



Curso: {NOMECURSO}

Campus: {CAMPUS}

Professor: {NOMEDOCENTE}

Data: {DATAPROVA}

Disciplina: {NOMEDISCIPLINA}

Turma: {TURMA}

Aluno: {NOMEALUNO}

Matrícula: {MATRICULA}

Assinatura: _____

Nota: _____

Avaliação Prova P2 – Introdução a Programação de Computadores

INSTRUÇÕES

1. Essa prova somente tem questões discursivas. Para cada questão você deverá escrever uma função na linguagem de programação Python (versão 3) que resolva o problema de acordo com as especificações descritas na questão.
2. Você poderá usar a Internet durante a resolução da prova. Não será permitido o uso de chats e equivalentes durante a resolução da prova. Também não é permitido consultar os colegas durante a resolução da prova.
3. Leia atentamente as questões antes de respondê-las.
4. Teste o seu código e tenha certeza de que ele está executando corretamente antes de entregar a sua avaliação.
5. A sua resposta da prova deverá ser submetida através de um sistema digital que será informado durante a resolução da prova
6. A sua resposta da prova deverá ser um arquivo texto com extensão .py contendo somente as funções em Python (versão 3) que compõem a sua entrega.

Sobre a Pontuação da Prova

1. Esta avaliação é composta de **10 QUESTÕES DISCURSIVAS**, sendo 3 questões valendo 1,00 ponto , 2 questões valendo 1,50 pontos , 1 questão valendo 2,00 pontos , 1 questão valendo 2,50 pontos e 1 questão valendo 3,00 pontos. Totalizando 13,50 pontos o somatório dos pontos.
2. Não é necessário fazer todas as questões da prova, a sua nota será a somatória dos pontos das questões que você fizer corretamente. A nota máxima da prova é 10,0.

BOA PROVA!

Questão 01 (Valor 1,00 ponto) (OBI2015)

Alice e Bia criaram uma página na Internet com informações sobre o Macaco-prego-de-peito-amarelo, uma espécie em extinção. A página mostra como todos podem ajudar a manter o habitat natural para evitar que a espécie seja extinta.

Uma empresa gostou tanto da iniciativa de Alice e Bia que prometeu doar um prêmio para que as duas amigas possam realizar outras iniciativas semelhantes. A empresa decidiu que o prêmio seria dado quando a soma do número de acessos à página chegasse a 1 milhão.

Dada a lista de acessos diários que ocorreram à página de Alice e Bia, escreva uma função chamada **dias_premio** para determinar quantos dias foram necessários para a soma dos acessos chegar a 1 milhão e as amigas ganharem o prêmio.

A sua função vai receber como argumento uma lista de números inteiros, onde cada número representa o número de acessos em um dia. O primeiro número da lista indica o número de acessos no primeiro dia, o segundo número da lista indica o número de acessos no segundo dia, e assim por diante.

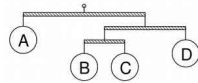
Exemplos:

Chamada da função	Retorno da função
dias_premio ([100,99900,400000,500000,600000])	4
dias_premio ([1,1000000])	2
dias_premio ([100000,250000,250000,500000])	4

Questão 02 (Valor 1,00 ponto) (OBI2015)

O móbile na sala da Maria é composto de três hastes exatamente como na figura abaixo. Para que ele esteja completamente equilibrado, com todas as hastes na horizontal, os pesos das quatro bolas A, B, C e D têm que satisfazer todas as seguintes três condições:

$A = B + C + D$; e
 $B + C = D$; e
 $B = C$



Nesta tarefa, dados os pesos das quatro bolas, você deve desenvolver uma função chamada **em_equilibrio** que deve decidir se o móbile está ou não completamente equilibrado. Retornando True se estiver e retornando False se não estiver. A sua função deve receber quatro argumentos, todos números inteiros, indicando os pesos das bolas. O primeiro argumento é o peso da bola A, o segundo argumento é o peso da bola B, o terceiro argumento é o peso da bola C e o quarto argumento é o peso da bola D

Exemplos:

Chamada da função	Retorno da função
em_equilibrio(12,3,3,6)	True
em_equilibrio(2002,560,560,882)	False
em_equilibrio(500,125,125,250)	True

Questão 03 (Valor 1,00 ponto)

Desenvolva uma função chamada **calcula_juros_compostos** que recebe três argumentos. O primeiro é um número real representando um valor em dinheiro, o segundo é uma string representando a taxa mensal do juros e o terceiro é um número inteiro representando a quantidade de meses do calculo do juros. A função deve retornar o valor do juros de acordo com a formula de *juros compostos* dada a seguir:

$$j = M - C$$
$$M = C \cdot (1 + i)^t$$

Na qual:

j = juros, M = montante, C = capital, i = taxa, t = tempo

Exemplos:

