



REVISTA

educare

*Órgano Divulgativo de la Subdirección de Investigación y Postgrado
del Instituto Pedagógico de Barquisimeto “Luis Beltrán Prieto
Figueroa”*

BARQUISIMETO – EDO. LARA – VENEZUELA

NUEVA ETAPA

FORMATO ELECTRÓNICO

DEPOSITO LEGAL: ppi201002LA3674

ISSN: 2244-7296

**Volumen 17 N° 1
Enero-Abril 2013**

**MODELO DE LA DINÁMICA LÓGICO–ALGORÍTMICA PARA LA RESOLUCIÓN DE
PROBLEMAS DE PROGRAMACIÓN COMPUTACIONAL**

**MODEL OF THE ALGORITHMIC–LOGIC DINAMIC TO SOLVE COMPUTER
PROGRAMMING PROBLEMS**

**Antonio Salgado Castillo
Alexander Gorina Sánchez
Isabel Alonso Berenguer
Universidad de Oriente
Cuba**

**MODELO DE LA DINÁMICA LÓGICO–ALGORÍTMICA PARA LA
RESOLUCIÓN DE PROBLEMAS DE PROGRAMACIÓN COMPUTACIONAL**
*MODEL OF THE ALGORITHMIC–LOGIC DINAMIC TO SOLVE COMPUTER
PROGRAMMING PROBLEMS¹*

TRABAJO DE INVESTIGACIÓN

Antonio Salgado Castillo*
Alexander Gorina Sánchez**
Isabel Alonso Berenguer***
Universidad de Oriente-Cuba

Recibido:10-01-13

Acceptado: 10-04-13

RESUMEN

El presente artículo es un informe de investigación doctoral que propone un modelo de la dinámica lógico–algorítmica para la resolución de problemas de programación computacional. Dicho modelo se distingue por revelar la doble modelación (matemática y computacional) que caracteriza la resolución de un problema de programación. Fue estructurado en cuatro dimensiones: construcción lógico–matemática, orientación matemático–algorítmica, estructuración algorítmico–generalizadora y validación algorítmico–computacional, las que son expresión de los movimientos internos de la citada dinámica y permiten revelar su transformación, encaminada a la formación del pensamiento algorítmico–computacional. El mismo ha servido de base para la elaboración de un sistema de procedimientos didácticos que permite orientar la formación del referido pensamiento. La principal novedad del modelo propuesto radica en haber revelado la lógica integradora entre la representación matemática de la situación problemática y la generalización de la representación pseudocodificada.

Descriptores: lógica algorítmica, programación computacional, situación problemática

ABSTRACT

The present paper is a doctoral dissertation that proposes a model of the logical–algorithmic dynamic for computer programming problems solving. That model is distinguished for revealing the double modeling (mathematical and computational) that characterizes programming problems solving. The study was structured in four dimensions: logical–mathematical construction, algorithmic–mathematical orientation, algorithmic–generalizing structuring and algorithmic–computational validation which are the expression of the internal movements of that dynamics and allows revealing its transformation, addressed to the formation of algorithmic–computational thinking. It has been the basis for the creation of a system of didactic procedures that allow the orientation of the formation of the aforementioned thinking. The main innovation of the proposed model lies on the revelation of the integrating logic between the mathematical representation of the problematic situation and the generalization of pseudo–encoded representations.

Keywords: algorithmic logic, computer programming, problems solving.

¹ Resultado de una investigación doctoral desarrollada por el autor principal y tutorada por los otros dos autores

* Licenciado en Ciencia de la Computación. Profesor del Departamento de Computación. Facultad de Matemática y Computación. Universidad de Oriente (UO), Cuba. Estudiante de doctorado del Programa Curricular en Ciencias Pedagógicas del Centro de Estudios “Manuel F. Gran”. UO. Cuba. Email: asalgado@csd.uo.edu.cu

** Licenciado en Matemática. Doctor en Ciencias Pedagógicas. Profesor del Departamento de Contabilidad y Finanzas. Filial Universitaria Contra maestra. Universidad de Oriente. Cuba. Email: gorina@contre.sum.uo.edu.cu

*** Licenciada en Matemática. Doctora en Ciencias Pedagógicas. Profesora Titular. Departamento de Matemática. Facultad de Matemática y Computación. Universidad de Oriente. Cuba. Email: ialonso@csd.uo.edu.cu

