

Introdução ao Processamento de Dados

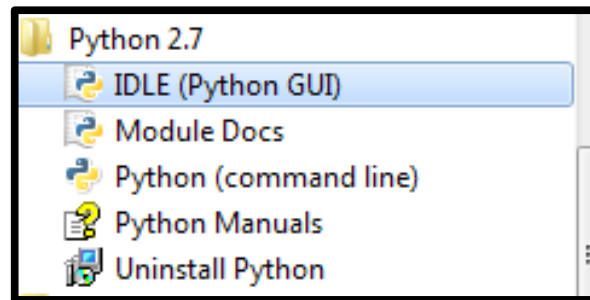
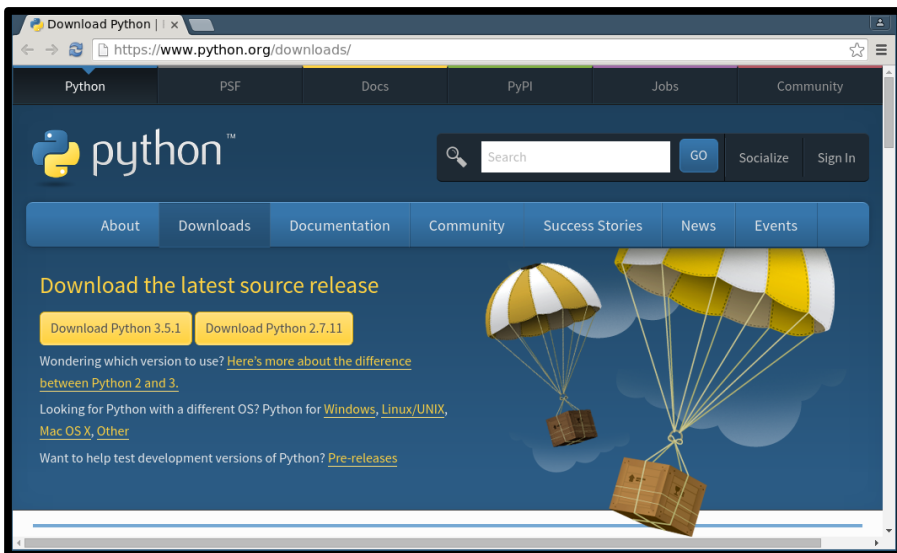
Francisco Sant'Anna

`francisco@ime.uerj.br`

`http://github.com/fsantanna/IPD`

Python

- Download 2.7.*
 - www.python.org/downloads
- Instalar o arquivo baixado
 - <https://www.python.org/ftp/python/2.7.11/python-2.7.11.msi>
- Executar o “IDLE”



anim.gif

Conceitos de Programação

- Variável
- Atribuição
- Controle de Fluxo de Execução
 - Sequência
 - Condicional
 - Repetição
- Constante (e.g., 1, "Ola Mundo")
- Expressão (e.g., 1+10, n==5, input())
- Função (e.g., print("Ola Mundo"), input())

Variáveis

- Uma abstração da memória do computador
- Uma “etiqueta” que representa uma região de memória
- O programa pode **ler** e **escrever** valores na memória através dessas etiquetas.

Escrita
(atribuição)

```
escreva( "Numero: " )  
n ← leia( )  
soma ← 0  
enquanto n > 0:  
    soma ← (soma + n)  
    n ← (n - 1)  
escreva( "somou", soma )
```

