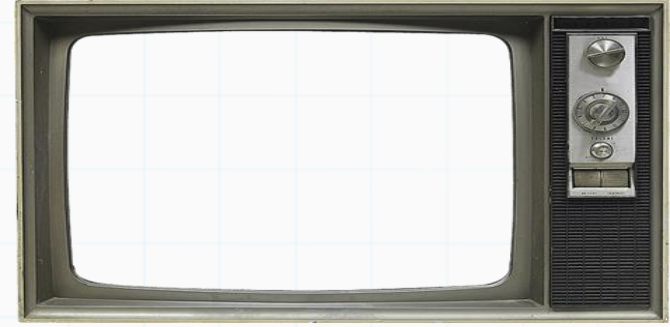


Laços Aninhados

Na literatura da língua inglesa, o termo utilizado é “*nested loops*” (laços aninhados), nada mais é do que laços encaixados, um dentro do outro.

Como funciona ?

```
for h in range (1,24+1,1):  
    for m in range (1,60+1,1):  
        for s in range (1,60+1,1):  
            print (h,m,s)
```



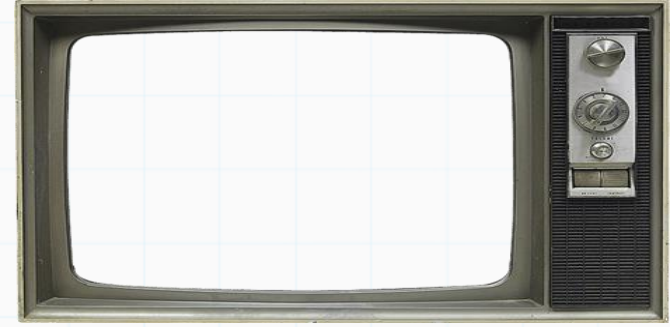
Laços Aninhados

Na literatura da língua inglesa, o termo utilizado é “*nested loops*” (laços aninhados), nada mais é do que laços encaixados, um dentro do outro.

Como funciona ?

```
for h in range (1,24+1,1):  
    for m in range (1,60+1,1):  
        for s in range (1,60+1,1):  
            print (h,m,s)
```

1 1 1



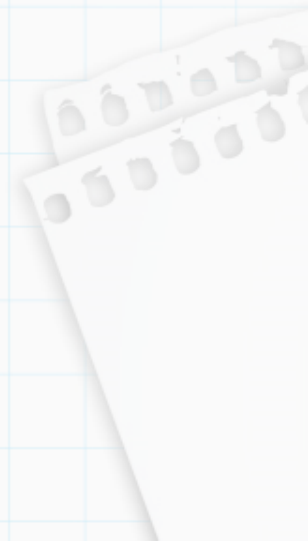
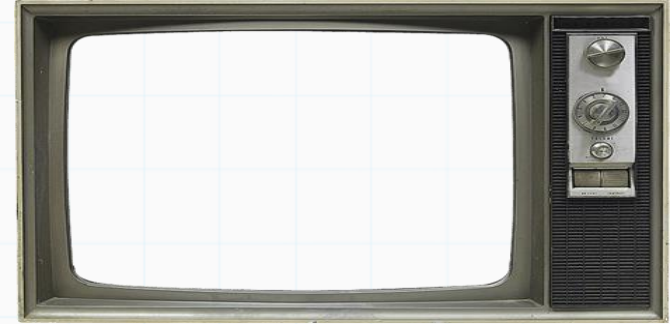
Laços Aninhados

Na literatura da língua inglesa, o termo utilizado é “*nested loops*” (laços aninhados), nada mais é do que laços encaixados, um dentro do outro.

Como funciona ?

```
for h in range (1,24+1,1):  
    for m in range (1,60+1,1):  
        for s in range (1,60+1,1):  
            print (h,m,s)
```

```
1 1 1  
1 1 2
```



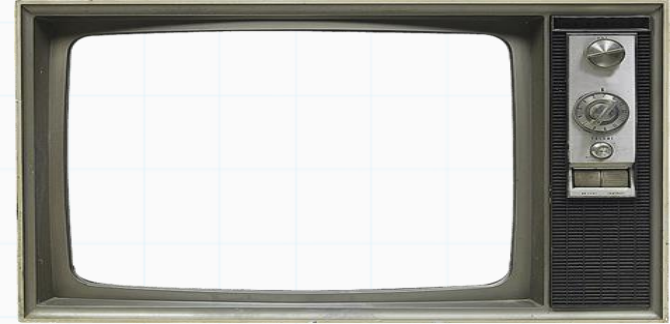
Laços Aninhados

Na literatura da língua inglesa, o termo utilizado é “*nested loops*” (laços aninhados), nada mais é do que laços encaixados, um dentro do outro.

