



GOVERNO DO ESTADO
SÃO PAULO

FATEC COTIA

Aluno (a):			R.A.:		
Curso: Desenvolvimento de Soft.Multip			Disciplina : Algoritmos e Lógica de programação		
Semestre Letivo: 2024/2	Período:	Bimestre: 1º	Prof.: Dr. Alexandre I. Hashimoto		
Boa Avaliação	Data da Avaliação:		Nota		Visto do Aluno:
	__/__/2024				

- A prova é individual e as respostas devem ser feitas no colab , tirar foto e anexar as fotos e arquivo ipynb no teams
-
- Leia atentamente a prova e responda como solicitado.

Avaliação 2 (4 partes)

Parte 1: Conceitos Básicos (12 pontos) (em python por exemplo)

1. **(3 pontos)** Explique a diferença entre uma lista, uma tupla e um dicionário em Python. Dê exemplos de como cada estrutura de dados pode ser utilizada.
2. **(3 pontos)** O que são operadores lógicos e quais são os operadores lógicos disponíveis em Python? Dê exemplos de como usá-los em expressões condicionais.
3. **(3 pontos)** Descreva a diferença entre as estruturas de repetição `for` e `while` em Python. Dê um exemplo de um problema que seria mais adequado de ser resolvido com um laço `for` e outro exemplo com um laço `while`.
4. **(3 pontos)** Explique o conceito de escopo de variável em Python. O que são variáveis locais e globais? Dê um exemplo de código que demonstre a diferença entre elas.

Parte 2: Aplicação de Conceitos (58 pontos)

1. **(5 pontos)** Escreva um programa que recebe do usuário uma lista de números inteiros e retorna uma nova lista contendo apenas os números primos.
 - **Dica:** Um número primo é um número natural maior que 1 que possui apenas dois divisores distintos: o número 1 e ele mesmo.
2. **(10 pontos)** Crie um programa que simule um sistema de gerenciamento de estoque. O programa deve permitir ao usuário:
 - Adicionar novos produtos ao estoque, informando nome, quantidade e preço.
 - Remover produtos do estoque, informando o nome e a quantidade a ser removida.
 - Consultar a quantidade em estoque de um produto específico.
 - Listar todos os produtos em estoque com suas respectivas quantidades e preços.
 - **Dica:** Utilize um dicionário para armazenar os produtos em estoque, onde a chave é o nome do produto e o valor é uma lista contendo a quantidade e o preço.

3. **(15 pontos)** Implemente um programa que permita ao usuário jogar o jogo da forca. O programa deve:
- Escolher aleatoriamente uma palavra de uma lista predefinida.
 - Exibir a palavra com as letras ocultas, representadas por underlines.
 - Solicitar ao usuário que digite uma letra.
 - Se a letra estiver presente na palavra, revelá-la nas posições corretas.
 - Se a letra não estiver presente na palavra, desenhar uma parte da forca.
 - O jogo termina quando o usuário adivinhar a palavra ou for enforcado (após um número determinado de erros).
 - **Dica:** Utilize uma lista para armazenar as letras da palavra e outra lista para armazenar as letras que já foram adivinhadas.
4. **(10 pontos)** Escreva uma função que recebe duas listas como parâmetros e retorna uma nova lista contendo apenas os elementos que estão presentes em ambas as listas.
- **Dica:** Utilize laços de repetição e estruturas condicionais para comparar os elementos das listas.
5. **(10 pontos)** Crie um programa que solicite ao usuário dois números inteiros positivos, N e M, que representem as dimensões de uma matriz. O programa deve então solicitar ao usuário que digite N x M números inteiros, armazená-los em uma matriz e, como saída, imprimir o maior número digitado e sua posição (linha e coluna) na matriz. ([2])
6. **(8 pontos)** Implemente um programa em Python para criar as seguintes matrizes:
- Uma matriz 3x3 preenchida com zeros.
 - Uma matriz 4x4 preenchida com números aleatórios e dizer o maior número e em qual posição está, a media aritmética da 3 linha e da 2 coluna

Parte 3: Desafio (20 pontos)

1. **(30 pontos)** Implemente um programa que simule um sistema de gerenciamento de tarefas. O programa deve permitir ao usuário:
- Adicionar novas tarefas, informando nome, descrição e prazo.
 - Marcar tarefas como concluídas.
 - Listar todas as tarefas, separando-as por status (concluídas e pendentes).
 - Filtrar tarefas por prazo.
 - Editar informações de uma tarefa existente.
 - **Dica:** Utilize uma lista de dicionários para armazenar as tarefas. Cada dicionário deve conter as chaves nome, descrição, prazo e status.

Parte 4: Desafio (10 pontos)

Colocar as respostas da IA (exemplo - copilot)

Observações:

- A prova é individual.
- As respostas devem ser feitas no Google Colab, e o arquivo .ipynb deve ser submetido no Microsoft Teams.
- Leia atentamente a prova e responda como solicitado.

Boa Prova!