

EVALUACIÓN COMPARATIVA DE LOS LENGUAJES LEXICO Y C# EN EL APRENDIZAJE DE LA PROGRAMACIÓN ORIENTADA A OBJETOS

Sandra María Morales Múnera ¹, Alexander Jaramillo ², Fabián Ríos ³

¹Licenciada en Matemáticas y Física, Ing. de Sistemas, Especialista en Computadores para la Docencia, Docente de la Institución Educativa José Miguel de Restrepo y Puerta. smoralesmunera@gmail.com

²Estudiante de Ingeniería de Sistemas, Universidad de Antioquia. Auxiliar de investigación y monitor de las cátedras de algoritmos y programación de computadores, Departamento de Ingeniería de Sistemas. alj5@yahoo.es

³Ing. Industrial, Magister en Educación. Profesor de algoritmos y programación del departamento de ingeniería de sistemas de la Universidad de Antioquia. frios@udea.edu.co

RESUMEN

Se presentan algunos aspectos de un estudio referente a la enseñanza de la Programación Orientada a Objetos (POO) a jóvenes de instituciones de educación media del municipio de Copacabana-Antioquia. Particularmente, mediante un análisis estadístico, se comparan los resultados de utilizar los lenguajes de programación Léxico y C# en la enseñanza de la POO.

Palabras clave: Algoritmo, enseñanza, POO, aprendizaje.

Recibido: 10 de Abril de 2009. Aceptado: 30 de Junio de 2009

Received: April 10, 2009. Accepted: June 30, 2009

COMPARATIVE EXAMINATION OF LEXICO AND C# LANGUAGES IN LEARNING OF OBJECT ORIENTED PROGRAMMING

ABSTRACT

This paper present some aspects of a study about Object Oriented Programming (OOP) teaching to young from high school (Copacabana, Antioquia). Particularly, by statistical analysis, compare results about using C# and Lexico programming languages in OOP teaching.

Keywords: Algorithm, teaching, OOP, learning.

1. INTRODUCCIÓN

El panorama que presenta en nuestro medio la enseñanza de la media técnica en tecnología y específicamente en informática, es a lo sumo, inicial y suele encontrarse una brecha entre los aspectos teóricos y los prácticos debido posiblemente a que no existen parámetros generales por parte del Ministerio de Educación Nacional para los contenidos, lo que implica que cada maestro enseña lo que sabe.

En este contexto, es importante investigar métodos que faciliten la enseñanza de la programación de computadores, y específicamente, la programación orientada a objetos (POO), con jóvenes de educación media, y determinar en qué grado la experiencia en la construcción de algoritmos es más vivencial y significativa en el proceso de aprendizaje, con el fin de buscar que el estudiante tome participación en el proceso, como ser interactuante con el medio y no como un simple espectador, logrando que solucionen problemas creativamente, desarrollando sus potencialidades y diseñando proyectos que más adelante podrían convertirse en su proyecto de vida como futuros ingenieros de informática, sistemas o afines.

Antecedentes

La programación de computadores se dio inicialmente mediante el paradigma estructural en donde intervienen procesos secuenciales, permitiendo la construcción de algoritmos de una manera sencilla. Este paradigma se ha trabajado desde los años 70 en algunas instituciones educativas preocupadas por la forma de enseñanza y que buscan construir herramientas de software que faciliten al estudiante la adquisición de habilidades y destrezas lógicas. Entre estas herramientas están: Logo [1], Pauscal [2], DFD [3], CS1 [4], Basic [5], entre otros.

Se han realizado algunos estudios que relacionan los Lenguajes de programación con procesos de enseñanza. En [6] se realiza la "Evaluación de Software Educativo, específicamente del tipo orientado al desarrollo del pensamiento. Prueba piloto de LOGO versiones Logowr y Micromundos". Los resultados planteados para dicha investigación dicen que: *"la fundamentación de ambas versiones en Lenguaje Logo, hace muy familiar el uso del lenguaje computacional al del lenguaje humano, el cual está compuesto por reglas, estructuras y posee elementos discretos: palabras y símbolos*

que pueden ser recombinados para producir nuevas expresiones". En [7] se presenta el estudio denominado "Técnicas de solución de problemas simulando una computadora", donde se inicia enseñando algoritmos y termina programando en el lenguaje Logo. Su objetivo es que los niños comprendan las instrucciones del lenguaje y elaboren sus propios algoritmos. En [8] se realiza la "Evaluación de la influencia del Lenguaje Lexico y creación de un modelo de su comportamiento", la cual determinó el desempeño del compilador Lexico y el apoyo que daría esta herramienta especializada para mejorar el nivel de los estudiantes en aquellos cursos que inician el estudio de la computación por la vía de la programación secuencial-imperativa. Sin embargo, estas investigaciones solo hacen referencia a la programación estructurada.

