Alumno:

Calva Hernández José Manuel 2017630201

Análisis de Algoritmos

M. en C. Edgardo Adrián Franco Martínez  
Grupo: 3CM3  
Fecha: 11 /Marzo/2018

Complejidad temporal y análisis de casos

Ejercicio 02



Instituto Politécnico Nacional

Escuela Superior de Cómputo

Índice

Sección A2

Algoritmo 12

Algoritmo 22

Algoritmo 33

Algoritmo 43

Algoritmo 54

Sección B5

Algoritmo 65

Algoritmo 76

Algoritmo 87

Sección C9

Algoritmo 99

Algoritmo 1010

Algoritmo 1111

Algoritmo 1212

Algoritmo 1313

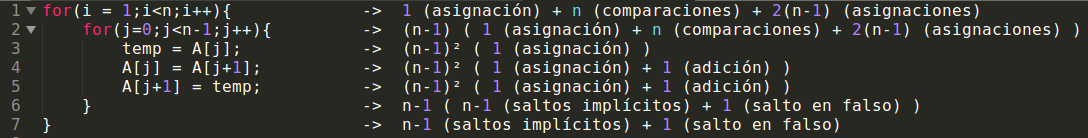
Algoritmo 1414

Algoritmo 1515

# Sección A.

Para los siguientes 5 algoritmos determine la función de complejidad temporal y espacial en términos de n. \*Considere las operaciones de: asignación, aritméticas, condicionales y saltos implícitos.

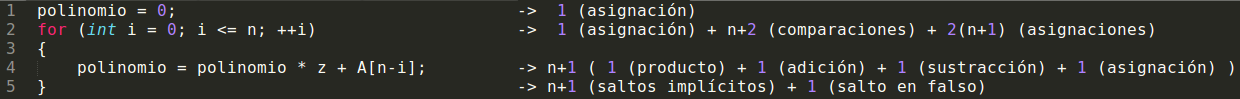
Algoritmo 1



Realizando las operaciones, obtenemos la siguiente fórmula temporal:

Dado que se utiliza una variable auxiliar para guardar los elementos a intercambiar, además de las dos variables para los ciclos y el arreglo que presuntamente tiene tamaño n, la fórmula espacial es:

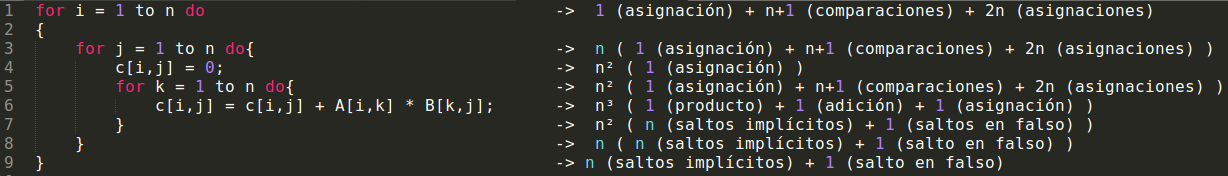
Algoritmo 2



Realizando las operaciones, obtenemos la siguiente fórmula temporal:

Tomaremos en cuenta un arreglo presuntamente de tamaño n, además de una variable auxiliar para ir llevan el resultado, una variable multiplicativa "z" y la variable utilizada para el ciclo; por tanto la fórmula espacial es:

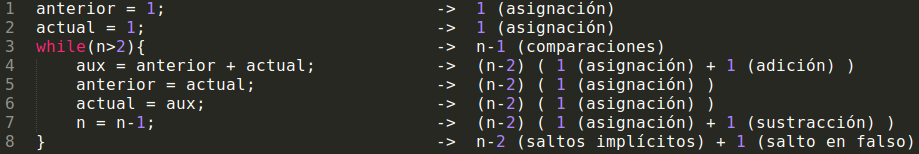
Algoritmo 3



Realizando las operaciones, obtenemos la siguiente fórmula temporal:

Se utilizarán tres variables para los ciclos, además, se tienen tres arreglos bidimensionales, presuntamente de tamaño n\*n, por tanto, la fórmula espacial es:

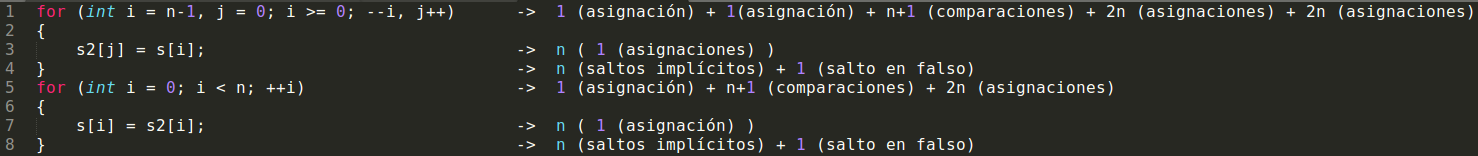
Algoritmo 4



Realizando las operaciones, obtenemos la siguiente fórmula temporal:

Únicamente se utilizarán tres variables durante la ejecución del código, por tanto, la fórmula espacial es:

Algoritmo 5



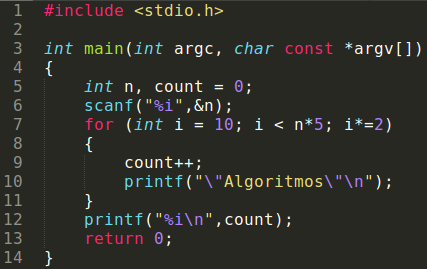
Realizando las operaciones, obtenemos la siguiente fórmula temporal:

Se utiliza una misma variable para ambos ciclos, pero en el primero se tiene otra variable adicional, además, tenemos dos arreglos que asumiremos tienen de tamaño n, así, la fórmula espacial es la siguiente:

Sección B.

Para los siguientes 3 algoritmos determine el número de veces que se imprime la palabra “Algoritmos”. Determine una función lo más cercana a su comportamiento para cualquier n y compruébela codificando los tres algoritmos (Adjuntar códigos). De una tabla de comparación de sus pruebas para n’s igual a 10, 100, 1000, 5000 y 100000 y demostrar lo que la función encontrada determina será el número de impresiones.

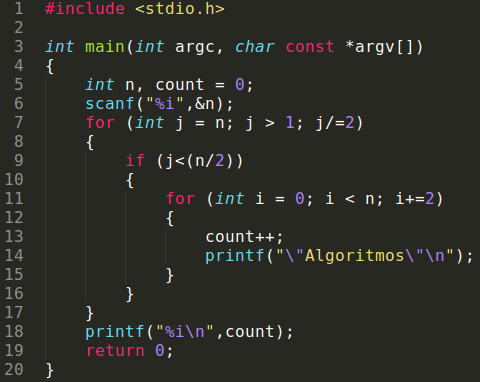
Algoritmo 6



Dado que el ciclo irá aumentando al doble, se considera que este es exponencial. Por tanto, la función principal irá respecto de un logaritmo de base 2, y se considera únicamente la n porque la i está iniciándose en el doble de 5, que es 10. Se le hará función piso a la fórmula debido a que el ciclo nunca llegará a igualarse con el valor de n.

|  |  |  |
| --- | --- | --- |
|  |  | # Impresiones Reales |
| 10 |  | 3 |
| 100 |  | 6 |
| 1000 |  | 9 |
| 5000 |  | 12 |
| 100000 |  | 16 |

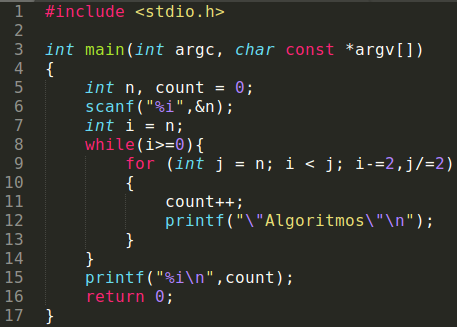
Algoritmo 7



La primera parte corresponde al ciclo interno, debido a que inicia en 0 y va aumentando de 2 en 2, se considera una función techo respecto a la mitad de n. Por otra parte, el ciclo externo se considera que irá reduciendo a la mitad la n, sin embargo, al tener la restricción de menor a su primera mitad y ser mayor a 1, descontaremos dos valores del resultado inicial que nos arroje la función piso sobre el logaritmo.

