print ("Responda o Questionário com True ou False")

pergunta1 = (input("Você mora perto da Vítima? "))

pergunta2 = (input("Você ja trabalhou com a Vítima? "))

pergunta3 = (input("Você telefonou para a Vítima? "))

pergunta4 = (input("Você esteve no local do Crime? "))

pergunta5 = (input("Você devia dinheiro à Vítima? "))

if pergunta1 == 'True' and pergunta2 == 'True' and pergunta3 == 'False' and pergunta4 == 'False' and pergunta5 =='True':

print ("Você é Cúmplice")

else:

print ("Dados não conclusivos")

print ("Responda as perguntas somente com sim ou não")

pergunta1 = input("Você sente dor no corpo? ")

pergunta2 = input("Você tem Febre? ")

pergunta3 = input("Você tem Tosse? ")

pergunta4 = input("Você esta com congestão nasal? ")

pergunta5 = input("Você tem manchas no corpo? ")

if pergunta1 == 'sim' and pergunta2 == 'sim' and pergunta3 == 'não' and pergunta4 == 'não' and pergunta5 =='sim':

resultado = 'Você está com dengue'

elif pergunta1 == 'sim' or pergunta1 == 'não' and pergunta2 == 'sim' and pergunta3 == 'sim' and pergunta4 == 'sim' and pergunta5 =='não':

resultado = 'Você está com Gripe'

elif pergunta1 == 'sim' or pergunta1 == 'não' and pergunta2 == 'não' and pergunta3 == 'não' and pergunta4 == 'não' and pergunta5 == 'não':

resultado = "Você não está doente"

else:

resultado = 'Sem sintomas'

print (resultado)

Aula 6 - Laços de repetição.ipynb\_

Arquivo

Editar

Ver

Inserir

Ambiente de execução

Ferramentas

Ajuda

As alterações não serão salvas

Compartilhar

[](https://accounts.google.com/SignOutOptions?hl=pt-BR&continue=https://colab.research.google.com/drive/1BEdG8PbYK0Fsqy7klCDBvNoBQswLi8ex%3Fusp%3Dsharing)

CódigoTextoCopiar para o DriveConectarEditar

Laços de repetição

Assim como a tomada de decisão é um conceito muito importante em nosso cotidiano, também é a ideia de repetição. Por diversas vezes precisamos repetir algumas ações a fim de realizar alguma tarefa. Alguns exemplos do dia-a-dia: precisamos estudar em repetição até dominar algum assunto, entre outras situações....

Assim, vamos trazer esse conceito para os algoritmos. Fazer **laços de repetição**, que são uma extensão natural da ideia de **controle de fluxo**.

Relembrando controle de fluxo

Todos os conceitos que tratavam da lógica, das operações lógicas, das proposições e afins, ainda estarão presentes nesse novo assunto.

Considere o seguinte algoritmo:

Exemplo 1  
1. idade <- ler idade  
2.1 se idade é maior que 18, apresente 'Vc é maior de idade'  
2.2 caso contrário, apresente 'Vc é menor de idade'

O algoritmo acima é de uma execução muito bem conhecida. O usuário informa sua idade e nosso algoritmo responde se ele é maior ou menor de idade.

O algoritmo do Exemplo 1 possui o seguinte fluxograma:

[IMAGEM AQUI]

Exemplo prático de laço

Suponha a seguinte situação: fazer um algoritmo que peça a idade do usuário e que *enquanto* o usuário for menor de idade, o algoritmo continua pedindo uma novo idade para o usuário. A ideia dessa situação seria pedir uma idade *enquanto* essa for menor que 18.

Para implementarmos essa nova versão vamos criar um novo comando em nosso algoritmo: o comando enquanto. Assim, podemos implementar essa ideia de repetição.

Um algoritmo para satisfazer a situação exposta é o seguinte:

Exemplo 2  
1. idade <- ler idade  
2.1 enquanto idade menor que 18, volte para linha 1.  
2.2 caso contrário, apresente "Vc é maior de idade"

O fluxograma do algoritmo do Exemplo 2 é o seguinte:

[IMAGEM AQUI]

A ideia por trás de laços de repetição é **repetir uma ação enquanto uma condição lógica for verdadeira**.

Tipos de laços

A ideia em aplicar laço é que teremos uma ação que será repetida dependendo de uma condição lógica. É muito parecido com o se/caso contrário, a diferença é que controles de fluxo são sequenciais e não repetitivos. O novo comando enquanto é repetitivo.

O comando enquanto pode resultar em dois tipos de laço: os **infinitos** e os **finitos**.

