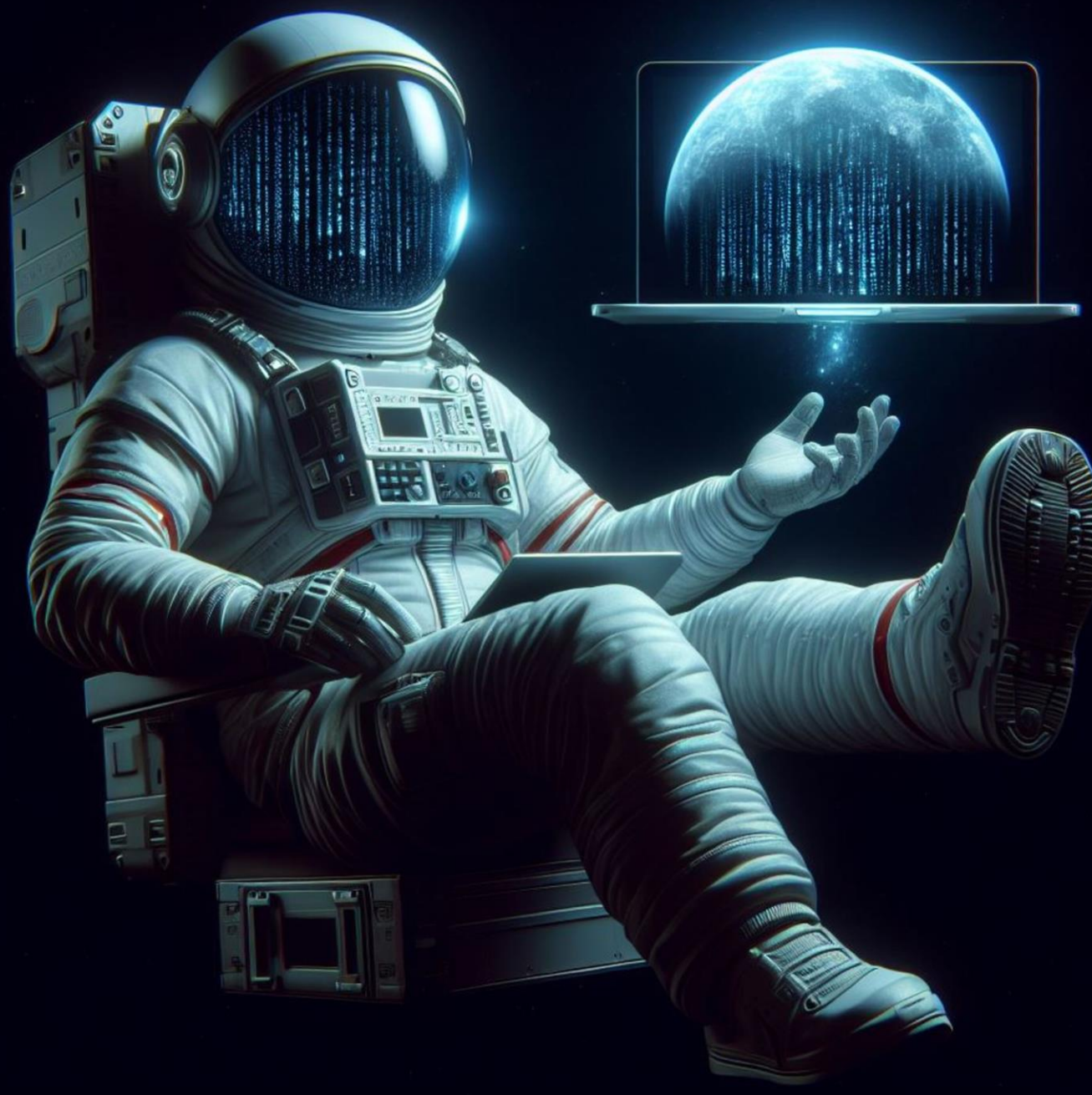
ALGORITMOS DE BUSCA

O GUIA PARA NAVEGAR NO UNIVERSO DA PROGRAMAÇÃO



CAMILA V. MATOS

01

INTRODUÇÃO

Introdução

Os algoritmos de busca desempenham um papel fundamental na ciência da computação, permitindo que os programadores encontrem rapidamente informações em grandes conjuntos de dados.

Neste guia, exploraremos os principais algoritmos de busca, sua explicação simplificada e exemplos de uso em contextos reais.