Como funciona ?

```
for h in range (1,24+1,1):  
    for m in range (1,60+1,1):  
        for s in range (1,60+1,1):  
            print (h,m,s)
```

```
1 1 1  
1 1 2  
1 1 3  
...
```



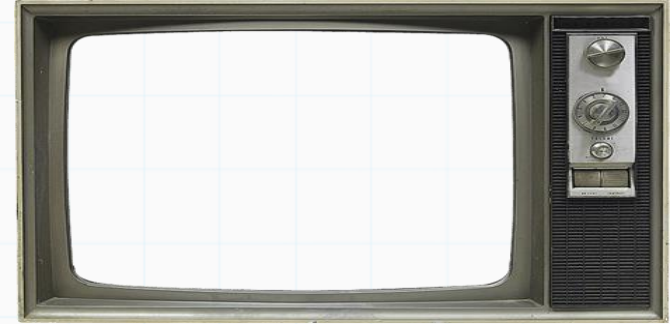
Laços Aninhados

Na literatura da língua inglesa, o termo utilizado é “*nested loops*” (laços aninhados), nada mais é do que laços encaixados, um dentro do outro.

Como funciona ?

```
for h in range (1,24+1,1):  
    for m in range (1,60+1,1):  
        for s in range (1,60+1,1):  
            print (h,m,s)
```

```
1 1 1  
1 1 2  
1 1 3  
...  
1 1 58  
1 1 59  
1 1 60
```



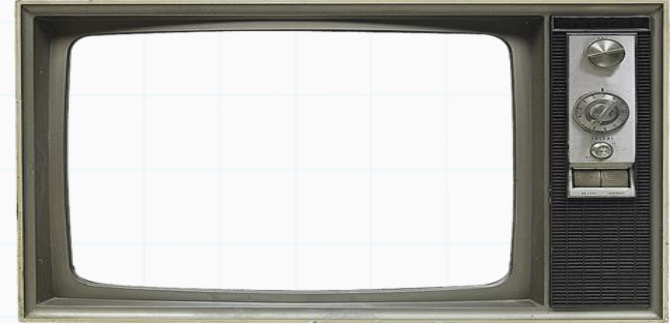
Laços Aninhados

Na literatura da língua inglesa, o termo utilizado é “*nested loops*” (laços aninhados), nada mais é do que laços encaixados, um dentro do outro.

Como funciona ?

```
for h in range (1,24+1,1):  
    for m in range (1,60+1,1):  
        for s in range (1,60+1,1):  
            print (h,m,s)
```

```
1 1 1  
1 1 2  
1 1 3  
...  
1 1 58  
1 1 59  
1 1 60
```



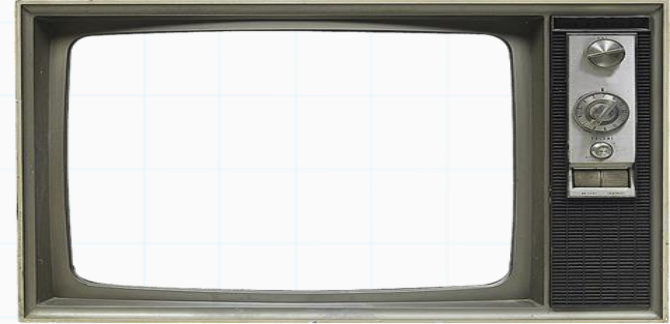
Laços Aninhados

Na literatura da língua inglesa, o termo utilizado é “*nested loops*” (laços aninhados), nada mais é do que laços encaixados, um dentro do outro.