En la década de los noventa, surge el paradigma de programación orientada a objetos, con el cual se intenta simular el mundo real a través del significado de las características y funciones de los objetos. Desde entonces, se ha popularizado el uso de los lenguajes de programación C++ y JAVA [9], al grado de que muchas instituciones de educación decidieron incorporar dichos lenguajes a los cursos de computación básica. El tiempo requerido para explicar los conceptos básicos de objetos y la estructura del lenguaje fueron la causa de un nuevo desfase en el aprendizaje de la computación, pues se requería que incluyeran al mismo tiempo una cantidad de términos como objetos, clases, instancias, polimorfismo entre otros, creando confusión al estudiante. Como lo plantea Kolling [10], *"el estudio de un lenguaje comercial y la comprensión de sus peculiaridades resulta más sencillo y mucho más enriquecedor una vez asumidos los conceptos de orientación a objetos, que si se aborda directamente el lenguaje comercial desde el comienzo"*. Es así como varias instituciones de educación han construido herramientas de software orientadas a objetos como, por ejemplo Greenfoot [11], BlueJ [12], Karen [13], entre otros; lo han hecho para ayudar a que los estudiantes que se inician en este tipo de programación experimenten la lógica del manejo de clases y objetos. Sin embargo, no hay registros de pruebas experimentales de sus verdaderos efectos en el aprendizaje de la programación.

Precisamente en lo concerniente a la enseñanza de la POO, se vienen realizando esfuerzos como el rediseño del compilador Lexico, desarrollado en

1985 y rediseñado en 1992, 2002 y 2009 por Laboratorios Riosur.net [14], Lexico es un lenguaje de programación didáctico, que facilita el aprendizaje y la enseñanza de la programación orientada a objetos.

El objetivo de la presente investigación consiste en dar solución a la siguiente pregunta: ¿Se obtienen los mismos resultados en el desempeño de los jóvenes ubicados en los niveles 10 y 11 cuando la enseñanza y el aprendizaje de la programación orientada a objetos se apoya en los lenguajes Léxico y C#? [15]. Es de aclarar que en este artículo no se pretende explicar como se debe enseñar la POO, se muestran y analizan los resultados de utilizar dos lenguajes en condiciones similares como herramientas para que los estudiantes que aprenden POO puedan llevar a la práctica los conceptos teóricos y puedan implementar programas que permitan completar el proceso enseñanza – aprendizaje.

Para responder a la pregunta planteada, se proponen dos fases. La primera fase consiste en la búsqueda de un software útil para la enseñanza de la POO y que posea algunas características similares con el lenguaje Léxico. La segunda fase consiste en la elaboración del trabajo de campo utilizando los lenguajes seleccionados, para determinar cuál de ellos facilita más al aprendizaje de la POO.

Para el desarrollo de la primera fase, se tuvo en cuenta las características del lenguaje de programación Léxico (Lenguaje, EXperimental, Introductorio, a la Computación con Objetos). Su principal función radica en servir de apoyo didáctico en aquellos cursos que inician a las personas en la disciplina de la programación de computadores, para que puedan concentrarse en la adquisición de destrezas para la formulación de los algoritmos y así puedan experimentarlos en una máquina física. Fue creado por Fabián Ríos y Fabio León Ruiz [14]. Entre las principales características que hacen de Léxico un lenguaje ventajoso para iniciarse en la programación están la facilidad de programar en el idioma nativo de quien lo use, pero sobre todo, la manera en la cual se especifican las instrucciones para generar programas de computadora. Su sintaxis se asemeja bastante a una forma de comunicación más humana y por ello facilita enormemente el desarrollo de destrezas lógicas.