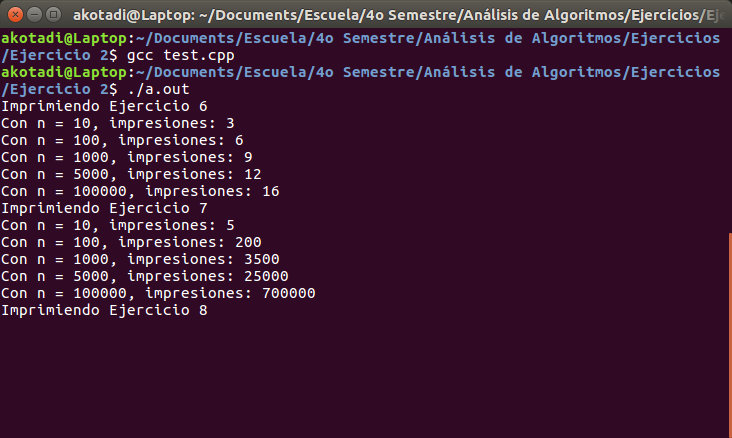
|  |  |  |
| --- | --- | --- |
|  |  | # Impresiones Reales |
| 10 |  | 5 |
| 100 |  | 200 |
| 1000 |  | 3500 |
| 5000 |  | 25000 |
| 100000 |  | 700000 |

Algoritmo 8



Es sencillo ver que este ciclo será infinito siempre que la n sea mayor a 0, sin embargo, nunca va a imprimir nada debido a que la condición del ciclo interno nunca se podrá inicializar al ser tanto la i como la j del mismo valor que n.

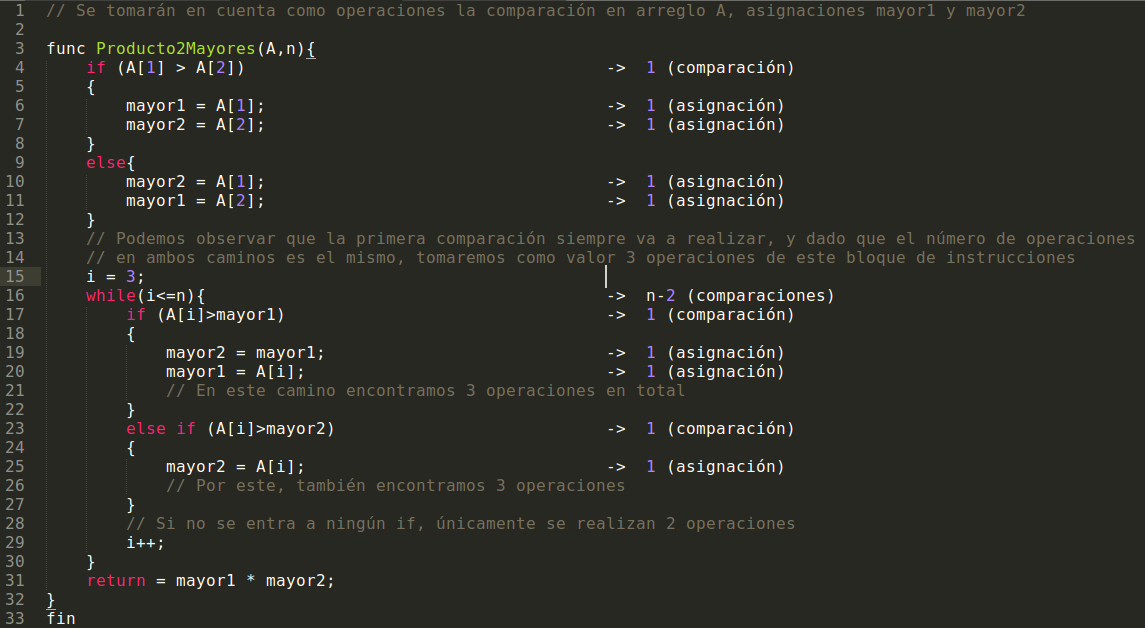
|  |  |  |
| --- | --- | --- |
|  |  | # Impresiones Reales |
| 10 |  | 0 |
| 100 |  | 0 |
| 1000 |  | 0 |
| 5000 |  | 0 |
| 100000 |  | 0 |



Sección C.