Leitura

Controle de Fluxo

- Sequência
- Condicional
- Repetição

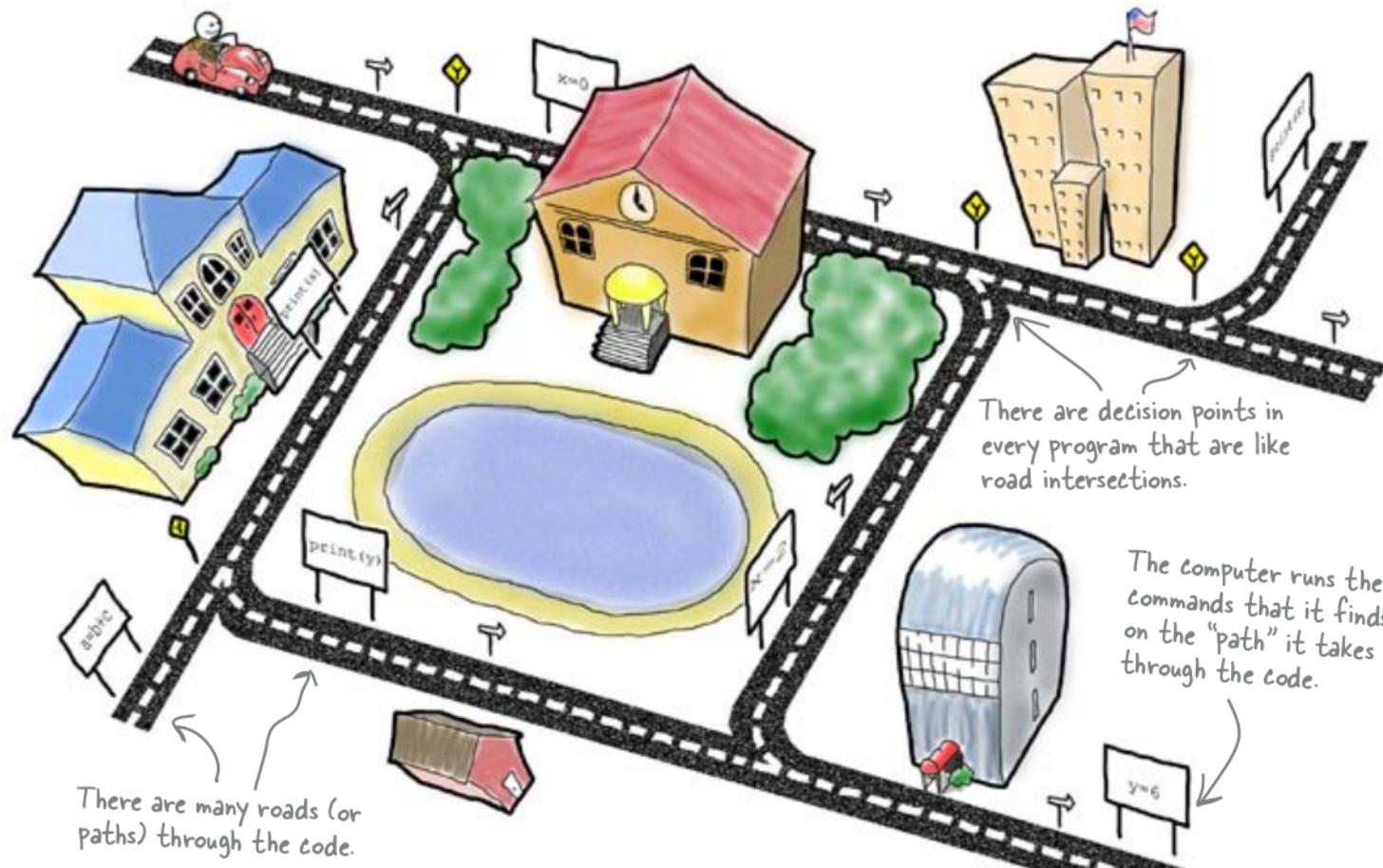
```
escreva( "Escolha entre 1-9:")  
n ← leia( )  
se n == 5:  
    escreva( "Voce acertou!")  
senao:  
    escreva( "Voce errou!")
```

```
escreva( "Escolha um numero:")  
n ← leia( )  
soma ← 0  
enquanto n > 0:  
    soma ← (soma + n)  
    n ← (n - 1)  
escreva( "somou", soma)
```

Controle de Fluxo

Codeville: Your program is like a network of roads

Programs need to do different things under different circumstances. In the game, the code displays “You win!” if the user guesses the number correctly, and “You lose!” if not. This means that all programs, even really simple programs, typically have multiple **paths** through them.

