Proyecto de curso

Moderando el extremismo de opiniones en una red social

realizado por:
[1]Juan Camilo Narvaez Tascón - 202140112,
[2]Julián Ernesto Puyo Mora - 202226905,
[3]Cristian David Pachecho Torres - 20222743,
[4]Juan Sebastián Molina Cuéllar - 202224491

Informe realizado para el curso de Análisis y Diseño de Algoritmos II,
Profesor: Juan Francisco Díaz Frías - Profesor: Jesús Alexander Aranda
Monitor: Mauricio Muñoz

de la

Escuela de Ingeniería de Sistemas y Computación,
Facultad de Ingeniería,
Universidad del Valle

[1]*juan.narvaez.tascon@correounivalle.edu.co,* [2]*julian.puyo@correounivalle.edu.co,*
[3]*cristian.pacheco@correounivalle.edu.co,* [4]*juan.sebastian.molina@correounivalle.edu.co*
September 1, 2024

Contents

1 Introducción

2 Solución con Fuerza Bruta

"Entiende correctamente el algoritmo de fuerza bruta propuesto, calcula correctamente su complejidad y la justifica, analiza acertadamente si el algoritmo da siempre la respuesta correcta y lo justifica."

2.1 Descripción del Algoritmo

2.2 Complejidad

2.3 Corrección

3 Solución con Algoritmo Voraz

"Describen correctamente un algoritmo voraz para resolver el problema, calcula correctamente su complejidad y la justifica, y analiza acertadamente si el algoritmo da siempre la respuesta correcta y lo justifica. El algoritmo quedó clasificado en el grupo de los mejores voraces del curso."

3.1 Descripción del Algoritmo

3.2 Análisis del Algoritmo

3.3 Complejidad

3.4 Corrección

4 Solución con Programación Dinámica

"Usa la programación dinámica correctamente, lo que significa: (1) Caracteriza correctamente la estructura de una solución óptima, (2) Define recursivamente y de forma correcta el costo de una solución óptima para cada subproblema, (3) Implementa correctamente el cálculo del costo de una solución óptima, (4) Implementa correctamente el cálculo de una solución de costo óptimo, (5) Calcula correctamente la complejidad en tiempo de la solución, (6) Calcula correctamente la complejidad en espacio de la solución, y (7) Analiza correctamente si la solución provista es útil en la práctica."

4.1 Caracterización de la Estructura de una Solución Óptima

4.2 Definición Recursiva del Valor de una Solución Óptima

4.3 Algoritmo para Calcular el Costo de una Solución Óptima

4.4 Algoritmo para Calcular una Solución Óptima

4.5 Complejidad

5 Comparación de Estrategias

USAR TABLAS DISEÑAR CASOS DE PRUEBA DOCUMENTAR CÓMO SE DISEÑÓ "Desarrolla tablas que permiten comparar la complejidad y la cercanía al óptimo de los diferentes enfoques al problema (Fuerza Bruta, voraz, y programación dinámica)"

6 Conclusión

CONCLUIR SOBRE LAS VENTAJAS Y DESVENTAJAS DE USAR EN LA PRÁCTICA LOS DIFERENTES ENFOQUES "Desarrolla una sección de conclusiones donde concluye coherentemente, a partir de sus datos, qué ventajas y desventajas encontró al usar diferentes estrategias de diseño."

7 EJEMPLOS TABLAS CODIGO ETC... BORRAR AL FINAL

```go
package main

import (
  "fmt"
)
func binarySearch(arr []int, target int) int {
  left, right := 0, len(arr)-1

  for left <= right {
    mid := left + (right-left)/2
    if arr[mid] == target {
      return mid
    }

    // Si el target es mayor, ignora la mitad izquierda
    if arr[mid] < target {
      left = mid + 1
    } else {
      // Si el target es menor, ignora la mitad derecha
      right = mid - 1
    }
  }

  // Si no encuentra el target
  return -1
}

func main() {
  arr := []int{2, 3, 4, 10, 40}
  target := 10
  result := binarySearch(arr, target)

  if result != -1 {
    fmt.Printf("Elemento encontrado en el indice %d\n", result)
  } else {
    fmt.Println("Elemento no encontrado en el arreglo")
  }
}
```

Listing 1: Algoritmo de búsqueda binaria en Go

ID	Nombre	Calificación
1	Ana Gómez	95
2	Carlos Pérez	88
3	Laura Rodríguez	76
4	José Martínez	84
5	María Fernández	91

Table 1: Tabla de ejemplo

ID	Nombre	Calificación
1	Ana Gómez	95
2	Carlos Pérez	88
3	Laura Rodríguez	76

Table 2: Estudiantes del Grupo A

ID	Nombre	Calificación
1	José Martínez	84
2	María Fernández	91
3	Pedro Ramírez	87

Table 3: Estudiantes del Grupo B

Table 4: EJEMPLO 2 TABLAS LADO A LADO

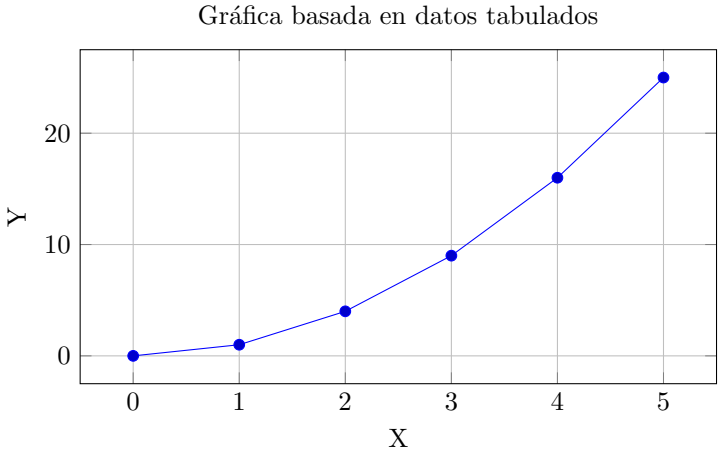

Figure 1: Ejemplo de gráfica creada a partir de una tabla de datos

EJEMPLO DE URL (LINK)