Estos elementos son muy importantes, porque entendiendo la lógica de programación se puede, sin tanta dificultad, acoplarse a cualquier lenguaje de programación, a la vez que se vuelve mucho más simple para el usuario entender lo que hace.

Cabe anotar que aunque para aprender un lenguaje de programación hay que hacer gran uso de la memoria, es sumamente importante lograr que buena parte de lo que se programe se comprenda mejor. Alcanzar esto por medio de lenguajes comerciales es bastante complicado para aquellos que apenas se inician en esta disciplina, los cuales, en muchas ocasiones, comienzan elaborando programas en pseudocódigo, pero sin poder observar en el momento resultados específicos que son fundamentales para despertar el interés y la creatividad.

Lexico requiere de la plataforma Microsoft.NET, la cual se constituye en un conjunto de nuevas tecnologías reunidas con el objetivo de obtener distribuciones de software para servicios que puedan ser suministrados remotamente de manera sencilla y potente, que puedan comunicarse y combinarse unos con otros de manera totalmente independiente del lenguaje de programación y del modelo de componentes con los que hayan sido desarrollados.

Microsoft.NET cuenta con un lenguaje de programación llamado C# pronunciado "C Sharp", el cual es un lenguaje comercial de propósito general diseñado por Microsoft para su plataforma .NET, sus principales creadores son Scott Wiltamuth y Anders Hejlsberg [16]. Éste conserva la sintaxis de anteriores lenguajes, tales como C++, pero enfocado completamente al concepto de clases, y requiriendo de la plataforma .NET de Microsoft para el desarrollo de aplicaciones. Esto hace que algunas características sean similares en los dos lenguajes.

2. DESARROLLO DE LA INVESTIGACIÓN

Los estudiantes de educación media de las Instituciones Educativas del municipio de Copacabana, fueron convocados por el Centro de Recursos Educativos Municipales CREM, ellos fueron seleccionados aleatoriamente por los respectivos docentes del área de informática de cada una de las Instituciones.

Las Instituciones que participaron en dicha investigación fueron: I.E. Villa Nueva, I.E. José Miguel de Restrepo, I.E. Granjas infantiles, I.E. Nuestra Señora de la Luz, I.E. la Trinidad, I.E. Presbítero Bernardo Montoya, Instituto La Asunción.

Los cursos de los lenguajes Léxico y C# tuvieron una duración de cincuenta horas, distribuidas en sesiones de cuatro horas semanales durante tres meses.

Las condiciones tenidas para desarrollar el curso fueron las siguientes:

Los estudiantes participaron del curso en tiempo extraclase, es decir, sin algún tipo de obligatoriedad.

Ninguno de los estudiantes participantes tuvo contacto anterior con algún lenguaje de programación.

Las condiciones del software a tener en cuenta fueron:

Requiere para su funcionamiento la instalación de la plataforma .Net. Se eligió la versión 1.1 y una versión de Internet Explorer mayor a 5.5.

Para el curso de C# se utilizó la IDE SharpDevelop. En el curso de Léxico se utilizó el editor versión marzo 14/2005

2.1 Método de Recolección de Información

Para llevar a cabo el registro de la información se optó por seleccionar la metodología de solución de problemas según Polya [17], la cual se desarrolla usando cuatro etapas esenciales así: comprender el problema, trazar un plan para resolverlo, poner en práctica el plan usando cualquiera de los dos lenguajes prueba y por último comprobar los resultados.

Para seguir dichos pasos la recolección de información se realizó mediante reportes escritos de problemas propuestos en cada sección, a los cuales se les asignó una calificación, que al final se promedió; el valor de esta calificación se asoció con la facilidad con la que el estudiante aprende POO.

Las hipótesis planteadas, validadas con un nivel de significancia del 95%, fueron las siguientes:

H₀: No existen diferencias significativas en la calificación promedio de los estudiantes de educación media del municipio de Copacabana, cuando en el proceso de enseñanza de la POO utilizan una sola de las herramientas de software: Léxico o C#.

H₁: Existen diferencias significativas a favor de la herramienta Léxico en la calificación promedio de los estudiantes de educación media del municipio de Copacabana, cuando en el proceso de enseñanza de la POO utilizan una sola de las herramientas de software: Léxico o C#.