INTRODUCCIÓN

La creciente complejidad de los programas computacionales que se desarrollan actualmente para dar respuesta a las exigencias que se originan en las más variadas esferas de la sociedad, demanda de una alta rigurosidad en la concepción e implementación de los productos que se elaboran. Todo lo cual conduce a la necesidad de iniciar a los alumnos en un camino que los lleve a utilizar técnicas de programación eficientes. De aquí la importancia de poner énfasis en el diseño previo que debe hacerse para desarrollar programas en un lenguaje específico. Una estrategia válida en esta dirección es comenzar a enseñar programación utilizando los algoritmos como recursos esquemáticos, a partir del uso de pseudocódigos como lenguaje principal, para plasmar el modelo de la resolución de un problema (Levy, 1994; Salgado y otros, 2013).

Desde esta perspectiva, se revela la necesidad de potenciar el proceso de enseñanza – aprendizaje de la resolución de problemas de programación computacional en las carreras de ciencias computacionales, tales como: Licenciatura en Ciencia de la Computación, Ingeniería Informática, Ingeniería en Control Automático e Ingeniería en Telecomunicaciones.

El impacto de estas carreras se observa en los notables cambios que va experimentando la sociedad con la vertiginosa irrupción de ellas en numerosas ramas de la actividad social; motivo que de antemano justifica la presente investigación, encaminada a solventar las insuficiencias que existen actualmente en los estudiantes al momento de resolver un problema computacional a través de la programación, dotándolos de un pensamiento algorítmico que les permita, crear, modelar y refinar sus programas de forma eficiente y eficaz, es decir, de forma que alcancen los objetivos o efectos que se desean o esperan a partir de los mismos, con el mínimo de coste computacional posible. De manera que puedan brindar soluciones óptimas a los problemas que deben enfrentar en los diversos contextos de actuación profesional.

Precisamente en la resolución de los citados problemas se aprecian las mayores dificultades que tienen los egresados de las citadas carreras computacionales, los que fueron evidenciados en un diagnóstico realizado en los cursos 2003-2004 al 2011-2012 en la Universidad de Oriente, Santiago de Cuba, Cuba (Salgado y otros, 2013, pp. 59-60), revelándose las siguientes insuficiencias:

- a) Limitaciones en la comprensión de las situaciones problemáticas que se les plantean y en su respectiva modelación desde la programación.
- b) Selección y empleo inadecuado de estructuras computacionales que no permiten la verificación y validación de los algoritmos que se conciben y se implementan.
- c) Imprecisiones en las soluciones computacionales que se dan a las situaciones problemáticas, las cuales no siempre satisfacen las exigencias originales.
- d) Escasas destrezas en la codificación de procedimientos computacionales en una diversidad de lenguajes computacionales.

Estas insuficiencias en la apropiación de los contenidos de programación han sido también confirmadas a nivel internacional por destacados investigadores como Oviedo y Ortiz (2002), Ferreira y Rojo (2005), Guibert, Guittet y Girard (2005) y Faouzia y Mostafa (2007), quienes de una forma u otra reconocen que el proceso de enseñanza–aprendizaje de la resolución de problemas de programación computacional, tiene como centro de sus dificultades la algoritmización. Sin embargo, sus propuestas investigativas se centran en la praxis de dicha programación, sin profundizar en la fundamentación teórica de la lógica del proceso de algoritmización.

Las bases teórico–metodológicas de la citada lógica han sido trabajadas en Salgado y otros (2013), proponiéndose una solución didáctica a la contradicción que se establece entre la modelación matemática y su sistematización algorítmica en el proceso de resolución de problemas de programación computacional, la que sirve de sustento para modelar la citada lógica.

Consecuentemente, el presente trabajo tiene como objetivo realizar una modelación de la dinámica lógico–algorítmica para la resolución de problemas de programación computacional; a partir de la cual se podrán construir instrumentos didácticos que permitan elevar a niveles cualitativamente superiores la actividad formativa del futuro egresado de las mencionadas carreras.

DESARROLLO

La modelación que se propone parte de considerar que la dinámica del proceso de algoritmización computacional demanda una reconstrucción que considere la didáctica de la resolución de los problemas de programación computacional, en su integración con la Matemática y otras ciencias que facilitan una explicación apropiada de dicha dinámica.

Para llevar a cabo esta reconstrucción se concibe la dinámica según Matos y Cruz (2011), es decir, como aquella interactividad del proceso de enseñanza-aprendizaje que connota la relación entre los procesos de comprensión, apropiación, sistematización y generalización del contenido, a través de la aplicación de métodos u otras vías que potencien la interacción entre los sujetos socializadores de la formación.

