Instituto Politécnico Nacional

UNIDAD PROFESIONAL INTERDISCIPLINARIA DE
INGENIERÍA ZACATECAS

PRÁCTICA 01: ANÁLISIS DE CASOS

Docente: M. en C. Erika Sánchez-Femat

Lucia Iveth De La Vega Hernández
3CM2

18 Septiembre 2023

Introducción

En esta práctica se comprenderá el comportamiento del método de burbuja y un derivado que es burbuja optimizado. En los métodos se analizarán los tres tipos de casos: mejor, peor y promedio; registrando el tiempo de ejecución para posteriormente entender el funcionamiento.

Objetivo de la Práctica

El objetivo principal de esta práctica es que los estudiantes adquieran una comprensión sólida y profunda de los conceptos clave relacionados con la complejidad computacional de los algoritmos. Los tres casos prin- cipales a analizar, mejor caso, peor caso y caso promedio, permiten una evaluación completa del rendimiento de un algoritmo bajo diferentes circunstancias.

Objetivos Particulares

- Comprender la Variabilidad de la Ejecución: Los algoritmos pueden comportarse de manera diferente según las características de los datos de entrada. Al analizar el mejor, peor y caso promedio, los estudiantes aprenderán a apreciar cómo un algoritmo puede variar en su rendimiento.
- Aplicar Conceptos de Complejidad Computacional: La práctica ayudará a los estudiantes a aplicar conceptos fundamentales de la teoria de la complejidad computacional, como la notación O- grande (Big O), para describir y comparar el rendimiento de los algoritmos.
- Desarrollar Habilidades de Análisis Crítico: A través del análisis de diferentes casos, los estu- diantes mejorarán sus habilidades de análisis crítico, aprendiendo a identificar situaciones en las que un algoritmo puede ser más eficiente o ineficiente.
- Preparación para Desarrollo de Algoritmos Eficientes: Al comprender los casos de mejor, peor y caso promedio, los estudiantes estarán mejor preparados para diseñar y seleccionar algoritmos eficientes en situaciones reales, lo que es fundamental en la resolución de problemas computacionales.
- Promover la Resolución de Problemas: A través de la práctica, los estudiantes aprenderán a abordar y resolver problemas computacionales de manera más efectiva, seleccionando o adaptando algoritmos en función de las restricciones de tiempo y recursos.

Desarrollo de la Práctica

Implementación de los algoritmos

Escribe en python los siguientes métodos de ordenamiento:

- Burbuja
- Burbuja Optimizada

Cada método de ordenamiento deberá desarrollarse en una función de programación diferente, con el objetivo de que el usuario, mediante un menú, pueda seleccionar cuál método de ordenamiento desea utilizar. Una vez que el usuario seleccione el método de ordenamiento, el programa pedirá al usuario ingresar el tamaño de la lista y los elementos de la misma, los cuales serán la entrada de los algoritmos.

Análisis de Casos

Desarrollar un reporte con lo siguiente:

- Calcular el mejor, peor y caso promedio para ambos algoritmos de ordenamiento, dando 3 ejemplos de datos de entrada para cada caso.

  Mejor Caso
  En general, el mejor caso será en el que todos los elementos de la lista ya estén ordenados. De este modo el programa no tiene necesidad de hacer cambios, solo entrar al ciclo y al corroborar que está bien ordenado terminar de ejecutarse.

  - *E*jemplo 01:
    elementos = [2,5,7,8,9,12]
  - *E*jemplo 02:
    elementos = [4,16,64]
  - *E*jemplo 03:
    elementos = [1,2,3,4,5,6,7,8,9,10]

  Peor Caso
  El peor caso sucederá cuando la lista esté desordenada en su totalidad y requiera de repetir el ciclo muchas veces para lograr ordenar todos los elementos.