Para los siguientes algoritmos determine las funciones de complejidad temporal, para el mejor caso, peor caso y caso medio. Indique cual(es) son las condiciones (instancia de entrada) del peor caso y cual(es) la del mejor caso.

Algoritmo 9



Mejor Caso

Se da cuando los dos primeros elementos del arreglo son los mayores, ya que ello implica que nunca entrará a realizar las operaciones dentro de los ifs, por tanto, su fórmula sería:

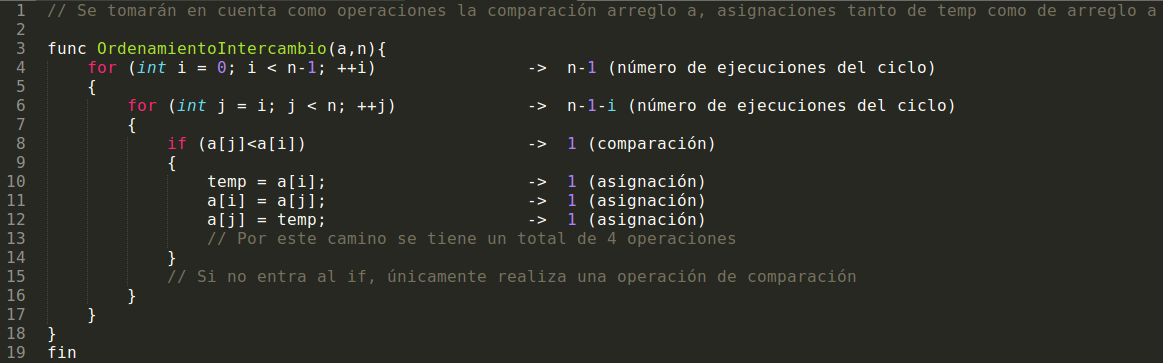
Peor Caso

Lo encontramos cuando el arreglo está ordenado de menor a mayor, o bien, el primer o segundo elemento es uno de los mayores, pero los demás se encuentran ordenamos de forma ascendente. Esto implica que entrará en la parte del código de los ifs y realizará la totalidad de las operaciones posibles, por tanto, su fórmula sería:

Caso Medio

Como se puede observar en las anotaciones del código, encontramos 3 posibles caminos si no tomamos en cuenta la primer parte, debido a que las posibilidades dan el mismo resultado, lo cual no afectará nuestro resultado. Por tanto, suponiendo que la probabilidad de que tome alguno de los tres posibles caminos del ciclo while es igual, consideraremos la siguiente fórmula:

Algoritmo 10



Mejor Caso

Si el arreglo se encuentra ordenando de manera ascendente, las instrucciones dentro del bloque if nunca se ejecutarán, por tanto, su fórmula sería la siguiente:

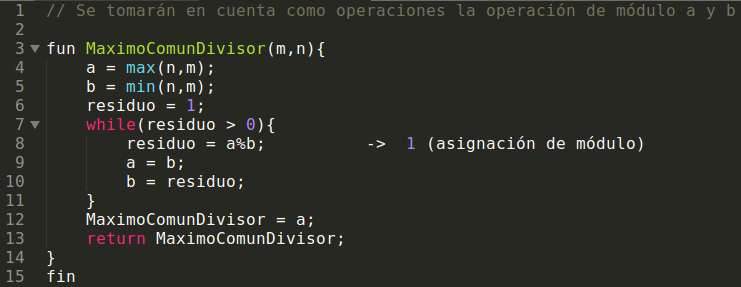
Peor Caso

En caso de que el arreglo se encuentre de manera descendente, se ejecutaría siempre el bloque if, por tanto, la fórmula sería la siguiente:

Caso Medio

En este caso tenemos únicamente dos posibles caminos, uno donde se ejecuta todo el bloque if, y otro donde no lo hace. Suponiendo que ambos tengan la misma posibilidad de ejecutarse, la fórmula sería la siguiente:

Algoritmo 11



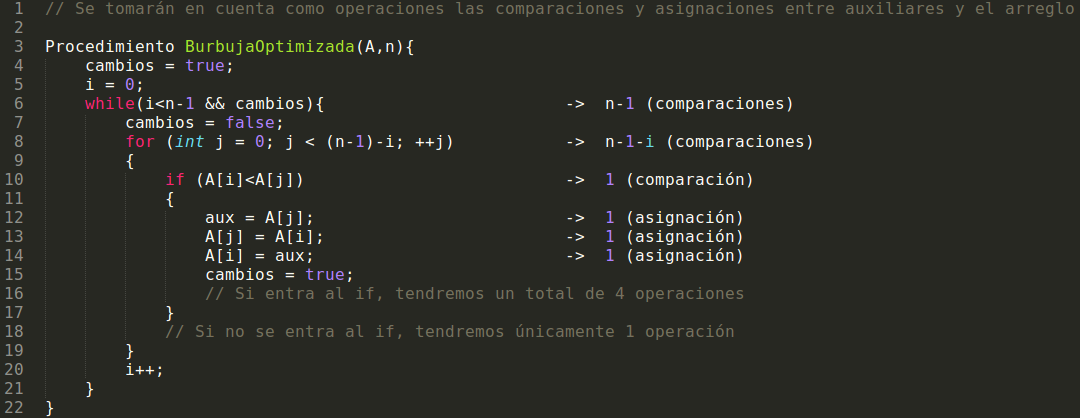
Mejor Caso

El mejor caso se tiene cuando n o m es igual a 1, ya que éste será el MCD y el ciclo sólo se ejecutará una vez, por tanto, la fórmula estaría dada por:

Peor Caso

Éste viene dado por dos números subsecuentes en la serie de Fibonacci debido a que su módulo vendrá a ser el número anterior de la serie y así sucesivamente hasta llegar al MCD. Por tanto, su función sería:

Algoritmo 12



Mejor Caso

Se presenta cuando el arreglo se encuentra ordenado ascendentemente, ya que la bandera de cambios nunca se cambiaría y sólo haría una vez el recorrido externo, considerando que el recorrido interno lo hará de manera completa, tenemos que la función queda de la siguiente manera:

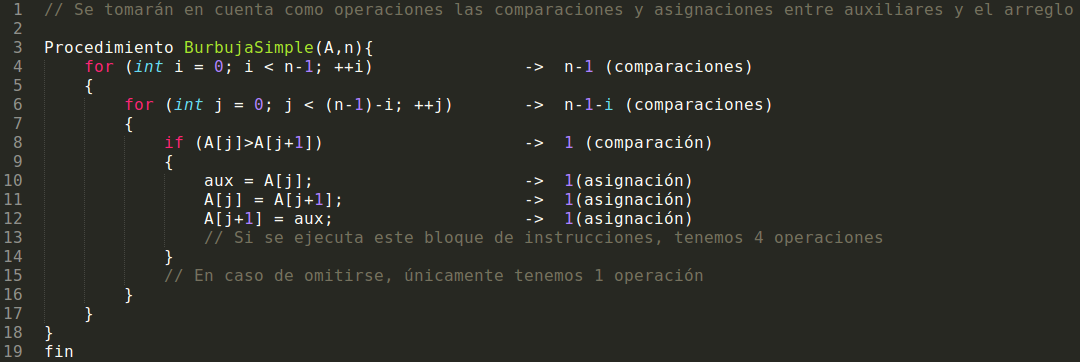
Peor Caso

Cuando el arreglo se encuentra ordenado de manera descendente, ya que en ese caso siempre se ejecutará el bloque de instrucciones dentro del if, y su función temporal quedaría como se muestra a continuación:

Caso Medio

Únicamente se pueden presentar dos casos, que se ejecute el bloque de instrucciones dentro del if, o que no lo haga. Asumiendo que ambos tienen la misma probabilidad de suceder, la fórmula sería:

Algoritmo 13



Mejor Caso

Se presenta cuando el arreglo se encuentra ordenado ascendentemente, por lo tanto se ejecutarán ambos ciclos con una única instrucción dentro, tenemos que la función queda de la siguiente manera:

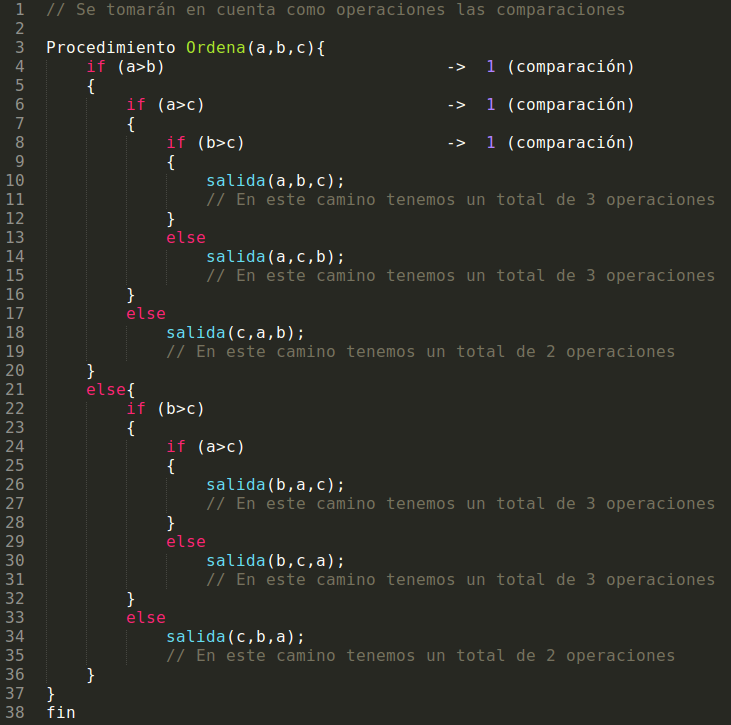
Peor Caso

Cuando el arreglo se encuentra ordenado de manera descendente, en ese caso siempre se ejecutará el bloque de instrucciones dentro del if, y su función temporal quedaría como se muestra a continuación:

Caso Medio

Únicamente se pueden presentar dos casos, que se ejecute el bloque de instrucciones dentro del if, o que no lo haga. Asumiendo que ambos tienen la misma probabilidad de suceder, la fórmula sería:

Algoritmo 14



Mejor Caso

Hay dos posible mejores casos, cuando o , para estos casos la función temporal sería:

Peor Caso

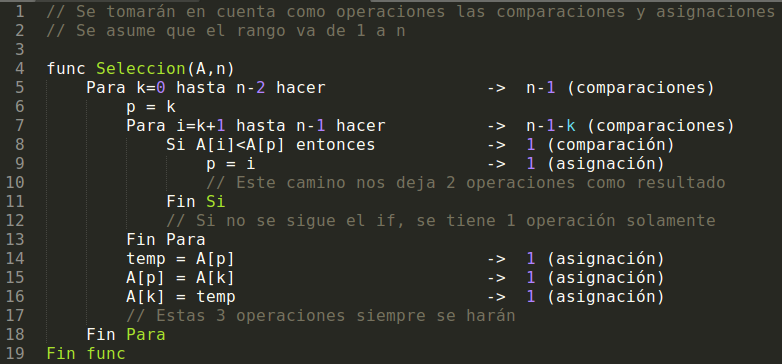
Tenemos cuatro posibles peores casos, estos son:

En estos casos, tenemos que la función temporal está dada por:

Caso Medio

Como es posible apreciarse, tenemos seis posibles caminos dentro del procedimientos, por tanto, asumiremos que tienen la misma probabilidad de suceder y eso nos daría la siguiente función temporal:

Algoritmo 15



Mejor Caso

Cuando el arreglo se encuentra ordenado ascendentemente, nunca se seguirán las instrucciones del if, lo que nos daría como resultado la siguiente función:

Peor Caso

Si se encontrase ordenado de manera descendente, siempre se estaría ejecutando la asignación del if, por tanto, resultaría la siguiente función:

Caso Medio

En este caso tenemos dos posibles caminos, descritos arriba donde podemos o no ejecutar la asignación del bloque if. Asumiendo que tienen la misma posibilidad de ser ejecutados, resulta en un promedio de las dos funciones de arriba: