

Trabajo practico numero 1

Integrantes: Castillo Conrado, Celi Yanina

Materia: Diseño de Algoritmos/Algoritmos II

A continuación, se darán a conocer ciertas características de los programas entregados junto con una descripción de cómo están organizados los archivos, para así saber cómo correr los programas.

Junto con este archive se encuentran dos carpetas, correspondientes a cada ejercicio del proyecto. Dentro de cada carpeta hay dos archivos .java: uno contiene las resoluciones a los incisos y otro los tests para esos incisos. Los nombres de estos archivos y carpetas son bien intuitivos, así que se podrá reconocer a que consigna corresponde cada uno.

Los archivos poseen la documentación necesaria para poder generar javadocs. En cuanto a los códigos de los programas demandados por las consignas del trabajo, dentro las clases que no son tests se encontrarán con los métodos con la misma estructura y características indicadas por los profesores en el proyecto. Estos métodos son los principales y son los que sobre se ejecutan los tests. También se podrán encontrar con métodos auxiliares que asisten a estos métodos principales. Para ninguna de las clases existe un método main, ya que las pruebas sobre los programas se realizan utilizando las clases que contienen los tests. Las clases test, en cada prueba, poseen comentado el resultado que se espera que se devuelva para ese test y esta nombrado con el método al cual se prueba. Para poder realizar las pruebas se deberán compilar la clase que contiene los métodos de la consigna y la clase que contiene los tests. Luego se deberá ejecutar el compilado de la clase de los tests.

Además de esto querríamos comentar algunas cosas con respecto a los tiempos que nos devolvieron los test ejecutados a los diferentes programas:

Para el ejercicio de StringMatching, los resultados de algunas pruebas:

- Con la cadena “hola BUEN dia” el algoritmo de fuerza bruta tardo 0,04 segundos, y con el de KMP tardo tambien 0,04 segundos. En cambio, con la cadena “Alan Turing fue un brillante matemático, criptoanalista e informático teórico nacido el veintitrés de Junio de 1912 en Maida Vale un distrito residencial al oeste de Londres. Turing, ademas de ser un brillante científico era homosexual, lo cual le costó la vida el siete de junio de 1954.” el algoritmo de fuerza bruta tardo 0,018 segundos, mientras que el de KMP tardo 0,03 segundos.

Para el ejercicio de LongestRepetitions, los resultados de algunas pruebas:

- Con la cadena “hola queque taltal comocomo estas”, el algoritmo de fuerza bruta tardo 0,002 segundos, y con el de divide and conquer 0,006 segundos. Pero con la cadena “Alan Turing fue un brillante brillante matemático, criptoanalista criptoanalista e informático teórico nacido nacido el veintitrés de Junio junio de 1912 en Maida Vale un distrito residencial al oeste oeste de Londres. Turing, ademas ademas de ser un brillante científico era homosexual, lo cualcual le costó lala vida vida el siete dede junio de 1954.\”, el algoritmo de fuerza bruta tardo 0,039 segundos , mientras que el de divide and conquer tardo 0,007 segundos.

- Con la cadena “(Karl o Carl Friedrich Gauss; Brunswick, actual Alemania, 1777 - Gotinga, id., 1855) Matemático, físico y astrónomo alemán. Nacido en el seno de una familia humilde, desde muy temprana edad Karl Friedrich Gauss dio muestras de una prodigiosa capacidad para las matemáticas” el algoritmo de fuerza bruta tardó 0,026 segundos y el de divide and conquer tardó 0,012 segundos.

Pudimos notar que cuando realizábamos pruebas con cadenas cortas, por lo general los tiempos entre fuerza bruta y la otra implementación correspondiente al ejercicio eran similares, incluso las implementaciones que no eran fuerza bruta tardaban más. Pero cuando realizamos tests con cadenas más largas, se notó que los algoritmos de Kmp y divide and conquer son más eficientes que los de fuerza bruta.