  - *E*jemplo 01:
    elementos = [3,7,1,4,2]
    Procedimiento : $[3,7,1,4,2] \rightarrow [3,\mathbf{1,7},4,2] \rightarrow [3,1,\mathbf{4,7},2] \rightarrow [3,1,4,\mathbf{2,7}] \rightarrow [\mathbf{1,3},4,2,7] \rightarrow [1,3,\mathbf{2,4},7] \rightarrow [1,\mathbf{2,3},4,7] \rightarrow$**[1,2,3,4,7]**
  - *E*jemplo 02:
    elementos = [3,2,1]
    Procedimiento : $[3,2,1] \rightarrow [\mathbf{2,3},1] \rightarrow [2,\mathbf{1,3}] \rightarrow [\mathbf{1,2},3] \rightarrow$**[1,2,3]**

- *E*jemplo 03:
  elementos = [52,4,22,1]
  Procedimiento : $[52,4,22,1] \rightarrow [\mathbf{4,52},22,1] \rightarrow [4,\mathbf{22,52},1] \rightarrow [4,22,\mathbf{1,52}] \rightarrow$
  $[4,\mathbf{1,22},52] \rightarrow [\mathbf{1,4},22,52] \rightarrow$**[1,4,22,52]**

Caso Promedio
El caso promedio será cuando la mitad de nuestros elementos estén ordenados, así solo tendrá que ordenar la otra mitad restante.

- *E*jemplo 01:
  elementos = [10,20,30,80,50,40]
  Procedimiento : $[10,20,30,80,50,40] \rightarrow [10,20,30,\mathbf{50,80},40] \rightarrow [10,20,30,50,\mathbf{40,80}] \rightarrow$
  $[10,20,30,\mathbf{40,50},80] \rightarrow$**[10,20,30,40,50,80]**

- *E*jemplo 02:
  elementos = [4,2,0,11,12]
  Procedimiento : $[4,2,0,11,12] \rightarrow [\mathbf{2,4},0,11,12] \rightarrow [2,\mathbf{0,4},11,12] \rightarrow [\mathbf{0,2},4,11,12] \rightarrow$**[0,2,4,11,12]**

- *E*jemplo 03:
  elementos = [3,5,6,9,8,7,10,13]
  Procedimiento : $[3,5,6,9,8,7,10,13] \rightarrow [3,5,6,\mathbf{8,9},7,10,13] \rightarrow [3,5,6,8,\mathbf{7,9},10,13] \rightarrow$
  $[3,5,6,\mathbf{7,8},9,10,13] \rightarrow$**[3,5,6,9,8,7,10,13]**

- Encontrar y justificar a cuál clase de las siguientes pertence cada caso de cada algoritmo:
  O(n^2): El algoritmo de burbuja tiene un rendimiento de O(n^2). Independientemente de si este o no ordenada la lista, siempre será ese el rendimiento debido a que necesita realizar una iteración por todos los elementos en cada uno de los elementos a procesar.

Comparación de Resultados

El tiempo en cuanto al mejor y peor caso son relativamente similares. En el mejor usando burbuja se obtuvo 1.4781e-05 y con la burbuja optimizada 1.1920e-05. Para el peor caso con burbuja se ejecutó en 1.6927e-05 y en burbuja optimizada 1.6450e-05. Por último, en el promedio con burbuja se obtuvo 8.1062e-6 y con el optimizado 1.1444e-05.

Entrega de Resultados

Se deberá desarrollar un reporte en Overleaf con Portada, Introducción, Desarrollo, Conclusiones y Referencias. En la sección de desarrollo se deberá explicar el procedimiento de implementación de ambos algoritmos, así como el desarrollo de los puntos 2.2 y 2.3.

El código en python y el archivo PDF del reporte se deberán subir al repositorio que cada uno de los alumnos tiene en Github.

Conclusión

Analizando los resultados, en general se esperaría que al estar todos los elementos ordenados sea el mejor caso, sin embargo el mejor tiempo fue en el caso promedio utilizando método de la burbuja optimizado; esto ya se mencionó en el análisis de su complejidad, ya que sin importar que esté en orden el método recorrerá y ejecutará lo que necesite con cada uno de los elementos. En general los tiempos de ejecución de los tres casos no difieren mucho entre ellos.