Teniendo en cuenta el concepto emitido dada la elaboración de prueba piloto antes de esta investigación, a priori, se puede suponer que son mejores los resultados en la enseñanza de POO con el lenguaje de programación Léxico que los resultados obtenidos con C#.

RESULTADOS

A continuación, se presentan los resultados de la inferencia de dos medias poblacionales con varianzas desconocidas y no homogéneas, en muestras pequeñas de tamaños distintos. Los datos recolectados en las dos poblaciones, corresponden al promedio de la calificación de los reportes escritos entregados en cada sesión por los estudiantes.

Tabla 1. Resumen estadístico

	Léxico	C#
Tamaño	13	22
Media	3.22308	3.90455
Mediana	3.2	4.0
Varianza	0.686923	0.257541
Desviación estándar	0.828808	0,507541
Mínimo	2.0	3.0
Máximo	4.5	4.6
Asimetría	-0.0055596	-1.48738
Curtosis	-0.720205	-0.262721

La tabla 1 muestra el resumen estadístico para los datos de las dos muestras. Se puede observar que las muestras son de tamaño diferente y los promedios de la calificación oscilan entre 2.0 y 4.6, con medias de 3.22308 para los estudiantes del curso de C# y 3.90455 para los estudiantes del

curso de Léxico. Las desviaciones estándar muestran una significativa dispersión en el curso de C#.

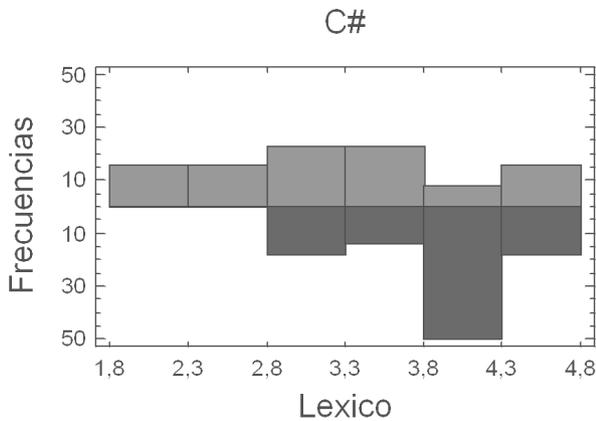


Fig.1. Histograma de los promedios en calificaciones definitivas. El eje horizontal denota el promedio de la calificación, el eje vertical denota la frecuencia

En la figura 1 se puede ver que en el curso Léxico la mayor concentración de promedios de las calificaciones definitivas estuvo entre 3.8 y 4.3. Mientras que en el curso C# dicha concentración estuvo entre 2.8 y 3.8, reflejando esto un mejor rendimiento en el curso de Léxico. También se puede observar una mayor dispersión de las calificaciones promedio en el caso del curso C#, como lo ilustra con mayor claridad la figura 2.

Gráfico Cajas y Bigotes

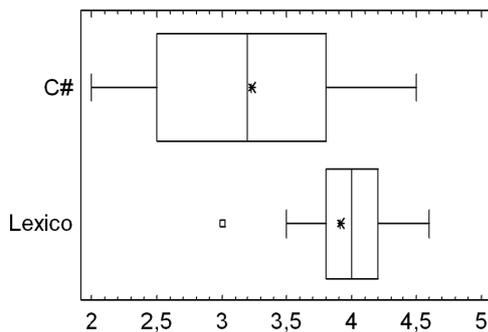


Fig.2. Medida de tendencia central de los promedios en calificaciones definitivas

Las medidas de tendencia central para el conjunto de calificaciones promedio en el curso de C# coinciden en cuanto a su media y mediana, siendo estas pequeñas comparadas con las del curso Lexico.

El coeficiente de asimetría mostrado en la tabla 1, para ambas poblaciones es negativo; en la figura 3 se puede anotar que la curva para el curso Lexico muestra una asimetría a la derecha, mostrando una mayor agrupación para la curva de frecuencias, la cual se interpreta como un mejor rendimiento en dicho curso. Los coeficientes de curtosis de ambas curvas de frecuencia, mostrados en la tabla 1, al ser menores que 0.26, revelan que los datos están muy cercanos al valor de la media.