Así para la citada modelación resultó pertinente utilizar desde el punto de vista epistemológico el sistema de categorías de la Teoría Holístico Configuracional sistematizado por Fuentes, Álvarez y Matos (2003). El haber asumido dicho sistema categorial propició una base adecuada a la naturaleza dinámica del modelo y a su carácter totalizador. Además, facilitó revelar su regularidad, con lo que se profundizó en el conocimiento del comportamiento de la dinámica lógico–algorítmica de la resolución de problemas de programación computacional.

Por otro lado, se tomó en consideración la Teoría del Aprendizaje Significativo de Ausubel (1973, 1976, 2002), para fundamentar la generación de un pensamiento algorítmico en el estudiante, que le permita relacionar la representación matemática de la situación problema con su conocimiento sobre las estructuras lógico–computacionales, en aras de generalizar dicha representación mediante pseudocódigos; fundamentación que se llevó a cabo utilizando el resultado de Ausubel relativo a la forma en que el sujeto logra relacionar los nuevos conocimientos con los anteriormente adquiridos, con lo que incorpora los primeros a su estructura cognitiva.

De igual forma constituye un referente psicológico el Enfoque del Procesamiento de la Información, particularmente sus resultados sobre el conocimiento humano en términos de los procesos por medio de los cuales las entradas sensoriales son transformadas, reducidas, almacenadas, recuperadas y usadas (Gagné, 1991); lo que permitió comprender cómo el estudiante percibe los elementos que componen las situaciones problemáticas y las relaciones que se dan entre estos, los transforma y reduce a representaciones matemáticas, para posteriormente generalizarlos mediante pseudocódigos, almacenándolos y recuperándolos para ser usados cada vez que requiera de su análisis a lo largo del proceso de algoritmización computacional. Así, la representación matemática y la generalización computacional de una situación problemática pueden verse como el procesamiento de la

información que ésta situación brinda, regulada por la base de conocimientos y experiencias del estudiante que las aborda.

Por último, desde la didáctica se consideró la concepción de la generalización dada por Matos y Cruz (2011), quienes aseguran que esta implica abstraer lo que es común a las categorías internas del objeto, pero en su relación con aquellas de su existencia ontológica y epistémica, logrando aportar consideraciones conceptuales que les son comunes y que reflejan el sistema de relaciones que tipifican de manera generalizadora dicho objeto. Consecuentemente, los resultados obtenidos, aun cuando hayan sido producidos en un contexto concreto determinado, tienen posibilidades de ampliarse a otros contextos de naturaleza similar para lograr transformaciones cualitativas.

De lo anterior se infiere que la generalización computacional implica la abstracción de los elementos esenciales de la situación problémica en su relación con los objetos, características y relaciones matemáticas que permiten una representación que la tipifica y eleva a un primer nivel de generalización, para luego representarla mediante pseudocódigos, con lo que lograría un segundo nivel de generalización, más esencial. Esto da cuenta de la doble representación o modelación que, a consideración de los autores, caracteriza la resolución de un problema de programación computacional.

Los referentes teóricos expuestos para la modelación que se propone permiten precisar categorías esenciales de la dinámica del proceso de resolución de problemas de programación computacional, las que surgen del análisis de las diversas fuentes teóricas relacionadas con la temática.

Tal es el caso de la categoría «situación problémica», entendida como una determinada situación en la cual existen nexos, relaciones, cualidades de y entre los objetos que intervienen en ella, que no son accesibles directa o inmediatamente al individuo que la aborda (Polya, 1965; Labarrere, 1994 y Alonso, 2001).

Así, cada vez que el individuo se enfrenta a una situación problémica debe comenzar por tratar de comprenderla, de ahí que la «comprensión», sea una categoría a tener en cuenta en el proceso de resolución de dicha situación; la que a criterio de Wiltrock (1990), “...es una representación estructural o conceptualmente ordenada de las relaciones entre las partes de la información que se debe aprender y entre esa información y esas ideas y nuestra base de conocimientos y experiencias” (citado por Alonso, 2001, p. 4).

Una vez comprendida la situación problémica a solucionar, es preciso interpretarla y representarla matemáticamente; por lo que la «interpretación» deviene en categoría intermedia entre la comprensión y la representación. Consecuentemente, la interpretación es concebida como el proceso mediante el cual se atribuye significado a los objetos que conforman la situación, de modo que adquieran sentido en función de los conocimientos adquiridos o en función del fenómeno o problemática real de que se trate (Hernández, 1989 y Delgado, 2000).

En particular la «representación matemática» es interpretada, en sentido amplio, como todos aquellos objetos matemáticos –signos o gráficos- que hacen presentes los conceptos y procedimientos, con los cuales los estudiantes abordan e interactúan con el conocimiento de esta ciencia, es decir, registran, asimilan y comunican su conocimiento sobre la misma (Alonso, 2001).

Ahora bien, desde el punto de vista computacional, emergen categorías necesarias, tales como «estructura computacional», que es toda aquella entidad computacional (if, for, while, etc.) que permite la ejecución de iteraciones o repeticiones dentro de un programa (Delgado, 2000; Salazar y Delrieux, 2004 y Martínez, 2005).

Otra categoría importante es el «pseudocódigo», definido como estructura computacional, expresada en lenguaje natural, que posibilita formular un algoritmo o programa sin necesidad de usar un lenguaje de programación específico (Ferreira y Rojo, 2005 y González, Estrada y Martínez, 2006).

Así también es significativa la categoría «algoritmización», definida como el planteamiento de una sucesión estricta de operaciones que describan un procedimiento conducente a la solución de una situación problémica. Ésta tiene una doble significación: cognoscitiva y metodológica. Cognoscitiva, porque el algoritmo se constituye en soporte teórico materializado que expresa la secuencia lógica y estricta de la dinámica del modelo. Metodológica, porque la sucesión de operaciones contenidas en el algoritmo, puede servir como base de orientación para realizar la acción (Hernández, 1989; Delgado, 2000 y Alonso, 2001).

Finalmente, la «modelación» permite asociar a un objeto cualquiera, un objeto matemático o computacional que represente determinados comportamientos, relaciones o características del mismo; permitiendo el estudio del mundo objetivo que rodea al hombre a

través de la simulación y representación matemática de los comportamientos y características de los objetos (Rodríguez, 1991 y Delgado, 2000).

Las categorías reveladas, junto con los referentes teóricos anteriormente explicados, se constituyen en elementos esenciales para llevar a cabo una modelación de la dinámica del proceso de algoritmización computacional.

Así, la modelación realizada se estructuró en cuatro dimensiones que son expresión de sus movimientos internos y permiten revelar la transformación de la dinámica lógico–algorítmica de la resolución de problemas de programación computacional. Estas dimensiones son: la construcción lógico–matemática, la orientación matemático–algorítmica, la estructuración algorítmico– generalizadora y la validación algorítmico–computacional. Para explicar estas dimensiones será necesario explicitar su movimiento a partir de las relaciones esenciales que se producen entre sus configuraciones, como procesos internos.

Dimensión de la construcción Lógico–Matemática

La explicación de esta dimensión del proceso lleva a explicitar su movimiento a partir de las relaciones esenciales que se producen entre sus procesos internos, es decir, entre sus configuraciones. Dicho movimiento se revela en la representación matemática de la situación problémica, al ser esta síntesis de la contradicción que se establece entre la comprensión de la situación problémica y la interpretación matemática de dicha situación (ver Figura 1).

Así, la configuración de comprensión de la situación problémica, que inicia el movimiento de la dimensión, es expresión de los sucesivos acercamientos que debe realizar el estudiante para comprender el significado de una determinada situación problémica, a

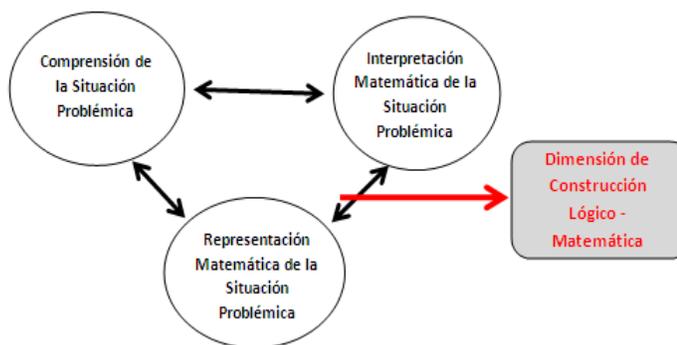


Figura 1: Dimensión de la Construcción Lógico–Matemática.

los efectos de su solución desde una lógica algorítmica. En esta dirección debe tenerse en cuenta que hasta que no se comprenda bien la situación que se aborda, todo el trabajo que se realice en aras de resolverla puede resultar inútil.

