Dibujo en blanco y negro

Descripción generada automáticamente con confianza baja**INSTITUTO POLITÉCNICO NACIONAL**

**ESCUELA SUPERIOR DE CÓMPUTO**

**--------------------ANÁLISIS DE ALGORITMOS------------------**

**ACTIVIDAD**

Simulación Producto 2 Mayores

**PROFESOR:**

Franco Martínez Edgardo Adrián

**ALUMNO:**

Meza Vargas Brandon David – 2020630288

**GRUPO:**

3CM13

Joven con camiseta negra

Descripción generada automáticamente

**Índice**

[Problema 3](#_Toc82205617)

[Mejor Caso 4](#_Toc82205618)

[Tabla Comparativa 5](#_Toc82205619)

[Peor Caso 5](#_Toc82205620)

[Tabla Comparativa 6](#_Toc82205621)

[Caso Medio 7](#_Toc82205622)

[Tabla Comparativa 8](#_Toc82205623)

[Instrucciones para compilar y ejecutar 8](#_Toc82205624)

# **Problema**

En el presente ejercicio se pide hacer la simulación del peor, mejor y caso medio del siguiente algoritmo:

Texto

Descripción generada automáticamente

Imagen 1. Algoritmo producto2Mayores.

Realice una modificación a las comparaciones para considerar a los números que se repiten.

Sabemos que las **operaciones básicas** son: **comparación con los elementos del arreglo** y las **asignaciones a los elementos mayores.**

De igual forma conocemos las funciones para el mejor, peor y caso medio:

* **Peor caso: f(n) = 3n – 3**
* **Mejor caso: f(n) = 2n – 1**
* **Caso medio:**

Sabiendo todo lo anterior podemos realizar la simulación realizando los cambios necesarios al código de la imagen 1.

# **Mejor Caso**

El código para este caso se presenta a continuación en la imagen 2.

Texto

Descripción generada automáticamente

Imagen 2. Código mejor caso.

Para que se de el mejor caso, los dos números mayores se deben encontrar en las dos primeras posiciones del arreglo, para esto me auxilie de un archivo que contiene números al azar, de estos números coloque a los dos mayores al inicio. Posteriormente leí el archivo con los números los n números para almacenar en el arreglo.

Para la n 2500, use un archivo que contiene 2500 números, de forma similar, para la n de 5000, use un archivo que contiene 5000 números.

## **Tabla Comparativa**

|  |  |  |
| --- | --- | --- |
| **N** | **Resultados teóricos** | **Resultados Prácticos** |
| 2500 |  |  |
| 5000 |  |  |

# **Peor Caso**

El código para el peor caso se presenta a continuación en la imagen 3.

Texto

Descripción generada automáticamente

Imagen . Código peor caso.

Para que se de el peor caso. los dos números mayores deben estar al final del arreglo, para esto se generaron 2500 números aleatorios y se ordenaron de manera ascendente, de igual forma con los 5000 números. Posteriormente leí el archivo con los números los n números para almacenar en el arreglo.

## **Tabla Comparativa**

|  |  |  |
| --- | --- | --- |
| **N** | **Resultados teóricos** | **Resultados Prácticos** |
| 2500 |  |  |
| 5000 |  |  |

# **Caso Medio**

El código para el caso medio se ve en la imagen 4.

Texto

Descripción generada automáticamente

Imagen . Código del caso medio.

Para obtener el caso medio se hizo el mismo algoritmo del mejor y peor caso, solo que en esta ocasión se hizo n veces, siendo específicos 10, 000 veces como se ve en la imagen 4. De esta forma, al hacerse 10, 000 veces el algoritmo, se pudo obtener el promedio del número de operaciones para el caso medio.

## **Tabla Comparativa**

|  |  |  |
| --- | --- | --- |
| **N** | **Resultados teóricos** | **Resultados Prácticos** |
| 2500 |  |  |
| 5000 |  |  |

# **Instrucciones para compilar y ejecutar**

En las carpetas mejor y peor caso encontramos el archivo con extensión .c, además de dos archivos con números, para compilar usamos:

***gcc nombre.c -o nombreSalida.exe***

Para su ejecución, tenemos que indicar un parámetro que será la n y de que archivo leerá los números:

***nombreSalida.exe n < archivo.txt***

En la carpeta del caso medio solo encontraremos el archivo .c:

***gcc nombre.c -o nombreSalida.exe***

Para ejecutarlo solo basta con indicar el parámetro n:

***nombreSalida.exe n***