Laços infinitos

Um laço corretamente implementado é um laço que **sempre** termina, em algum momento pode terminar programaticamente. Entretanto, o tipo infinito também pode acontecer.

Laços infinitos são laços que **jamais** terminam sozinhos e devem ser evitados. Alguns exemplos são os seguintes algoritmos:

Exemplo 3  
1. numero <- ler número  
2. enquanto Verdade, apresente o valor da variável numero

O Exemplo 3 é um laço infinito por que a condição do comando enquanto é fixada em Verdade, ou seja, é uma condição estática.

Exemplo 4  
1. numero <- ler número  
2. x = 0  
2. enquanto x é igual a 0, apresente o valor da variável número

O Exemplo 4 é um laço infinito por que apesar da condição não ser estática, a variável x nunca é alterada no decorrer do algoritmo.

Laço finito

Um laço finito é um caso normal de repetição, ou seja, ações específicas são executadas enquanto uma condição for verdadeira, mas em algum momento a execução termina.

Um laço só termina a execução de suas ações quando sua condição deixar de ser verdadeira. O Exemplo 5 é um exemplo de laço normal (finito).

Exemplo 5  
   
1. numero <- ler número  
2. enquanto numero é diferente de 0, leia o número novamente

Exemplos complementares

1. idade <- ler idade  
2.1 enquanto idade menor que 18,  
2.1.1 idade <- ler idade  
2.2 caso contrário,  
2.2.1 apresente "Vc é maior de idade"

Exercícios

1) Defina de forma sucinta a diferença entre comandos de repetição e controle de fluxo.

*Resposta: o controle de fluxo serve para que nosso algoritmo tome decisões e o laço de repetição é uma extensão da ideia de controle de fluxo, que executa uma ação enquanto uma condição for verdadeira*

2) Analise o algoritmo abaixo e diga qual será o valor de x ao final.

1. x <- 0  
2. enquanto x é menor que 5, x <- x + 1  
3. apresente o valor de x

*Resposta: x é 5*

3) Analise o código abaixo e diga quais serão os valores de x e y ao final.

1. x <- 1  
2. y <- 0  
3. enquanto x é menor que 5, y <- y + 2  
4. apresente o valor de x e y

*Resposta: esse algoritmo tem um laço infinito, portanto não tem final. Mas x = 1 e y sempre será um par (que cresce infinitamente)*

4) O que faz um laço ser infinito?

*Resposta: Um laço infinito é aquele cuja condição nunca deixa de ser Verdade. Ou seja, a condição de Falso nunca acontece*

5) Identifique qual dos algoritmos é um laço infinito:

Algoritmo 1  
1. x <- 1  
2. enquanto x é diferente de 0, apresente o valor de x

Algoritmo 2  
1. x <- 1  
2. enquanto x é diferente de 0, x <- x + 1

Algoritmo 3  
1. numero <- ler número  
2. enquanto numero é menor que 10, apresente o valor de numero

6) Faça um algoritmo que peça ao usuário um número e imprima todos os números de um até o número dado.

Exemplo:  
input: 5  
output: 1 2 3 4 5

*Resposta:*

1. numero <- ler número  
2. contador <- 1  
3.1 enquanto contador é menor igual a numero,  
3.1.1 apresente o valor de contador  
3.1.2 contador <- contador + 1

7) Faça um algoritmo que peça para um usuário digitar um número e que só finaliza quando o usuário digitar 0.

*Resposta:*

1. num <- ler número  
2.1 enquanto num é diferente de 0,  
2.1.1 num <- ler número

8) Faça um programa que peça para o usuário digitar a idade, o salário e o sexo de uma pessoa até que as entradas digitadas sejam válidas.

1. Idade: entre 0 e 150;
2. Salário: maior que 0;
3. Gênero: M, F ou Outro.

*Resposta:*

1. idade <- ler idade  
2.1 enquanto idade menor que 0 ou maior que 150,   
2.1.1 idade <- ler idade  
   
3. salario <- ler salário  
4.1 enquanto salario menor igual que 0,   
4.1.1 salario <- ler salário  
   
5. genero <- ler genero  
6.1 enquanto genero diferente de M ou F ou Outro,   
6.1.1 genero <- ler gênero

9) Peça ao usuário para digitar um número n e some todos os números de 1 a n.

Exemplo 1:  
input: 4  
output: 10

Exemplo 2:  
input: 3  
output: 6

*Resposta:*

1. numero <- ler número  
2. contador <- 1  
3. somatorio <- 0  
4.1 enquanto contador é menor igual a numero,  
4.1.1 apresente o valor de contador  
4.1.2 somatorio <- somatorio + contador  
4.1.3 contador <- contador + 1  
5. apresentar o valor da variável somatorio

Carregando…