La comprensión implica un análisis de la situación problemática por parte del estudiante, es decir, su fragmentación en partes para examinar detalladamente cada una de ellas con el propósito de identificar los objetos reales, matemáticos o computacionales que las conforman, sus características y sus funciones, así como determinar las más importantes o esenciales a los efectos del objetivo que se persigue. Todo esto conducirá a una primera comprensión de la situación problemática, la que deberá irse perfeccionando a lo largo de todo el proceso de algoritmización.

Ahora bien, la comprensión de una situación problemática depende de los conocimientos que posee el estudiante (computacionales, matemáticos y sobre el mundo que le rodea), pues tendrá que ir creando, imaginando objetos y relaciones, en correspondencia con los objetos y relaciones externas que presenta la citada situación.

De aquí que los conocimientos y experiencias sobre el mundo sean sumamente importantes para lograr una adecuada comprensión de la situación, la presencia de conocimientos lingüísticos, por ejemplo, es muy necesaria, ya que el estudiante debe comprender el lenguaje y las experiencias por medio de las cuales recibe el planteamiento de la situación. Pero no basta con el conocimiento del lenguaje en el que se expresa la situación, también es necesaria una comprensión del contexto en que se inscriben los hechos que la conforman, es decir; el conocimiento semántico y el conocimiento de cómo funciona el mundo en el que se da la misma.

Si todos estos conocimientos resultan suficientes para procesar la información que brinda la situación, es muy probable que el estudiante pueda lograr una adecuada comprensión de la misma, lo que facilitará su posterior solución.

Sin embargo, esta comprensión no es suficiente si no se complementa con una interpretación matemática de la solución problemática, como su par dialéctico, expresión del proceso de reconstrucción matemática que realiza el estudiante, a partir de establecer asociaciones y relaciones entre los objetos que intervienen en la situación problemática, con lo que se atribuye un significado a dicho proceso en función de la situación o problemática

real que se pretende resolver, desde un conocimiento matemático, adquiriendo con ello un sentido diferente y cualitativa mente superior.

Aquí resulta de importancia vital el proceso encaminado a establecer asociaciones y relaciones, el que se da una vez que se comprende la situación y conduce a la realización de abstracciones sustentadas en objetos y relaciones matemáticas. Así, para establecer estas relaciones deberán conectarse los resultados de la exploración y vincular las diversas partes de la información obtenida, es decir; conectar los datos similares y localizar las diferencias y semejanzas, para que, una vez hecha la comparación, se puedan establecer los vínculos entre los componentes de la información y expresarlos matemáticamente.

Si se tiene en cuenta que el conocimiento matemático es un conjunto de esquemas mentales en los que el estudiante ha organizado los conceptos, principios, fórmulas y procedimientos de esta materia, se podrá entender que a través de esos esquemas es que el mismo puede relacionar y organizar la información nueva, para interpretarla en forma significativa, en aras de resolver la situación problemática.

Entre las dos configuraciones anteriormente explicadas se establece una relación dialéctica, que se manifiesta entre lo general y lo particular, a partir de concebir que en la medida en que el estudiante va comprendiendo la situación problemática desde una imagen general y totalizadora, se favorece una adecuada interpretación de la misma desde su particularidad matemática, y en este mismo proceso de interpretación matemática, a su vez, se verifica su pertinencia y su adecuación o no a las exigencias de dicha situación, generando una nueva comprensión, más profunda, de la misma. La contradicción se manifiesta porque un cambio en la comprensión de la situación problemática, significa una transformación en su interpretación matemática y una profundización en esta última puede enriquecer la comprensión de la situación, negando la que se tenía inicialmente.

Ahora bien, la relación que se establece entre la comprensión de la situación problemática y su interpretación matemática se sintetiza dialécticamente en la configuración representación matemática de la situación problemática, la que es concebida como la concreción, en un modelo matemático, de la comprensión e interpretación realizadas, de manera que facilite la visualización y conceptualización matemática de los objetos y relaciones inherentes a la situación problemática, analizada desde la lógica esencial de esta ciencia.

Así la representación matemática de la situación problémica se caracteriza por tener como contenido un conocimiento generalizado de dicha situación, de aquí que se pueda decir que tiene como condición una determinada abstracción, que considera la situación en su esencia. Además, esta representación puede pasar por diferentes grados de generalización, esto hace que no se considere como una reproducción mecánica e invariable de la percepción de la situación, sino que se asuma como una configuración dinámica y variable. La relación entre la representación y la situación problémica regulará las modificaciones a las cuales será sometida la primera.