Como funciona ?

```
for h in range (1,24+1,1):  
    for m in range (1,60+1,1):  
        for s in range (1,60+1,1):  
            print (h,m,s)
```

```
1 1 1      1 2 1  
1 1 2  
1 1 3  
...  
1 1 58  
1 1 59  
1 1 60
```



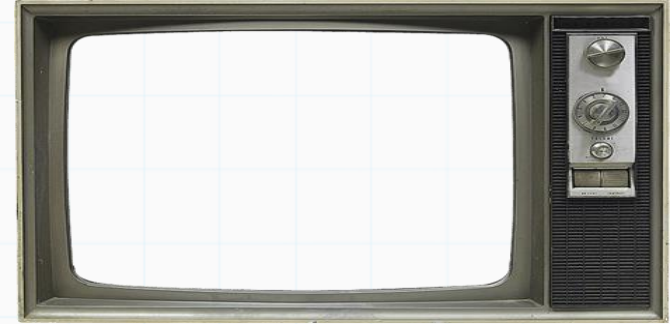
Laços Aninhados

Na literatura da língua inglesa, o termo utilizado é “*nested loops*” (laços aninhados), nada mais é do que laços encaixados, um dentro do outro.

Como funciona ?

```
for h in range (1,24+1,1):  
    for m in range (1,60+1,1):  
        for s in range (1,60+1,1):  
            print (h,m,s)
```

```
1 1 1      1 2 1  
1 1 2      1 2 2  
1 1 3      1 2 3  
...  
1 1 58  
1 1 59  
1 1 60
```



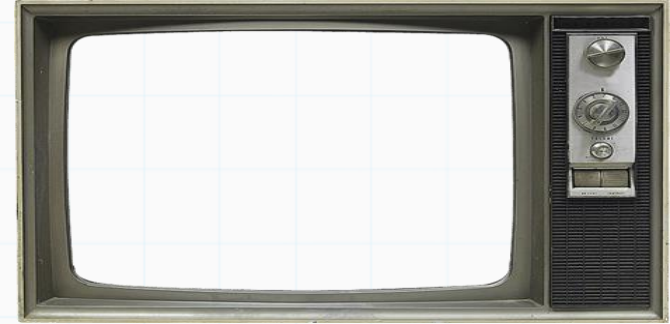
Laços Aninhados

Na literatura da língua inglesa, o termo utilizado é “*nested loops*” (laços aninhados), nada mais é do que laços encaixados, um dentro do outro.

Como funciona ?

```
for h in range (1,24+1,1):  
    for m in range (1,60+1,1):  
        for s in range (1,60+1,1):  
            print (h,m,s)
```

1	1	1	1	2	1
1	1	2	1	2	2
1	1	3	1	2	3
...
1	1	58	1	2	60
1	1	59	1	3	1
1	1	60			



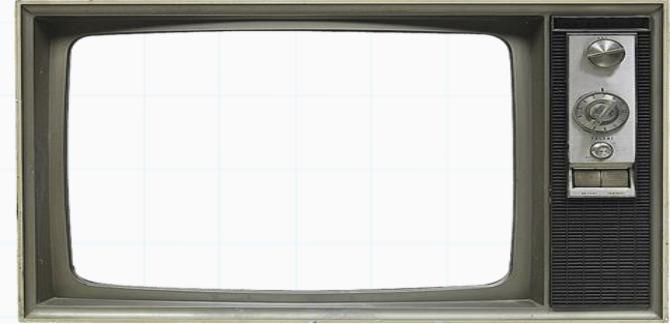
Laços Aninhados

Na literatura da língua inglesa, o termo utilizado é “*nested loops*” (laços aninhados), nada mais é do que laços encaixados, um dentro do outro.

