Relatório Projeto 2 AED 2023-2024

Nome: Alexandre José Martins Rodrigues Nº Estudante:2022249408

PL (inscrição): 1

Registar os tempos computacionais das 3 soluções. Os tamanhos das arrays (N) devem ser: 20000, 40000, 60000, 80000, 100000. Só deve ser contabilizado o tempo do algoritmo. Exclui-se o tempo de leitura do input e de impressão dos resultados. Devem apresentar e discutir as regressões para as 3 soluções, incluindo também o coeficiente de determinação/regressão (r quadrado).

**Tabela para as 3 soluções Gráfico para a solução A**

Uma imagem com texto, captura de ecrã, Tipo de letra, número

Descrição gerada automaticamente

**Gráfico para a solução B Gráfico para a solução C**

**Análise dos resultados tendo em conta as regressões obtidas e como estas se comparam com as complexidades teóricas:**

Começando pela abordagem exaustiva, que utiliza uma regressão quadrática, era de esperar que fosse a mais lenta. Após analisar os dados, confirmou-se essa expectativa.

Em contraste, a abordagem que utiliza uma regressão NlogN demonstrou tempos intermédios, conforme previsto. Por fim, o terceiro caso, utilizando uma regressão linear, foi o mais rápido, dado que apresenta a menor complexidade temporal.

Esta análise revela uma clara melhoria na eficiência à medida que se reduz a complexidade do algoritmo utilizado.

import time

import random

def crial(tam, k):

lista = []

for i in range(tam):

lista.append(random.randint(1, k))

return lista

def exaustiva(lista, k):

for i in range(len(lista)):

for j in range(1, len(lista)):

if k == lista[i] + lista[j]:

return True

return False

def ordenada(lista, k):

lista.sort()

tam = -1

n = 0

while(1):

while k < lista[n] + lista[tam]:

tam -= 1

while k > lista[n] + lista[tam]:

n += 1

if k == lista[n] + lista[tam]:

return True

if n >= tam:

return False

def elaborada(lista,l):

reserve = set(lista)

for i in lista:

for k in reserve:

if i == k:

return True

return False

def calcula\_tempo(k):

tamanhos = [20000, 40000, 60000, 80000, 100000]

for tamanho in tamanhos:

lista = crial(tamanho, k)

tempos\_1 = []

tempos\_2 = []

tempos\_3 = []

for \_ in range(10):

tempo\_inicial\_1 = time.time()

exaustiva(lista, k)

tempo\_final\_1 = time.time()

tempos\_1.append(tempo\_final\_1 - tempo\_inicial\_1)

tempo\_inicial\_2 = time.time()

ordenada(lista, k)

tempo\_final\_2 = time.time()

tempos\_2.append(tempo\_final\_2 - tempo\_inicial\_2)

tempo\_inicial\_3 = time.time()

elaborada(lista, k)

tempo\_final\_3 = time.time()

tempos\_3.append(tempo\_final\_3 - tempo\_inicial\_3)

resultados = {

'Tamanho da entrada': tamanho,

'Algoritmo 1': sum(tempos\_1) / len(tempos\_1),

'Algoritmo 2': sum(tempos\_2) / len(tempos\_2),

'Algoritmo 3': sum(tempos\_3) / len(tempos\_3)

}

print(resultados)

calcula\_tempo(181)