Por otro lado, la representación matemática de la situación problémica permitirá la determinación y activación del conocimiento a emplear en su algoritmización y solución. De aquí que al comienzo del proceso los conocimientos del estudiante le permitirán obtener una representación matemática de la situación y, en consecuencia, serán recuperados de su memoria los procedimientos concretos de algoritmización, si los conoce. En tal sentido, lo que guía la recuperación de los procedimientos apropiados es la representación matemática que de la situación se forma el individuo, por ello se considera crucial dicha representación para tener éxito en la algoritmización, al ser la que determina qué conocimiento se activará en la memoria.

Algo que el profesor no puede perder de vista es que cada individuo crea sus representaciones matemáticas, de ahí que diferentes estudiantes pueden crear diferentes representaciones de una misma situación problémica. Esto más que una dificultad puede favorecer el proceso de enseñanza–aprendizaje de la algoritmización, siempre que sea bien aprovechado desde el punto de vista didáctico.

Todo lo visto hasta aquí consolida, en un primer nivel de esencialidad matemática, la manera en que el estudiante se representa y construye su modelo mental, resultado de la abstracción realizada, como punto de partida para nuevas representaciones y abstracciones, en una búsqueda y resignificación constante de las relaciones lógico–matemáticas modeladas.

Las relaciones entre estas tres configuraciones anteriores permiten connotar la dimensión de la construcción lógico–matemática como expresión del movimiento que se establece entre la comprensión de la situación problémica y su interpretación matemática, que se sintetiza en la representación matemática de la situación problémica, como un

primer estadio de desarrollo en la dinámica de algoritmización computacional que se propone.

Esta dimensión es expresión del carácter de integración y sistematización lógico–matemática, desplegado por el estudiante para desarrollar sus potencialidades en aras de comprender e interpretar pertinentemente la significatividad de los objetos y relaciones modelados, lo que permite alcanzar estadios superiores en la representación matemática realizada durante el proceso resolutor.

Dimensión de la Orientación Matemático–Algorítmica

A su vez la configuración síntesis, representación matemática de la situación problémica, es expresión de otro movimiento del proceso, que a través de la dimensión de la orientación matemático algorítmica expresa la relación que se establece entre la identificación de estructuras lógico–computacionales y la integración jerárquica de dichas estructuras (ver Figura 2).

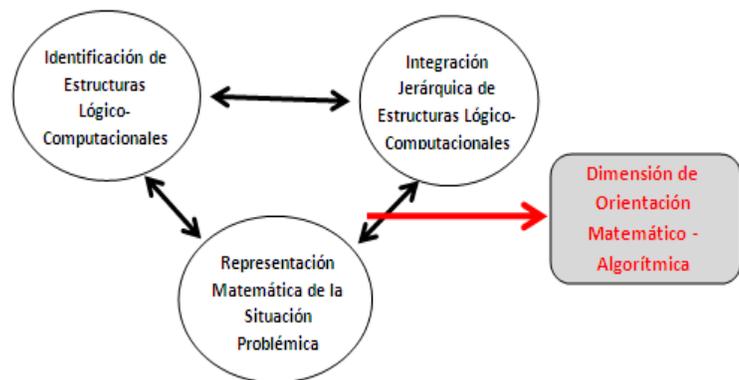


Figura 2: Dimensión de la Orientación Matemático–Algorítmica.

La configuración identificación de estructuras lógico computacionales es interpretada como aquel proceso mediante el cual el estudiante reconoce las estructuras algorítmicas necesarias para garantizar una lógica coherente en el proceso de algoritmización computacional. Este reconocimiento se logra a partir del análisis de la representación matemática de la situación problémica, contentiva de los rasgos esenciales de dicha situación, en correspondencia con los conocimientos computacionales y matemáticos que posee el individuo.

Así, cuando el estudiante ya tiene concebida una representación matemática de la situación problémica, tendrá que distinguir dentro de un conjunto de estructuras algorítmicas conocidas por él, aquellas que le serán útiles para transformar los objetos y relaciones que aparecen en la citada representación, a partir del conocimiento de las características y funciones de dichas estructuras.