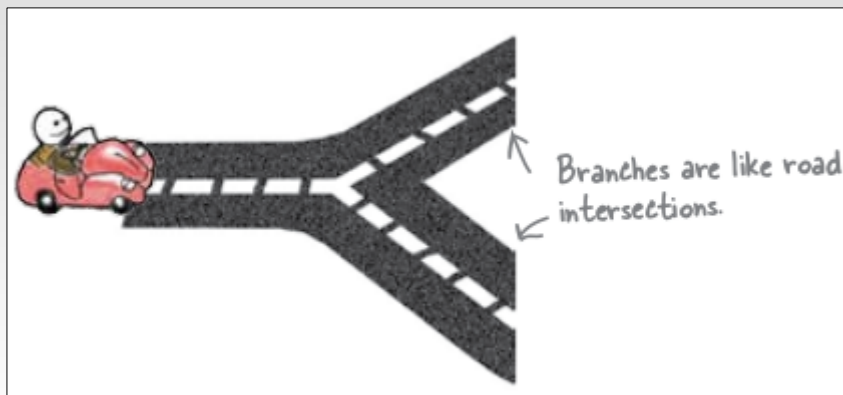


Sequência (linha, `;`)

```
escreva("Escolha entre 1-9:")  
n ← leia()  
se n == 5:  
    escreva("Voce acertou!")  
senao:  
    escreva("Voce errou!")
```

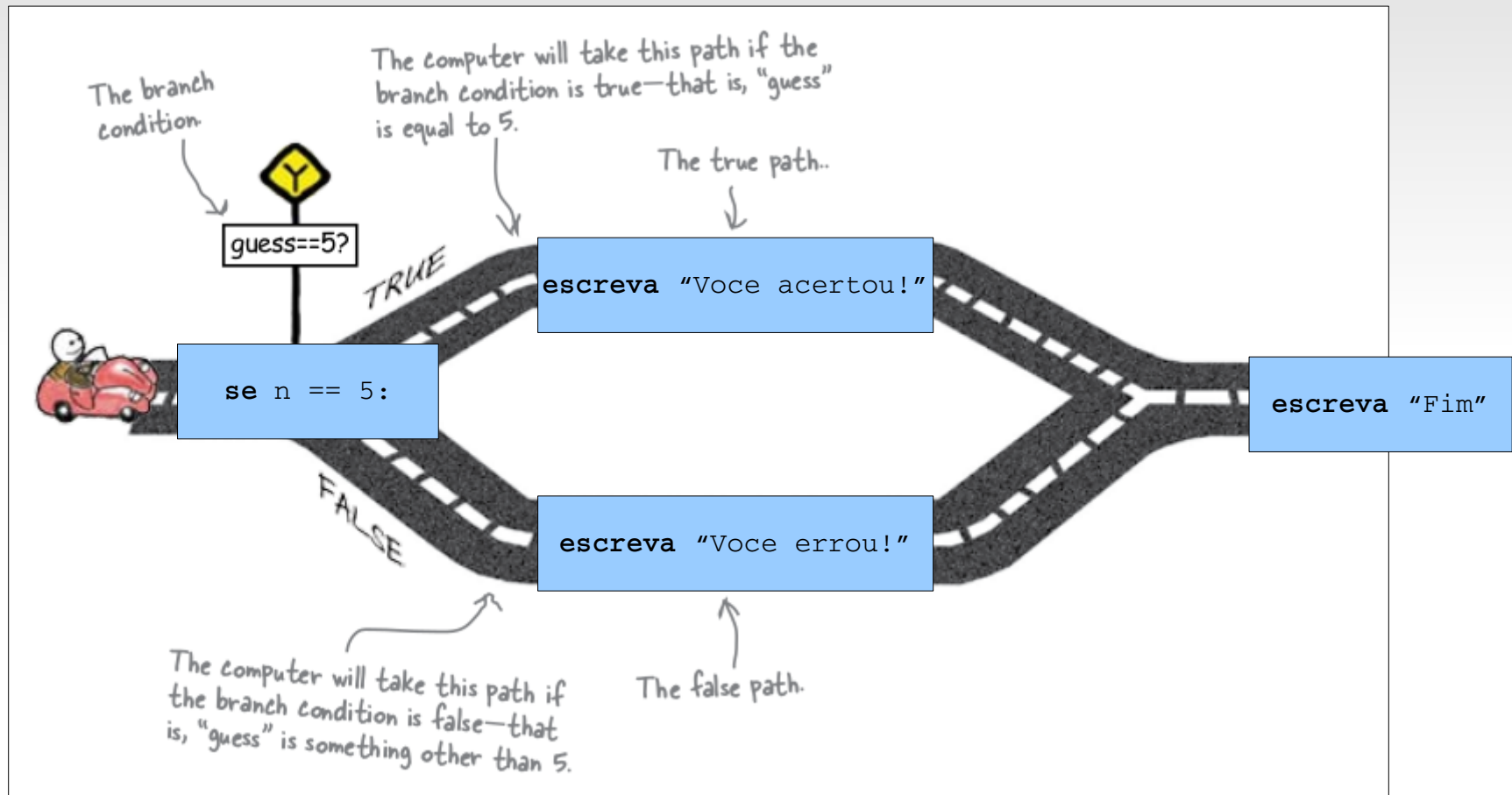
```
escreva("Escolha entre 1-9:") ; n ← leia()
```