Como funciona ?

```
for h in range (1,24+1,1):  
    for m in range (1,60+1,1):  
        for s in range (1,60+1,1):  
            print (h,m,s)
```

1	1	1	1	2	1
1	1	2	1	2	2
1	1	3	1	2	3
...
1	1	58	1	2	60
1	1	59	1	3	1
1	1	60	1	3	2
...
...	1	3	60
...
...	1	60	60



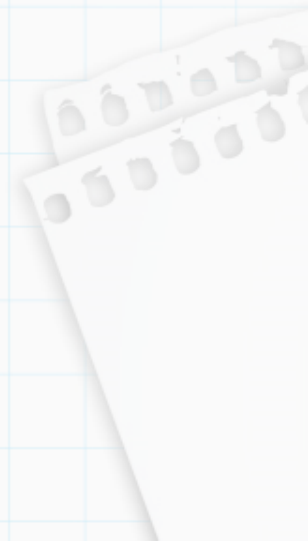
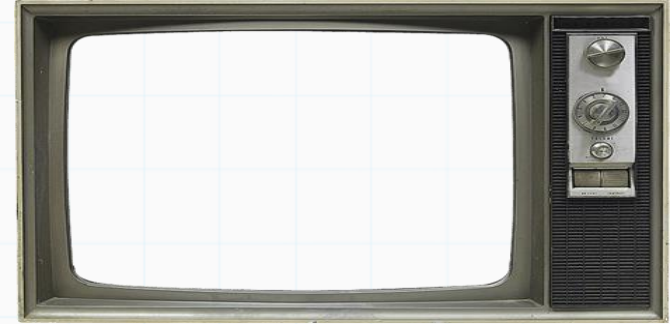
Laços Aninhados

Na literatura da língua inglesa, o termo utilizado é “*nested loops*” (laços aninhados), nada mais é do que laços encaixados, um dentro do outro.

Como funciona ?

```
for h in range (1,24+1,1):  
    for m in range (1,60+1,1):  
        for s in range (1,60+1,1):  
            print (h,m,s)
```

```
1 1 1      1 2 1      2 1 1  
1 1 2      1 2 2  
1 1 3      1 2 3  
...  
1 1 58     1 2 60  
1 1 59     1 3 1  
1 1 60     1 3 2  
...  
1 3 60  
...  
1 60 60
```



Laços Aninhados

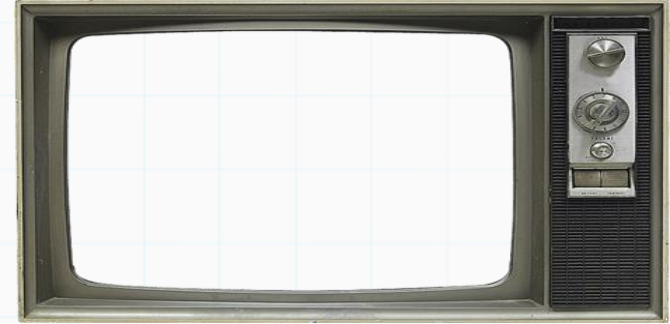
Na literatura da língua inglesa, o termo utilizado é “*nested loops*” (laços aninhados), nada mais é do que laços encaixados, um dentro do outro.

Como funciona ?

```
for h in range (1,24+1,1):  
    for m in range (1,60+1,1):  
        for s in range (1,60+1,1):  
            print (h,m,s)
```



```
1 1 1      1 2 1      2 1 1  
1 1 2      1 2 2      2 1 2  
1 1 3      1 2 3      2 1 3  
...  
1 1 58     1 2 60     2 1 60  
1 1 59     1 3 1  
1 1 60     1 3 2  
...  
1 3 60  
...  
1 60 60
```



Laços Aninhados

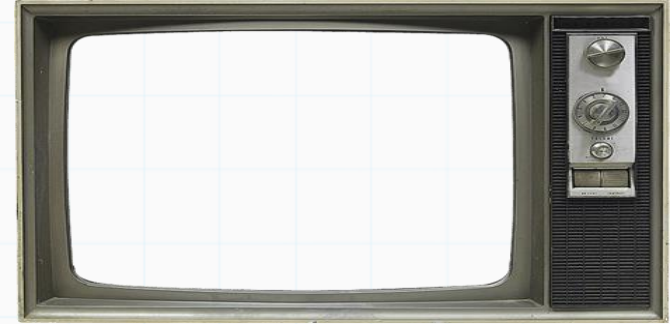
Na literatura da língua inglesa, o termo utilizado é “*nested loops*” (laços aninhados), nada mais é do que laços encaixados, um dentro do outro.