Ahora bien, la identificación de estructuras lógico computacionales, presupone el dominio de ciertas habilidades y operaciones. No se puede identificar adecuadamente una estructura si no se ha aprendido a determinar las propiedades y funciones de la misma por medio de la comparación con otras estructuras, esto es lo que permite diferenciarlas. Pero determinar las propiedades y funciones de una estructura es insuficiente, deben saberse diferenciar las propiedades y funciones esenciales de las no esenciales, como habilidad clave para lograr la identificación. Esto requiere del dominio del concepto de propiedad y función, además de habilidades para diferenciar en las estructuras diversas propiedades y características.

Sin embargo, esta identificación de estructuras lógico computacionales por sí sola no garantiza la concepción de un algoritmo que conduzca a la solución de la situación problemática que se aborda, y para lograrlo debe llevarse a cabo en estrecha relación con la integración jerárquica de estructuras lógico-computacionales, como configuración que garantiza la selección, análisis y concatenación adecuada de las estructuras lógicas, previamente identificadas por el individuo, para conformar el algoritmo que constituirá la estructura funcional del proceso de programación computacional.

De esta forma se podrán distinguir y ordenar dichas estructuras, a partir de una comparación previa, constituyéndose en expresión concreta de una elección lógico–computacional–estructurada, que deberá asegurar la eficiencia algorítmica del proceso y la solución de la situación problemática objeto de análisis.

Todo lo anterior implica el desarrollo de una dinámica formativa que facilite al estudiante la identificación e integración jerárquica de las citadas estructuras para dar significado a las secuencias lógicas desde una interpretación algorítmico–computacional, lo que se constituye en vía estratégica para dirigir las operaciones y procedimientos individuales, a partir de un proceso de ordenación, disposición y estructuración algorítmica. Esto propicia una pertinencia de la intencionalidad lógico–computacional que lleva a garantizar la eficiencia algorítmica del proceso.

Es en este momento que se debe garantizar que la integración de las estructuras computacionales concebidas por el estudiante llegue a ser óptima, mediante el empleo de herramientas de control, iterativas o de definición en el orden exacto. Esto permitirá que,

cuando implemente el algoritmo que está diseñando, haga un uso eficiente de la memoria del computador y que el tiempo de ejecución del mismo sea mínimo.

La optimización realizada contribuye al desarrollo de habilidades de programación, en lo referente al diseño y la concepción de algoritmos. Así, una vez culminada esta fase, aunque no se pueda asegurar que el estudiante haya logrado crear un pseudocódigo bien establecido, sí se debe haber producido una apropiación de todos los elementos necesarios para construirlo. De aquí que en esta etapa se connote el carácter integrador y desarrollador que posee la algoritmización como proceso computacional, capaz de propiciar el desarrollo de capacidades cognitivas, al tener que integrar y optimizar simultáneamente.

La interrelación entre estas configuraciones conforma un par dialéctico, que deviene en unidad indisoluble, ya que en la lógica de algoritmización la identificación de estructuras lógico–computacionales se constituye en base de toda integración jerárquica de la misma, al determinar su selección, ordenamiento y jerarquización lógica, dándole un sentido de estructura algorítmica a dicha integración jerárquica; a su vez, esta integración contiene su propia identificación de estructuras. La contradicción se manifiesta porque la identificación de una nueva estructura lógico–computacional conlleva a una jerarquización diferente y la aparición de una jerarquización ineficaz implica una revaloración de la pertinencia de la identificación realizada.

El proceder algorítmico del estudiante se connota entonces en el propio proceso de programación computacional, desde una doble perspectiva: por un lado, permitiéndole planificar y evaluar su acciones y estructuras lógicas en la resolución de la situación problemática; y por otro, retroalimentando constantemente su intencionalidad algorítmica a partir del reconocimiento, selección y concatenación jerárquica de las estructuras algorítmicas, para obtener niveles cada vez más esenciales y precisos de la representación matemática inicial. De aquí que esta representación sirva de guía para dirigir y operacionalizar la dinámica que se construye, convirtiéndose en un proceso que permite el autocontrol del estudiante sobre sus acciones matemático–algorítmicas.

De esta forma la representación matemática de la situación problemática se resignifica desde la relación entre la identificación e integración jerárquica de estructuras lógico–computacionales, ya que esta relación aporta al estudiante nueva información que le

permite valorar dicha representación a partir de una perspectiva algorítmico–computacional, elevándola a un nivel algorítmico cualitativamente superior.