Chamada da função	Retorno da função
<code>calcula_juros_compostos(1000,'5%',2)</code>	102.5
<code>calcula_juros_compostos(2000,'0.5%',10)</code>	102.28
<code>calcula_juros_compostos(830.50,'1%',12)</code>	105.33

Questão 04 (Valor 1,50 ponto) (OBI2017)

Um novo game de realidade aumentada tem, dentro dele, um mini-game que aparece em certas situações para aumentar o ganho de pontos do game principal. O mini-game é um joguinho de memória com quatro cartas, formando dois pares de cartas iguais. Quer dizer, duas cartas têm um número inteiro N marcado em uma de suas faces e as outras duas cartas têm um outro número inteiro M, $N \neq M$. Neste problema, o jogador já virou três cartas, como mostrado na figura.



Claro que, dadas as condições, a carta que falta virar vai formar par com uma das três que já foram viradas. No caso da figura, o número marcado na carta que ainda falta virar é 11. Implemente uma função chamada **carta_faltante** que, recebe três argumentos (os números de três cartas), retorna o número da carta que ainda falta virar!

Exemplos:

Chamada da função	Retorno da função
<code>carta_faltante (40,11,40)</code>	11
<code>carta_faltante (8,8,96)</code>	96
<code>carta_faltante (55,22,22)</code>	55

Questão 05 (Valor 1,50 ponto)

Desenvolva uma função chamada **nome_curto** que recebe uma string representando o nome completo de uma pessoa. A função deve então retornar uma string com o valor do primeiro e último nomes separados por um espaço.

Exemplos:

Chamada da função	Retorno da função
<code>nome_curto('Edgar Allan Poe')</code>	'Edgar Poe'
<code>nome_curto('John Ronald Reuel Tolkien')</code>	'John Tolkien'
<code>nome_curto('Terence David John Pratchett')</code>	'Terence Pratchett'

Questão 06 (Valor 2,00 ponto)

Desenvolva uma função chamada **verifica_hora** que recebe uma string no formato “HH:MM:SS” onde HH são os dígitos para horas, MM são os dígitos para minutos e SS são os dígitos para segundos. Sua função deve retornar True se for uma hora válida e False se for uma hora inválida.

Exemplos:

Chamada da função	Retorno da função
verifica_hora('15:30:00')	True
verifica_hora('15:00:61')	False
verifica_hora('99:99:00')	False

Questão 07 (Valor 2,50 ponto)

Desenvolva uma função chamada **ano_bissexto** que recebe como argumento um número inteiro positivo representando um ano. Sua função deve retornar True o ano for bissexto e False se não for uma ano bissexto. Para calcular se um ano é bissexto você deve seguir as regras abaixo:

1. Os anos divisíveis por 4 são bissextos
2. com a exceção dos anos divisíveis por 100, que não são bissextos
3. com a exceção dos anos divisíveis por 400, que são bissextos

Exemplos:

Chamada da função	Retorno da função
ano_bissexto(2020)	True
ano_bissexto(2019)	False
ano_bissexto(1600)	True

Questão 08 (Valor 3,00 ponto) (OBI2017)

A loja do Pará, especializada em vendas pela internet, está desenvolvendo drones para entrega de caixas com as compras dos clientes. Cada caixa tem a forma de um paralelepípedo reto retângulo (ou seja, no formato de um tijolo).

O drone entregará uma caixa de cada vez, e colocará a caixa diretamente dentro da casa do cliente, através de uma janela. Todas as janelas dos clientes têm o formato retangular e estão sempre totalmente abertas. O drone tem um aplicativo de visão computacional que calcula exatamente as dimensões H e L da janela. O drone consegue colocar a caixa através da janela somente quando uma das faces da caixa está paralela à janela, mas consegue virar e rotacionar a caixa antes de passá-la pela janela.

O aplicativo de controle do drone está quase pronto, mas falta um pequeno detalhe: um programa que, dadas as dimensões da maior janela do cliente e as dimensões da caixa que deve ser entregue, determine se o drone vai ser capaz de entregar a compra (pela janela) ou se a compra terá que ser entregue por meios normais.

Você deve desenvolver uma função chamada **passa_pela_janela** que recebe cinco argumentos, cada um sendo um número inteiro. Os três primeiros contêm os valores A, B, C, indicando as três dimensões da caixa, em centímetros. Os dois últimos contêm os valores H e L, indicando a altura e a largura da janela, em centímetros.

Sua função deve retornar True se a caixa passa pela janela e deve retornar False caso contrário.

Exemplos:

Chamada da função	Retorno da função
<code>passa_pela_janela(30,50,80,80,60)</code>	True
<code>passa_pela_janela(75,100,50,100,30)</code>	False
<code>passa_pela_janela(20,22,5,20,10)</code>	True