

PROYECTO COMPUTACIONAL:

Métricas de Calidad de Documentación de Software

ENERO 2013

En computación existen muchas líneas de investigación que enfocan sus estudios e investigaciones en áreas tales como redes, paralelismo, inteligencia artificial, sistemas de información, ingeniería de software, base de datos, etc. En particular, el área de Ingeniería de Software se encarga de estudiar los procesos y elementos involucrados en el desarrollo de sistemas de software; entre los aspectos a considerar está la necesidad de *«producir software en los plazos de tiempo adecuados, de forma eficiente, garantizada, y que el software sea adaptable, fácil de mantener y de usar»*¹, es decir, se requiere estudiar qué tan bueno es el sistema desarrollado y cómo debe hacerse ese desarrollo para que sea de alta calidad. Cuando se habla de calidad del software, este no se limita solo al producto como tal, sino que también se toma en cuenta la calidad de la documentación que acompaña al producto. Dicha documentación se refiere al manual de usuario, manual técnico, diccionario de datos, etc. Ahora bien, debido al auge de la información al estilo hipermidia o hipertexto (páginas informativas que poseen “links” o “enlaces” que permiten llegar a otras páginas distintas) y debido al creciente uso de Internet, se ha hecho cotidiano diseñar esta documentación al estilo hipertexto. Es decir, por ejemplo, los manuales de usuario ahora se desarrollan como documentos con diseño web.

En tal sentido, al estar en una página P_i , se puede acceder a una página P_j , si en la página P_i existe un enlace que lo permita; del mismo modo, siempre se puede acceder a la página inmediatamente anterior (en este caso, regresar de P_j a P_i), utilizando el botón “Atrás” del navegador que se esté empleando. Ahora bien, en Ingeniería del Software han surgido muchas fórmulas para medir la calidad de este tipo de documentación. Una de esas fórmulas se llama *Compactness metric* (C_p) o métrica de compactación, cuyo valor se obtiene de la siguiente forma:

$$C_p = \frac{D - \sum_{i=1}^n \sum_{j=1}^n C_{ij}}{D - d}$$

donde:

$D = (n^2 - n).n$

$d = (n^2 - n)$

$n =$ **número total de páginas** y

$C_{ij} =$ **distancia de acceso** desde la página i a la página j (el menor número de enlaces que se deben visitar para llegar desde la página i a la página j)

Se debe tener en cuenta que la distancia de acceso desde una página P_i a ella misma es cero y que en caso de no poderse llegar desde una página P_i a una página P_j , la distancia de acceso entre ellas es n .

El resultado de esta fórmula debe estar en intervalo $[0,1] \subseteq \mathbb{R}$, donde valores cercanos a 1 indican mayor calidad del documento (mayor compactación) y valores cercanos a 0 indican menor calidad del mismo (menor compactación).

¹ J.J.Dolado & L.Fernández "Medición para la gestión en la Ingeniería del Software", Editorial Ra-Ma, 2000. ISBN 84-7897-403-2

En base a todo lo anterior, se debe desarrollar un programa de computación que permita satisfacer los siguientes requerimientos:

- (a) Calcular el valor de métrica de compactación de la documentación (desarrollada al estilo de hipertexto) de un sistema de software, utilizando la menor cantidad de cálculos posibles.
- (b) Dadas dos páginas de la documentación: P_i y P_j , se desea conocer la distancia de acceso C_{ij} , y las páginas que corresponden a la secuencia de accesos involucradas en esa distancia.

En base a todo lo anterior:

- (a) Modele el problema usando teoría de grafos.
- (b) Construya el (o los) algoritmo(s) que permitan satisfacer los requerimientos planteados. Recuerde que debe validar los datos de entrada.
- (c) Implemente su solución algorítmica en un programa de computación, utilizando el lenguaje de programación C o Java.
- (d) El programa debe permitir la entrada de datos tanto por pantalla como por un archivo.

PRIMERA ENTREGA:

viernes, 08-02-2013

- (1) Modelación del problema, usando las herramientas de la teoría de grafos.
- (2) Diseño de las estructuras de datos (Matrices, vectores, registros, etc.).
- (3) Análisis del problema y solución algorítmica en alto nivel.

SEGUNDA ENTREGA:

viernes, 01-03-2013

El programa y la Documentación que se especifica a continuación debe ser cargada en el entorno virtual:

- (1) *Introducción:* Se hace el planteamiento del problema. Se describen, los objetivos del programa y el ambiente de programación en el cual fue desarrollado (lenguaje de programación, etc.). Se da una visión general del informe indicando las partes, secciones que lo componen.
- (2) *Especificaciones de diseño:*
 - (2.1) Modelación del problema, usando las herramientas de la teoría de grafos.
 - (2.2) Diseño de las estructuras de datos (Matrices, vectores, registros, etc.).
 - (2.3) Análisis del problema y solución algorítmica en alto nivel.
 - (2.4) Algoritmos en bajo nivel, con una descripción clara de los procedimientos y funciones, mediante el uso de comentarios.
- (3) *Limitaciones:* Se indican las posibles restricciones que pueda tener el programa. Por ejemplo, el número máximo de páginas que maneja el programa es de 100.
- (4) *Análisis Final:* Señalando los datos de prueba utilizados por usted para comprobar la efectividad del programa, cuáles fueron los resultados obtenidos y dando sugerencias sobre qué mejorar a su programa en versiones posteriores.

Consideraciones para la documentación: Trate de ser claro y conciso.

Ítem:	Número máximo de páginas:
(1)	1 pag.
(2)	15 pags.
(3)	1 pag.
(4)	3 pags.

DEFENSA DEL PROYECTO: **04-03-2013 al 08-03-2013**

Oportunamente se publicará el día y hora que corresponda a cada equipo.