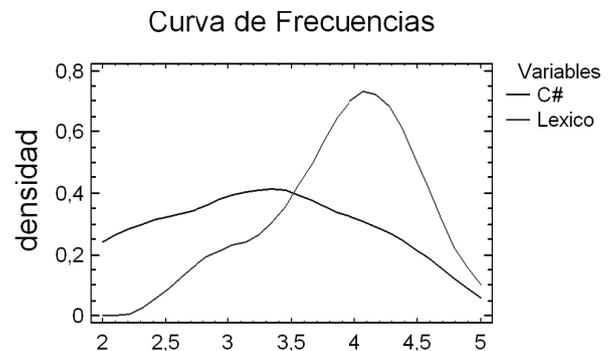


Fig.3. Asimetría de los promedios en calificaciones definitivas

Prueba de Kolmogorov-Smirnov

Esta prueba es realizada computando la distancia máxima entre las distribuciones acumulativas de las dos muestras. En este caso, la distancia máxima es 0.541958 y el valor-P aproximado para la prueba es menos de 0.05, es decir, hay diferencia significativa entre los resultados de la medición realizada a los dos grupos con un nivel de la confianza del 95%.

Estadística total estimada DN = 0.541958
Valor P Aproximado = 0.0164562

Hipótesis a contrastar

Las hipótesis escritas en el numeral 2.1 pueden ser escritas en términos estadísticos como:

$$\begin{aligned}
 H_0 : \mu_1 &= \mu_2 \\
 H_1 : \mu_1 &< \mu_2
 \end{aligned}
 \tag{1}$$

Cuando se hacen inferencias acerca de medias poblacionales cuyas varianzas también se desconocen; se debe hacer primero una prueba de normalidad y de homogeneidad de varianzas para

determinar cuál es el estadístico de prueba apropiado en la prueba de hipótesis.

Prueba de Normalidad

En las figuras 4 y 5 se observa la normalidad de ambos conjuntos de datos. En ellas se puede ver el comportamiento lineal de los datos que garantiza la procedencia de una distribución normal.

Comparación de Desviaciones Estándar

Una prueba-F se puede también utilizar para probar la hipótesis específica sobre las desviaciones estándar de las poblaciones. En este caso, la prueba se ha construido para determinar si el cociente de las desviaciones estándar es igual a 1.0 contra la hipótesis alternativa de que el cociente no sea igual 1.0. Puesto que el valor-P computado es menor de 0.05, podemos rechazar la hipótesis nula en el favor de la alternativa, es decir, no hay homogeneidad de varianzas.

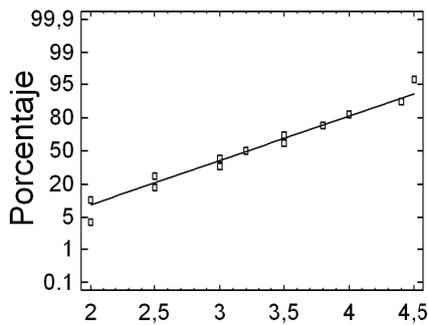


Fig.4. Grafico de normalidad de los promedios en calificaciones definitivas en C#

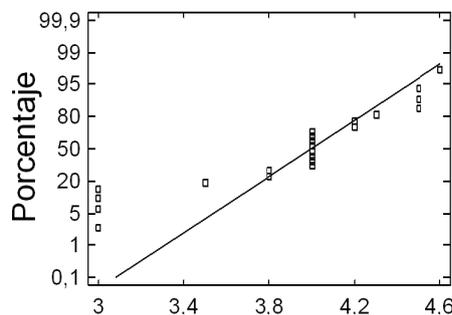


Fig.5. Grafico de normalidad de los promedios en calificaciones definitivas en Léxico

Los intervalos de la confianza mostrados en la tabla 2. para el cociente de las varianzas, donde σ_1 es la desviación estándar para el conjunto de datos del curso C# y σ_2 es la desviación estándar para el conjunto de datos del curso Léxico, se amplían

desde 1.01134 hasta 8.15319. Puesto que el intervalo no contiene el valor 1, hay una diferencia estadística significativa entre las desviaciones estándar de las dos muestras en un nivel de la confianza del 95%.

