

 Universidade Luterana do Brasil ULBRA Torres Pró-Reitoria de Graduação		Tipo de atividade: Prova () Trabalho () () Avaliação: AP1 () AP2 () AS () AF ()
Curso:	Disciplina:	Data:
Turma:	Professor(a):	Valor da Avaliação:
Acadêmico(a):	n°:	Nota:

Orientações para a Prova

Caro aluno, leia atentamente as orientações a seguir para a realização da sua prova:

A) Quantidade de Provas: Existem 4 versões diferentes da prova. As questões são semelhantes entre elas, porém, pequenas especificidades podem alterar as respostas. Portanto, leia cada questão com atenção e confie em seu conhecimento.

B) Estrutura da Prova:

- Bloco A:
 - Composto por 10 questões de múltipla escolha.
 - Peso: 0.5
- Bloco B:
 - Contém 4 algoritmos. Escolha e resolva 2 deles.
 - Peso: 0.5
- Bloco C:
 - Apresenta 3 algoritmos. Resolva apenas 1.
 - Peso: 0.5

- ****Bloco D**** (Opcional):

- Se preferir, pode optar por responder apenas 1 questão deste bloco, que tem peso de 1.5, em vez de responder questões dos Blocos A, B e C.

3. Uso de Computador:

- A prova é realizada no papel.
- No entanto, entre 20:22 e 20:45, você tem a opção de usar o computador exclusivamente para testar algoritmos e consultar algoritmos que já foram realizados por você anteriormente.
- O uso do computador é opcional. Certifique-se de gerenciar seu tempo adequadamente.

Lembramos que cada detalhe é importante. Organize-se e boa prova!

Gabarito

Questão	A	B	C	D	E
Questão 1					
Questão 2					
Questão 3					
Questão 4					
Questão 5					
Questão 6					
Questão 7					
Questão 8					
Questão 9					
Questão 10					

Introdução ao Bloco A

Bem-vindo ao Bloco A da prova. Nesta seção, você encontrará questões de múltipla escolha centradas em conceitos fundamentais da linguagem Python. Os tópicos abordados abrangem desde a sintaxe básica até características intrínsecas da linguagem.

Ao responder a cada questão, lembre-se de refletir sobre suas experiências e estudos anteriores em Python, e escolha a opção que você acredita ser a mais correta.

Boa sorte e que você faça uma ótima prova!

<p>1. Qual é a sintaxe correta para coletar uma entrada do usuário em Python?</p> <p>a) <code>user_input()</code> b) <code>input.user()</code> c) <code>get_input()</code> d) <code>input()</code> e) <code>ask_user()</code></p>	<p>2. Como você imprime "Olá, Mundo!" em Python?</p> <p>a) <code>echo("Olá, Mundo!")</code> b) <code>System.out.println("Olá, Mundo!")</code> c) <code>cout << "Olá, Mundo!";</code> d) <code>print "Olá, Mundo!"</code> e) <code>print("Olá, Mundo!")</code></p>
---	---

<p>3. Qual é o tipo de dado de `5.7` em Python?</p> <p>a) int b) char c) string d) long e) float</p>	<p>4. Qual é a saída do código a seguir?</p> <pre>Python x = 10 if x > 5: print("Maior") elif x < 5: print("Menor") else: print("Igual")</pre> <p>a) Maior b) Menor c) Igual d) x > 5 e) Nenhuma das anteriores</p>
<p>5. Dado o seguinte código em Python, qual será a saída?</p> <pre>Python x = 5 if x == 5: output = "Cinco" elif x == 6: output = "Seis" else: output = "Desconhecido" print(output)</pre> <p>a) 5 b) Seis c) Desconhecido d) Cinco e) x = 5</p>	<p>6. O que o código a seguir imprime?</p> <pre>Python if False: print("Isso não será impresso.") else: print("Isso será impresso.")</pre> <p>a) Isso não será impresso. b) Isso será impresso. c) False d) Nada será impresso. e) O código gerará um erro.</p>
<p>7. Qual comando em Python é usado para verificar múltiplas condições em sequência?</p> <p>a) switch b) select c) if-then-else d) elif e) choose</p>	<p>8. Qual é o tipo de dado de `True` em Python?</p> <p>a) string b) integer c) char d) float e) bool</p>
<p>9. Qual é a sintaxe correta para imprimir o valor de uma variável chamada "nome"?</p>	<p>10. Como você verifica se "x" é igual a 5 em Python?</p> <p>a) x == 5</p>

a) echo nome b) print(nome) c) System.out.println(nome) d) println(nome) e) print = nome	b) x equals 5 c) x = 5 d) x is 5 e) x eq 5
--	---

Bloco B

Adentrando agora no Bloco B, você será desafiado a resolver algoritmos com um grau de simplicidade. Esta seção é projetada para avaliar sua habilidade em estruturar soluções lógicas e implementá-las usando Python, sem a necessidade de usar estruturas condicionais.

1. Conversão de Temperatura

Você foi contratado para fazer um software meteorológico. Uma das funcionalidades mais pedidas pelos usuários é a conversão de temperaturas de Celsius para Fahrenheit.

Entrada:

O programa deve solicitar ao usuário uma temperatura em graus Celsius.

Saída:

O programa deve mostrar a temperatura convertida em Fahrenheit.

Fórmula:

$$F = (C * 9/5) + 32$$

2. Preparativos para Jantar

Você está ajudando na organização de um grande jantar de família. Para comprar os ingredientes, você decide calcular a quantidade de espaguete que será necessário.

Entrada:

O programa deve perguntar quantos membros da família vão participar do jantar.

Saída:

O programa deve informar a quantidade de espaguete (em gramas) necessária para servir a todos.

Nota:

Considere que cada pessoa consome 100g de espaguete.