Como funciona ?

```
for h in range (1,24+1,1):  
    for m in range (1,60+1,1):  
        for s in range (1,60+1,1):  
            print (h,m,s)
```



```
1 1 1      1 2 1      2 1 1  
1 1 2      1 2 2      2 1 2  
1 1 3      1 2 3      2 1 3  
...  
1 1 58     1 2 60     2 1 60  
1 1 59     1 3 1      2 2 1  
1 1 60     1 3 2      ...  
...  
1 3 60  
...  
1 60 60
```



Laços Aninhados

Na literatura da língua inglesa, o termo utilizado é “*nested loops*” (laços aninhados), nada mais é do que laços encaixados, um dentro do outro.

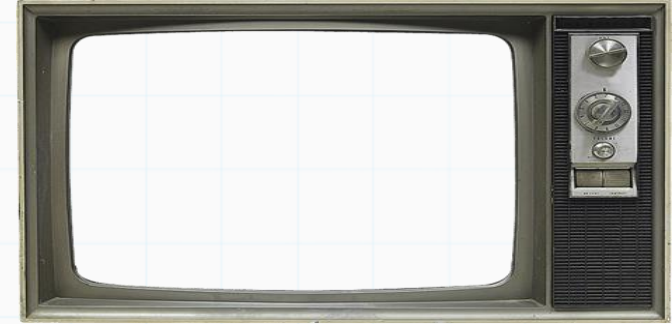
Como funciona ?

```
for h in range (1,24+1,1):  
    for m in range (1,60+1,1):  
        for s in range (1,60+1,1):  
            print (h,m,s)
```



1 1 1	1 2 1	2 1 1
1 1 2	1 2 2	2 1 2
1 1 3	1 2 3	2 1 3
...
1 1 58	1 2 60	2 1 60
1 1 59	1 3 1	2 2 1
1 1 60	1 3 2	...

	1 3 60	24 60 58
	...	24 60 59
	1 60 60	24 60 60

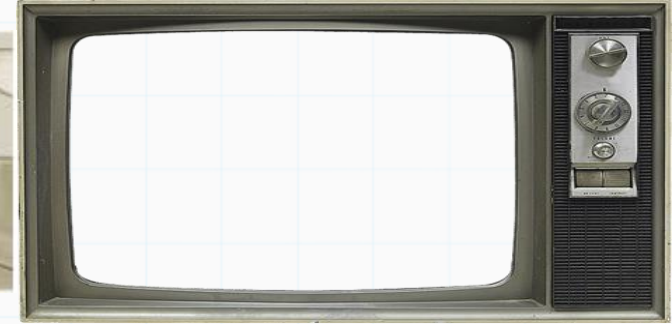


Laços Aninhados

Fura Olho: O que será escrito ?

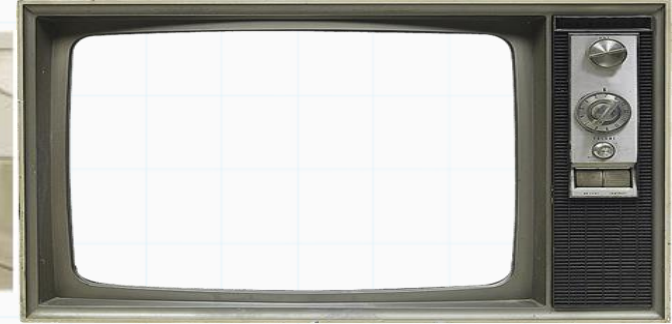
```
1 n = 0
2 for k in range(-5,6,1):
3     n = n+2
4     for m in range(3,1,1):
5         n = n+3
6     n = n+1
7
8 print(n)
```

```
1 s = 0
2 for k in range(1,6,2):
3     for j in range(7,-3,-3):
4         for m in range(1980, 1990,2):
5             s=s+1
6
7 print(s)
```



```
1 passado = 1
2 futuro = 100
3 v = True
4 f = False
5 k = passado
6 while (k < futuro):
7     v = (v and not v) or (f or not f)
8     f = not v
9     futuro = futuro*passado*(-1)
10    k = k + 5
11 if (v):
12     print("ser")
13 if (f):
14     print("não ser")
```

Laços Aninhados



Fura Olho: O que será escrito ?

```
1 n = 0
2 for k in range(-5,6,1):
3     n = n+2
4     for m in range(3,1,1):
5         n = n+3
6     n = n+1
7
8 print(n)
```