Condicional (if)



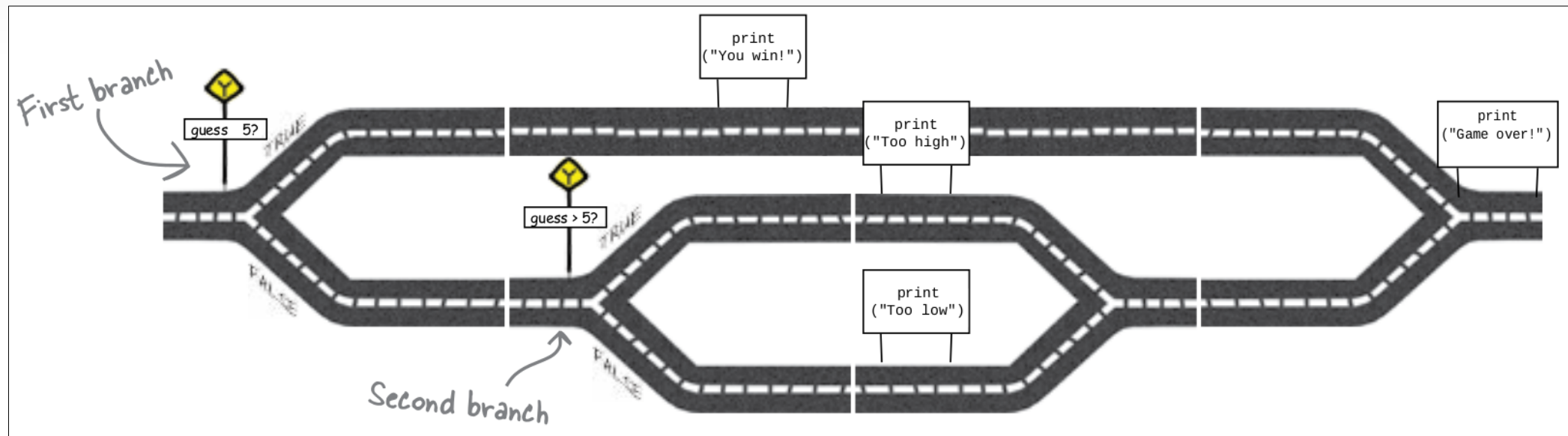
```
escreva("Escolha entre 1-9:")  
n ← leia()  
se n == 5:  
    escreva("Voce acertou!")  
senao:  
    escreva("Voce errou!")  
escreva("Fim")
```

Condicional (if)



Exercício: Condicional (if)

- Construir um programa que exiba a mensagem correta



Exercício: Condicional (if)

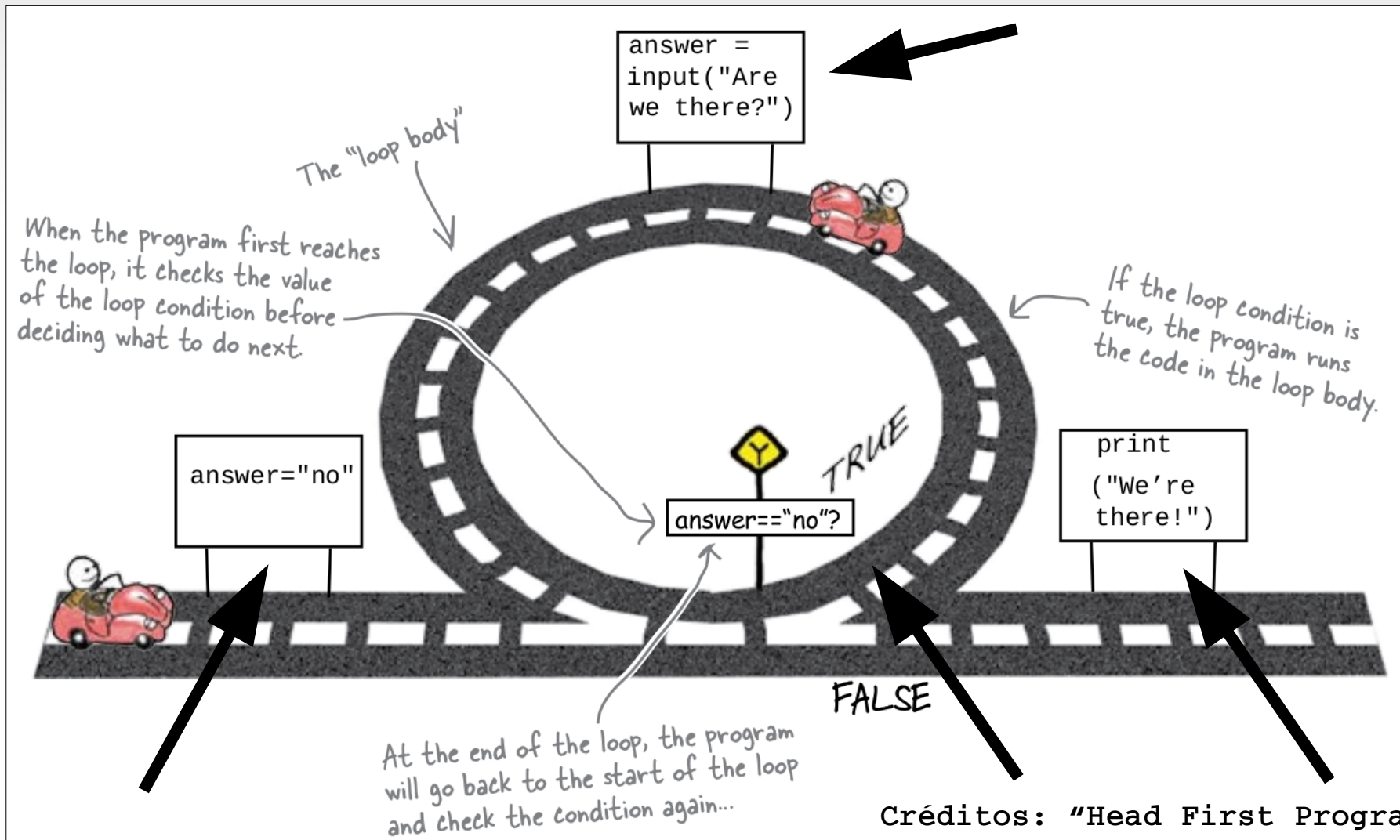
- Construir um programa que exiba a mensagem correta

```
escreva("Escolha entre 1-9:")  
n ← leia()  
se n == 5:  
    escreva("Voce acertou!")  
senao:  
    se n > 5:  
        escreva("Muito alto...")  
    else:  
        escreva("Muito baixo...")  
escreva("Fim")
```


Repetição (while)

```
resposta = "nao"
while resposta == "nao":
    print "Estamos chegando?"
    resposta = raw_input()
print "Chegamos!"
```

```
resposta = "nao"
while resposta == "nao":
    print "Estamos chegando?"
    resposta = raw_input()
print "Chegamos!"
```



Exercício: Condicional (if)

- **Leia dois números**
- **Exiba (imprima) o maior valor lido**

Exercício: Repetição (while)

- **Leia números continuamente**
- **Termine quando ler um número menor que o anterior**
- **Exiba (imprima) o maior valor lido**
- **(Assuma que só serão digitados números maiores que 0)**

Exercício: Repetição (while)

- **Leia números continuamente**
- **Termine quando ler um número menor que o anterior**
- **Exiba (imprima) a soma de todos os valores lidos**
- **(Assuma que só serão digitados números maiores que 0)**

Exercício: Repetição (while)

- **Leia 10 números**
- **Exiba o maior valor lido**
- **(Assuma que só serão digitados números maiores que 0)**

Repetição (for -> while)

```
frase = raw_input()  
print("----")  
for i in range(1,5):  
    print(i, frase)  
print("----")
```

```
frase = raw_input()  
print("----")  
i = 1  
while i < 5:  
    print(i, frase)  
    i = i + 1  
print("----")
```

```
frase = raw_input()  
print("----")  
i = 1  
while True:  
    if i < 5:  
        break;  
    else:  
        print(i, frase)  
        i = i + 1  
print("----")
```