3. Organização de Festa

Você está planejando uma festa de aniversário e quer garantir que não falte bebida. Decide criar um programa que calcule a quantidade de bebida necessária.

Entrada:

O programa deve solicitar o número de convidados.

Saída:

O programa deve informar a quantidade total de bebida (em litros) que você precisa comprar.

Nota:

Estime que cada convidado consuma 500ml de bebida.

4. Sistema de Caixa Simples

Para treinar seus conhecimentos em Python, você decide criar um programa simples de caixa que ajude a calcular o troco dado ao cliente.

Entrada:

O programa deve perguntar o valor total da compra e o valor entregue pelo cliente para pagar.

Saída:

O programa deve informar o valor do troco que o cliente deve receber.

Bloco C

Avançando para o Bloco C, você encontrará desafios que exigem um passo adicional na complexidade: o uso de estruturas condicionais. Este bloco foi cuidadosamente projetado para avaliar sua capacidade de tomar decisões lógicas no código, utilizando estruturas if, elif e else no Python.

1. Categoria de Piscinas

Uma loja vende piscinas retangulares de diferentes tamanhos. Com base na área da piscina, elas são classificadas em três categorias: Pequena, Média e Grande.

Entrada:

O programa deve solicitar ao usuário a altura e a largura de um retângulo representando a piscina.

Saída:

O programa deve informar a área da piscina e sua categoria.

Fórmula:

Área = altura x largura

* Pequena: Até 10 m²

* Média: Mais de 10 m² até 30 m²

* Grande: Acima de 30 m²

2. Ponto de Congelamento

Com as mudanças climáticas, uma cidade está experimentando temperaturas extremas. O departamento de meteorologia quer alertar os cidadãos quando a temperatura se aproxima do ponto de congelamento.

Entrada:

O programa deve solicitar ao usuário a temperatura atual em graus Celsius.

Saída:

O programa deve informar se está "Seguro", "Próximo ao ponto de congelamento" ou "Abaixo do ponto de congelamento".

* Seguro: Acima de 5°C

* Próximo ao ponto de congelamento: Entre 0°C e 5°C

* Abaixo do ponto de congelamento: Menos de 0°C

3. Jantar Especial

Para um jantar em família, além do espaguete, agora você também deseja servir um molho especial para pessoas veganas.

Entrada:

O programa deve perguntar quantos membros da família vão participar do jantar e quantos deles são veganos.

Saída

O programa deve informar a quantidade de espaguete (em gramas) e a quantidade de molho (em litros) que você precisa preparar.

* Espaguete: 100g por pessoa

* Molho: 0,5 litros para cada vegano

Bloco D

Chegamos ao Bloco D, um segmento um pouco mais desafiador, mas que certamente está ao alcance de suas habilidades. Aqui, os algoritmos exigem a implementação de estruturas condicionais, assim como no Bloco C. No entanto, há uma diferença notável: eles são mais detalhados e podem necessitar de uma atenção extra em sua elaboração.

Não deixe a palavra "trabalhoso" intimidar você. Este é apenas um convite para mergulhar um pouco mais fundo em seu pensamento lógico e em sua capacidade de planejamento. Lembre-se de que, em programação, muitas vezes precisamos descompor problemas maiores em partes menores e gerenciáveis. Ao abordar estes algoritmos, pense nas etapas necessárias para chegar à solução final e como as estruturas condicionais podem ajudar a orientar o fluxo do programa.

Ainda que estes algoritmos sejam mais extensos, a essência permanece a mesma: aplicar seus conhecimentos, manter a clareza e a organização em seu código e, acima de tudo, perseverar. Então, respire fundo, mantenha o foco e aproveite a jornada de resolução de problemas. Boa sorte!

Missão Espacial de Abastecimento

Você faz parte da equipe de logística de uma missão espacial que está planejando uma viagem à Lua. Sua tarefa é determinar a quantidade de suprimentos e combustível necessários, considerando vários fatores.

1. ``numAstronautas``: Número de astronautas na missão.
2. ``diasMissao``: Duração total da missão em dias.
3. ``consumoOxigenio``: Consumo diário médio de oxigênio por astronauta em litros.
4. ``consumoAgua``: Consumo diário médio de água por astronauta em litros.
5. ``consumoComida``: Consumo diário médio de comida por astronauta em quilogramas.
6. ``pesoFoguete``: Peso do foguete vazio em quilogramas.
7. ``eficienciaCombustivel``: Eficiência do combustível (quantos quilômetros por litro).
8. ``distanciaLua``: Distância média da Terra à Lua em quilômetros (384.400 km).
9. ``reservaEmergencia``: Porcentagem de suprimentos extras para emergências.
10. ``capacidadeTanque``: Capacidade máxima do tanque de combustível em litros.

Fórmulas:

1. Suprimentos de oxigênio: ``numAstronautas * diasMissao * consumoOxigenio * (1 + reservaEmergencia/100)``
2. Suprimentos de água: ``numAstronautas * diasMissao * consumoAgua * (1 + reservaEmergencia/100)``
3. Suprimentos de comida: ``numAstronautas * diasMissao * consumoComida * (1 + reservaEmergencia/100)``
4. Combustível necessário para a viagem (ida e volta): ``(2 * distanciaLua / eficienciaCombustivel)``

Entrada:

O programa deve coletar todas as 10 variáveis acima do usuário.

Saída:

O programa deve informar:

1. A quantidade total de oxigênio, água e comida necessária para a missão.
2. Se a quantidade de combustível necessária para a viagem (considerando ida e volta) excede a capacidade do tanque.
3. O peso total dos suprimentos e se isso pode ser uma preocupação para o lançamento (uma simplificação: considere um problema se o peso total dos suprimentos exceder 10% do peso do foguete).