Tabla 2. Intervalos de confianza para el cociente de las varianzas

	C#:	Léxico
Desviación Estándar	[0.594327, 1.36814]	[0.390477, 0.725308]
Razón de la varianza:	[1.01134, 8.15319]	

Test- F para la comparación de las Desviaciones Estándar

$$\begin{aligned}
 H_0 &: \sigma_1 = \sigma_2 \\
 H_1 &: \sigma_1 \neq \sigma_2 \quad (2) \\
 F &= 2.66665 \\
 \text{Valor P} &= 0.047441
 \end{aligned}$$

Estadístico de Prueba [18]

Cuando dos muestras aleatorias simples, independientes y de tamaño distinto han sido extraídas de poblaciones que siguen una distribución normal, con varianzas desconocidas y diferentes, la estadística de prueba para probar $H_0 : \mu_1 = \mu_2$ en muestras pequeñas es:

$$t' = \frac{(\bar{x}_1 - \bar{x}_2) - (\mu_1 - \mu_2)_0}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} \quad (3)$$

El valor crítico de para un nivel α de significación y una prueba unilateral es aproximadamente:

$$t' - \alpha/2 = \frac{w_1 t_1 + w_2 t_2}{w_1 + w_2} \quad (4)$$

Donde:

- $w_1 = s_1^2/n_1$
- $w_2 = s_2^2/n_2$
- $t_1 = t_{1-\alpha}$; para $n_1 - 1$ grados de libertad.
- $t_2 = t_{1-\alpha}$; para $n_2 - 1$ grados de libertad.

Comparación de Medias

El límite de la confianza para la diferencia entre las medias, donde μ_1 es la media para el promedio de las calificaciones del curso C# y μ_2 es la media para el promedio de las calificaciones del curso Léxico, se extiende hacia arriba a un valor de -0.290579.

Tabla 3. Intervalos de confianza para la prueba de medias

	C#:	Léxico
95% Intervalo de Confianza para la Media	[3,63277]	[4,09074]
95% Intervalo de Confianza para la Diferencia de Medias de asumiendo la no igualdad de varianzas [-0,290579]		

Esto indica el valor más grande para la diferencia que es apoyada por los datos. La prueba-t se puede también utilizar para probar una hipótesis específica sobre la diferencia entre las medias de las poblaciones de quienes las dos muestras provienen. En este caso, la prueba se ha construido para determinarse si la diferencia entre ambas medias es igual a 0 contra la hipótesis alternativa que la diferencia de ambas medias es menor que 0. Puesto que el valor-P computado es menor que 0.05, se puede rechazar la hipótesis nula en el favor del alternativa. Este resultado no asume que las varianzas de ambas muestras sean iguales.

Test t para la comparación de medias

$$\begin{aligned}
 H_0 : \mu_1 &= \mu_2 \\
 H_1 : \mu_1 &< \mu_2
 \end{aligned}
 \tag{5}$$

No asumiendo igualdad de varianzas $t = -2,68226$

$$\text{Valor-P}^* = 0.00775769
 \tag{6}$$

Como se concluye estadísticamente, se puede afirmar que el software Léxico, en comparación con C#, facilita un mayor aprendizaje de la programación orientada a objetos en los estudiantes de educación media, que comienzan con esta disciplina.

* Es el nivel (de significancia) más bajo en el que el valor observado de la estadística de prueba es significativo.

3. CONCLUSIÓN

El estudio y la evaluación realizada durante esta investigación han sido, sin duda, muy positivos y enriquecedores para la enseñanza y el aprendizaje de la informática en las Instituciones Educativas del Municipio de Copacabana, permitiendo que los estudiantes adquieran un conjunto de conocimientos teóricos y prácticos de la Programación Orientada a Objetos. Los estudiantes involucrados mostraron interés y motivación por aprender programación.

Es necesario abrirle espacios a estas herramientas para ampliar las potencialidades de los estudiantes dentro del aula de clase, implementando diversas actividades que permitan solucionar problemas con la ayuda de la programación.