33

```
1 s = 0
2 for k in range(1,6,2):
3     for j in range(7,-3,-3):
4         for m in range(1980, 1990,2):
5             s=s+1
6
7 print(s)
```

60

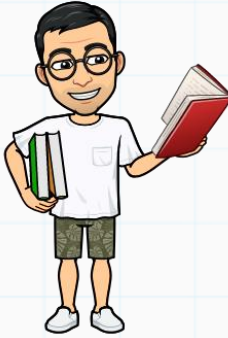
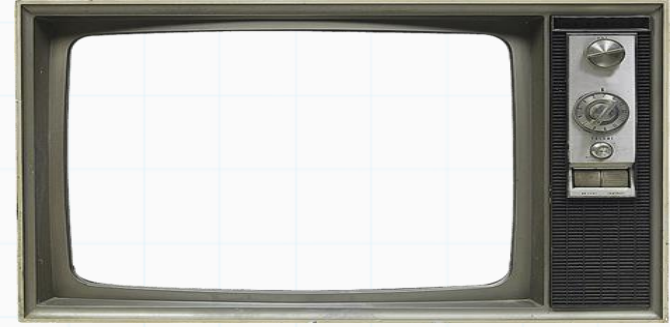
```
1 passado = 1
2 futuro = 100
3 v = True
4 f = False
5 k = passado
6 while (k < futuro):
7     v = (v and not v) or (f or not f)
8     f = not v
9     futuro = futuro*passado*(-1)
10    k = k + 5
11 if (v):
12     print("ser")
13 if (f):
14     print("não ser")
```

ser

Laços Aninhados

Exercício 1) Faça um programa que receba valores “a” e “b” (assuma que $a \leq b$) e imprima todos os números primos entre “a” e “b”

Ex. de Execução: a=5
 b=25
 5
 7
 11
 13
 17
 19
 23

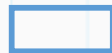


Laços Aninhados

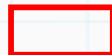
Exercício 1) Faça um programa que receba valores “a” e “b” (assuma que $a \leq b$) e imprima todos os números primos entre “a” e “b”

```
1 a = int(input("a="))
2 b = int(input("b="))
3
4 for i in range (a,b+1,1):
5     cont=0
6     for j in range(2,i,1):
7         if (i%j==0):
8             cont=cont+1
9     if (cont==0) and (i!=1): # pois 1 não é primo
10        print(i)
```

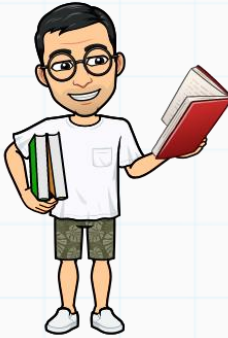
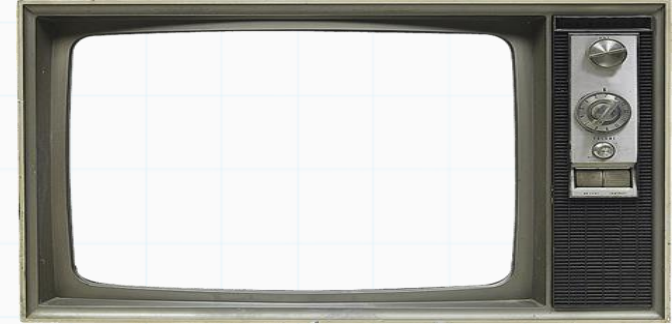
[código](#)



i é primo ?



i varia em [a,b]

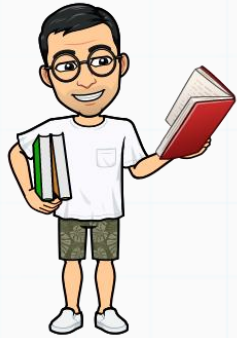


Laços Aninhados

Exercício 2) Calcule a soma da série S para os primeiros 20 termos (imprimindo seus termos) dado um valor inteiro X fornecido. No fim, pergunte se o usuário quer repetir a operação.

$$S = X + \frac{4X^2}{2} + \frac{9X^3}{5} + \frac{14X^4}{11} + \frac{19X^5}{23} + \frac{24X^6}{47} + \dots$$

Ex. de Execução:

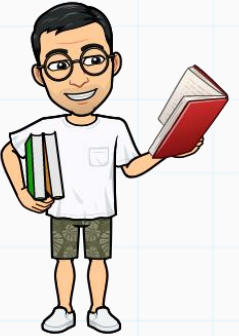
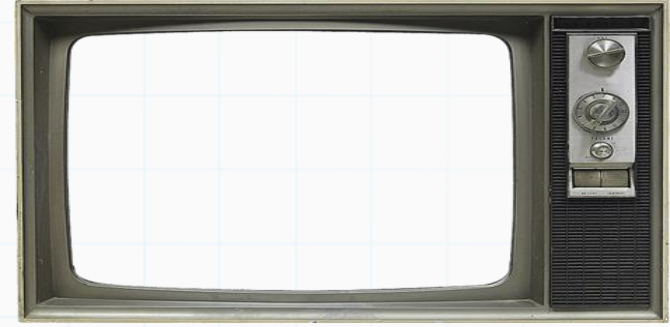


```
x=50
( 4 * 50 ** 2 ) / 2
( 9 * 50 ** 3 ) / 5
( 14 * 50 ** 4 ) / 11
( 19 * 50 ** 5 ) / 23
( 24 * 50 ** 6 ) / 47
( 29 * 50 ** 7 ) / 95
( 34 * 50 ** 8 ) / 191
( 39 * 50 ** 9 ) / 383
( 44 * 50 ** 10 ) / 767
( 49 * 50 ** 11 ) / 1535
( 54 * 50 ** 12 ) / 3071
( 59 * 50 ** 13 ) / 6143
( 64 * 50 ** 14 ) / 12287
( 69 * 50 ** 15 ) / 24575
( 74 * 50 ** 16 ) / 49151
( 79 * 50 ** 17 ) / 98303
( 84 * 50 ** 18 ) / 196607
( 89 * 50 ** 19 ) / 393215
( 94 * 50 ** 20 ) / 786431
s= 1.1847657890985137e+30
De novo? (1)-Sim (0)-Não1
x=12
( 4 * 12 ** 2 ) / 2
( 9 * 12 ** 3 ) / 5
( 14 * 12 ** 4 ) / 11
( 19 * 12 ** 5 ) / 23
( 24 * 12 ** 6 ) / 47
( 29 * 12 ** 7 ) / 95
( 34 * 12 ** 8 ) / 191
( 39 * 12 ** 9 ) / 383
( 44 * 12 ** 10 ) / 767
( 49 * 12 ** 11 ) / 1535
( 54 * 12 ** 12 ) / 3071
( 59 * 12 ** 13 ) / 6143
( 64 * 12 ** 14 ) / 12287
( 69 * 12 ** 15 ) / 24575
( 74 * 12 ** 16 ) / 49151
( 79 * 12 ** 17 ) / 98303
( 84 * 12 ** 18 ) / 196607
( 89 * 12 ** 19 ) / 393215
( 94 * 12 ** 20 ) / 786431
s= 5.440372266587512e+17
De novo? (1)-Sim (0)-Não0
```

Laços Aninhados

Exercício 2) Calcule a soma da série S para os primeiros 20 termos (imprimindo seus termos) dado um valor inteiro X fornecido. No fim, pergunte se o usuário quer repetir a operação.

$$S = X + \frac{4X^2}{2} + \frac{9X^3}{5} + \frac{14X^4}{11} + \frac{19X^5}{23} + \frac{24X^6}{47} + \dots$$



```
1 resp=1
2 while (resp==1):
3     x = int(input("x="))
4     s = x
5     a = 4
6     b = 2
7     c = 2
8
9     for i in range(1,20,1):
10        s = s + (a*(x**b)/c)
11        print("(" ,a, "*",x, "**",b, ")/",c)
12        a = a + 5
13        b = b + 1
14        c = 2*c + 1
15    print("s=",s)
16    resp = int(input("De novo? (1)-Sim (0)-Não"))
```



calcula série para um X



faz de novo ?

[código](#)

Laços Aninhados

Exercício 3) Calcule a soma da série S para os primeiros “n” termos (inteiro), imprimindo seus termos

$$S = \frac{2}{1!} + \frac{4}{4!} + \frac{8}{7!} + \frac{16}{10!} + \frac{32}{13!} + \dots$$

Ex. de Execução:

n=7

2 / 1

4 / 24

8 / 5040

16 / 3628800

32 / 6227020800

64 / 20922789888000

128 / 121645100408832000

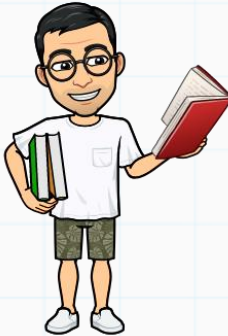
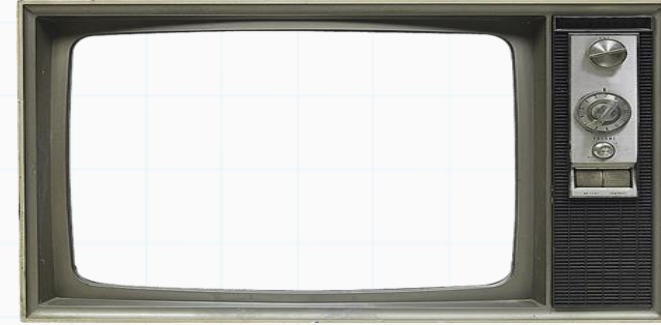
s= 2.168258382566998



laço dos termos

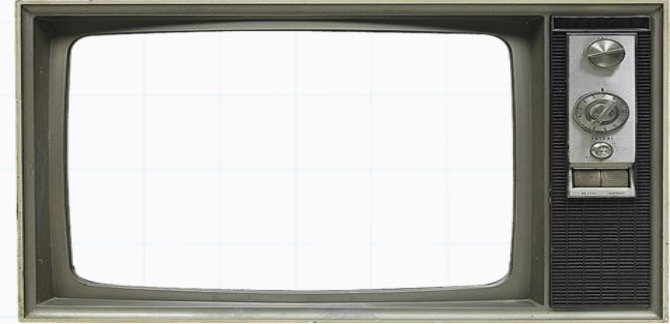


laço do fatorial do termo



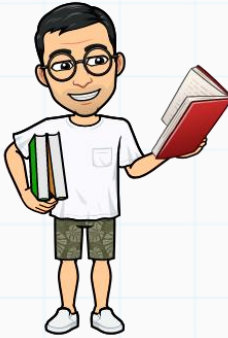
Dica: Veja que agora cada termo possui um fatorial que pode ser calculado com um laço (interno)

Laços Aninhados



Exercício 3) Calcule a soma da série S para os primeiros “n” termos (inteiro), imprimindo seus termos

$$S = \frac{2}{1!} + \frac{4}{4!} + \frac{8}{7!} + \frac{16}{10!} + \frac{32}{13!} + \dots$$



```
1 n = int(input("n="))
```

```
2 s = 0
```

```
3 a = 2
```

```
4 b = 1
```

inicialização do termo genérico (a/b)

```
5  
6 for i in range(0,n,1):
```

```
7     f = 1
```

```
8     for j in range(2,b+1,1):
```

```
9         f = f*j
```

fatorial de b

```
10     print(a,"/",f)
```

```
11  
12     s = s + (a/f)
```

```
13     a = a * 2
```

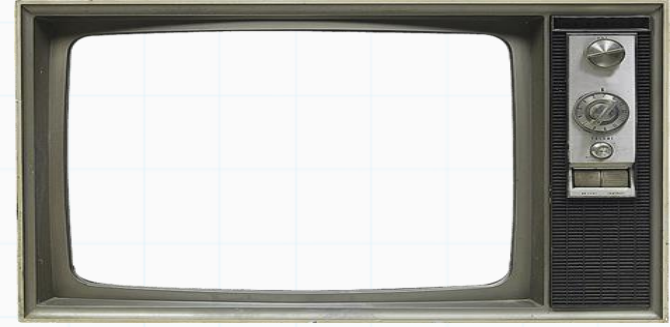
```
14     b = b + 3
```

atualização do termo genérico

```
15  
16 print("s=",s)
```

[código](#)

Até a próxima



Slides baseados no curso de Vanessa Braganholo