02

BUSCA LINEAR: A JORNADA PELO DADO PERDIDO

RESUMO

A busca linear é o método mais simples de busca, percorrendo cada elemento da lista até encontrar o que está sendo procurado.

EXPLICAÇÃO

- Percorre cada elemento da lista sequencialmente.
- Verifica se o elemento atual é igual ao que está sendo procurado.
- Retorna o índice do elemento se encontrado, ou -1 se não encontrado.

EXEMPLO DE CÓDIGO

Python

```python
def busca_linear(lista, alvo):
    for i in range(len(lista)):
        if lista[i] == alvo:
            return i
    return -1


# Exemplo de uso
lista = [3, 7, 1, 9, 5]
alvo = 9
print(busca_linear(lista, alvo)) # Saída: 3
```


03

BUSCA BINÁRIA: A BUSCA INTELIGENTE

RESUMO

A busca binária é um algoritmo eficiente para encontrar um elemento em uma lista ordenada, reduzindo pela metade o espaço de busca a cada iteração.

EXPLICAÇÃO

- Divide repetidamente a lista ao meio e verifica se o elemento está na metade esquerda ou direita.
- Repete o processo até encontrar o elemento desejado ou determinar que ele não está na lista.

EXEMPLO DE CÓDIGO

Python

```python
def busca_binaria(lista, alvo):
    esquerda, direita = 0, len(lista) - 1
    while esquerda <= direita:
        meio = (esquerda + direita) // 2
        if lista[meio] == alvo:
            return meio
        elif lista[meio] < alvo:
            esquerda = meio + 1
        else:
            direita = meio - 1
    return -1


# Exemplo de uso
lista = [1, 3, 5, 7, 9]
alvo = 7
print(busca_binaria(lista, alvo))  # Saída: 3
```


04

BUSCA EM PROFUNDIDADE (DFS): EXPLORANDO NOVOS CAMINHOS

RESUMO

A busca em largura é um algoritmo para percorrer ou pesquisar itens em uma estrutura de dados em forma de árvore ou grafo.

EXPLICAÇÃO

- Começa pelo nó raiz e explora todos os vizinhos do nível mais próximo antes de passar para o próximo nível.
- Utiliza uma fila para controlar os nós a serem explorados.
- Garante que todos os nós de um mesmo nível sejam visitados antes de passar para o próximo nível.

EXEMPLO DE CÓDIGO

Python

```python
def DFS(grafo, inicio, alvo, visitados=None):
    if visitados is None:
        visitados = set()
    visitados.add(inicio)

    if inicio == alvo:
        return True
    for vizinho in grafo[inicio]:
        if vizinho not in visitados:
            if DFS(grafo, vizinho, alvo, visitados):
                return True
    return False

# Exemplo de uso
grafo = {'A': ['B', 'C'],
         'B': ['D', 'E'],
         'C': ['F'],
         'D': [],
         'E': ['F'],
         'F': []}

print(DFS(grafo, 'A', 'F'))  # Saída: True
```


05

BUSCA EM LARGURA (BFS): EXPANDINDO HORIZONTES

RESUMO

A busca em profundidade é um algoritmo para percorrer ou pesquisar itens em uma estrutura de dados em forma de árvore ou grafo.

BUSCA EM PROFUNDIDADE

EXPLICAÇÃO

Começa pelo nó raiz e explora o máximo possível ao longo de cada ramificação antes de retroceder.
Utiliza uma pilha para controlar os nós a serem explorados.
Garante que todos os nós de uma ramificação sejam visitados antes de retroceder para explorar outros ramos.

EXEMPLO DE CÓDIGO

Python

```python
from collections import deque

def BFS(grafo, inicio, alvo):
    visitados = set()
    fila = deque([inicio])

    while fila:
        vertice = fila.popleft()
        if vertice == alvo:
            return True
        if vertice not in visitados:
            visitados.add(vertice)
            fila.extend(grafo[vertice])

    return False

# Exemplo de uso
grafo = {'A': ['B', 'C'],
         'B': ['D'],
         'C': [],
         'D': [],

print(BFS(grafo, 'A', 'D'))  # Saída: True
```


OBRIGADO POR LER ATÉ AQUI

ESSE EBOOK FOI GERADO POR IA E DIAGRAMADO POR HUMANO

ESSE CONTEÚDO NÃO FOI GERADO PARA FINS DIDÁTICOS E NÃO FOI REALIZADO UMA VALIDAÇÃO CUIDADOSA NO CONTEÚDO E PODE CONTER ERROS GERADOS POR IA

CAMILA VANESSA DE MATOS SOUSA