Se recomienda a las Instituciones educativas que tienen media técnica en informática, el uso de Léxico como lenguaje de programación a abordar en los cursos de iniciación a la programación, Léxico es un lenguaje escrito que permite al usuario manipular conceptos propios de la programación utilizando términos familiares, pasando a un segundo plano el manejo de la sintaxis, pues es muy similar al lenguaje natural.

Iniciar la programación con un lenguaje como C# lógicamente es posible, pero puede implicar demasiado tiempo por la necesidad de la adquisición de la sintaxis propia del lenguaje. Lo cual puede causar desinterés de los estudiantes debido a que no terminarían comprendiendo bien lo que hacen o por qué lo hacen, generalmente lo resuelven memorísticamente.

4. AGRADECIMIENTOS

Al Centro de Recursos Educativos Municipales de Copacabana –Colombia, por el apoyo logístico para llevar a cabo esta investigación.

5. REFERENCIAS BIBLIOGRÁFICAS

[1] ¿Trabajas con LOGO? Base de datos. Disponible en: <http://roble.pntic.mec.es/~apantoja/trabajos/bdatos.htm>. [consultado en julio 2005].

- [2] ¿Qué es Pauscal? Disponible en: http://www.geocities.com/pauscal_soft/que_es.html [consultado en junio 2005]
- [3] Grupo Smart. SOFTWARE DFD, Editor e Intérprete de Algoritmos Representados en Diagramas de Flujo.1999. Disponible en: <http://www.expage.com/page/softwaredfd> [consultado julio 2005]
- [4] Ayuda sobre el programa CS1. Departamento de Tecnología Electrónica de la Universidad de Sevilla. Disponible en: http://www.arquired.es/users/aldelgado/proy/cs1/html/cs1_prog.htm [consultado en julio2005]
- [5] El lenguaje Basic cumple 40 años. Disponible en: <http://www.laflecha.net/canales/blackhasts/200405011/> [Consultado en agosto 2004]
- [6] Terció Jenny Sean. "Evaluación de Software Educativo, específicamente del tipo orientado al desarrollo del pensamiento". España 1998.
- [7] Río Gómez, Sara Lúz Del. Técnicas de Solución de Problemas Simulando una Computadora. México El autor: 2001, Disponible en: <http://www.bibliodgsca.unam.mx/tesis/tes9sarg/tes9sarg.htm> [consultado en julio 2005].
- [8] Ríos Fabián y otros. Informe Final del Proyecto "Evaluación de la Influencia del Lenguaje LEXICO y Creación de un modelo de su Comportamiento". Universidad de Antioquia.1993.
- [9] SUN Microsystem. Disponible en: <http://java.sun.com> [Consultada agosto 2005]
- [10] Kölling, M (1999b) Teaching Object Orientation Programming, Part 2: Environments. Journal of Object-Oriented Programming, 11(9) 6-12.
- [11] Henriksen Poul y Michael Kölling. Greenfoot: combining object visualisation with Interaction. University of Southern Denmark. 2003.
- [12] Kölling Michael. The BlueJ and its pedagogy. University of Southern Denmark. 2003. Disponible en: <http://www.bluej.org/papers/2003-12-CSEd-bluej.pdf>
- [13] Pattis Richard."EL MUNDO DE KAREL". Disponible en: <http://www.geocities.com/tintinago/karel.htm> [consultado en agosto 2004]
- [14] Ríos Fabián y otros. Léxico de riosur.net: Un lenguaje POO didáctico en español Disponible en: <http://riosur.mobi/> [consultado abril 2009].
- [15] Morales Munera, Sandra Maria. Evaluación de las herramientas de software Lexico y c# como alternativas en la enseñanza y el aprendizaje de la programación orientada a objetos. Tesis (Ingeniero de Sistemas) Universidad de Antioquia. Facultad de Ingeniería, 2006
- [16] González Seco José Antonio. El lenguaje de programación C#. Disponible en: <http://www.programacion.com/tutorial/csharp/2/> [Consultado en agosto 2004].
- [17] Polya George "Cómo plantear y resolver problemas", Ed. Trillas, México, 1965.
- [18] Daniel, Wayne. Bioestadística: base para el análisis de las ciencias de la salud. Editorial Limusa Wiley. pp. 274. México. 2002.