

QUESTÃO EM PYTHON					
PADRÃO DE EQUÍVOCO					
ID		NOME DO PADRÃO DE EQUÍVOCO			
P_RS1		Ordem incorreta dos comandos na estrutura			
QUESTÃO					
ID da QUESTÃO		TÍTULO DA QUESTÃO			
29_FHI01		Seleção em Vetor			
<p>Problema: Um aluno aprendendo sobre vetores, resolveu criar um código em que podia digitar uma quantidade infinita de números que seria guardado em um vetor, e ao digitar o número 0 o programa deveria parar de receber números e imprimir todos os números digitados. O código desenvolvido foi o seguinte:</p> <div><pre>vetor = [] x = int(input("Digite um valor: ")) while x != 0: x = int(input("Digite um valor: ")) vetor.append(x) print(vetor)</pre></div>					
<p>Ao executar o código ele percebeu que estava saindo alguns valores indesejados, inclusive o 0, analisando o código, o que pode estar causando esse erro?</p> <div><div>a. O comando append adiciona o valor digitado no final do vetor, e não no começo, fazendo com que quando o 0 é digitado ele seja adicionado no final do vetor, logo o vetor não para de receber valores</div><div>b. Realizando um teste de mesa percebe-se que os comandos dentro do while está em ordem incorreta e isso faz com que o 0 seja adicionado no vetor e o primeiro valor não seja adicionado</div><div>c. Substituir o while por um for faz com que o código funcione corretamente pois o for é utilizado para percorrer uma lista e o while é utilizado para repetir um bloco de código enquanto uma condição for verdadeira</div><div>d. O ":" é utilizado somente em estruturas de seleção e não em estruturas de repetição como o while e o for e isso faz com que o python não entenda que o bloco de código abaixo dele pertence a ele</div></div>					
Alternativa correta: (B)					
TIPO DE ERRO:		Sintaxe	x	Semântica	Estilo
<p>Conteúdos necessários para entender o código: F: Expressões Relacionais (>, >=, <, <=, ==, !=) H: Estruturas de Repetição (While, for, do..while, ...) I: Vetores</p>					
<p>Feedback geral: A importância de testes de mesa nos código é que eles nos ajudam a identificar erros de lógica, e também nos ajudam a identificar erros de sintaxe, pois ao analisar o código com calma é possível identificar que os comandos estão em ordem incorreta.</p>					
<p>Feedback sobre as respostas:</p> <div><div>a. Falso, o comando append adiciona o valor digitado no final do vetor, mas isso não faz com que o 0 seja adicionado no vetor, o motivo disso estar ocorrendo é que o comando append está dentro do while, fazendo com que ele seja executado a cada iteração do while, e como o 0 é digitado dentro do while antes de ser validado, ele é adicionado no vetor</div><div>b. Verdadeiro,, por conta dos comandos estarem em ordem incorreta o valor é informado, adicionado no vetor e depois validado, fazendo com que o 0 seja adicionado no vetor</div><div>c. Falso, o for é utilizado para percorrer uma lista e o while é utilizado para repetir um bloco de código enquanto uma condição for verdadeira, mas isso não faz com que o código funcione corretamente, pois o problema está na ordem dos comandos</div><div>d. Falso, o ":" é utilizado em estruturas de seleção e de repetição, e isso não faz com que o python não entenda que o bloco de código abaixo dele pertence a ele.</div></div>					