Es así que surge un nuevo movimiento de la dinámica bajo estudio, a partir de las relaciones que se establecen entre las tres configuraciones citadas, lo que deviene en un segundo nivel de esencialidad matemática que está dado por la dimensión de orientación matemático algorítmica. Esta dimensión es interpretada como expresión del proceso que direcciona la construcción de una estructura algorítmico–funcional para la resolución de la situación problémica, desde su representación matemática. Su esencia está en la precisión de las vías y procedimientos lógicos que posee el estudiante, inmerso en un sistemático análisis de los objetos y relaciones matemáticas, así como de las estructuras algorítmico–computacionales pertinentes.

Dimensión de Estructuración Algorítmica–Generalizadora

Como resultado de la relación dialéctica que se establece entre las configuraciones identificación e integración jerárquica de estructuras lógico–computacionales se configura, con un carácter de integralidad cualitativamente superior, la configuración de generalización de la representación pseudocodificada, como constructo teórico que es síntesis y dinamizador de la lógica de algoritmización computacional desplegada por el estudiante. Esta es expresión del proceso de resignificación y reestructuración de la representación matemática creada a partir de una identificación e integración sistemática de estructuras computacionales, lo que conduce a una representación más general, expresada mediante pseudocódigos. Este proceso de generalización trasciende el simple aprendizaje de las características de un lenguaje específico, encaminándose hacia el reconocimiento y aplicación del pseudocódigo, como herramienta base para la escritura del algoritmo.

En síntesis, al sistematizar e integrar, lógicamente y coherentemente, la relación dialéctica que se establece entre las configuraciones identificación de estructuras lógico–computacionales e integración jerárquica de dichas estructuras, se potencia el surgimiento de la generalización de la representación pseudocodificada, como cualidad que da cuenta del carácter generalizador que debe tener todo algoritmo, dando lugar así a la Dimensión de Estructuración Algorítmica–Generalizadora.

Esta estructuración se constituye en una primera aproximación algorítmica de la vía de solución del problema computacional, lo que permite una explicitación de la representación matemática realizada a través de la generalización como habilidad fundamental, a partir del uso de pseudocódigos a modo de herramienta principal para realizar la traducción del lenguaje matemático al computacional (ver Figura 3).

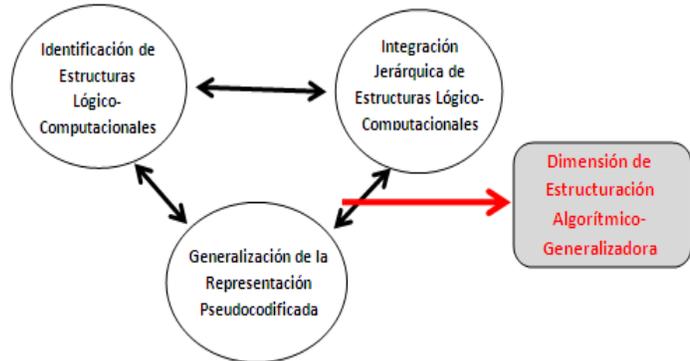


Figura 3: Dimensión de la Estructuración Algorítmico- Generalizadora.

Dimensión de la Validación Algorítmico-Computacional

La configuración síntesis, generalización de la representación pseudocodificada, es expresión, a su vez, de un nuevo movimiento interno en la dinámica de algoritmización computacional, el que se expresa a través de la relación que se establece entre el control sintáctico computacional y la confirmación semántica de la representación computacional.

El control sintáctico computacional es interpretado como los sucesivos refinamientos que debe realizar el estudiante antes, durante y después de estructurado el algoritmo, a partir de tomar en cuenta la sintaxis o escritura del pseudocódigo, a los efectos de regular y evaluar dicho proceso, lo que favorece la toma de decisiones desde un perfeccionamiento constante del mismo.

Una premisa esencial a considerar es que el empleo de pseudocódigos no significa la ausencia de errores, es decir, que pueden crearse algoritmos en pseudocódigo que no sean correctos debido a errores sintácticos. Esto connota el valor didáctico que tiene formar en el estudiante habilidades de exactitud y precisión para el logro y control eficiente de algoritmos, máxime cuando se reconoce que dos estudiantes pueden crear algoritmos diferentes para dar respuesta a un mismo problema, obteniendo resultados equivalentes.

Es importante considerar que el lenguaje algorítmico facilita la representación de las construcciones de los lenguajes de programación, con la ventaja de conservar cierta

coincidencia con el lenguaje natural y con algunos convenios sintácticos inherentes a estos lenguajes, tales como asignaciones, ciclos y condicionales.

Por otro lado, se podría utilizar para describir un algoritmo el lenguaje natural pero, debido a los innumerables problemas que introduce, como la imprecisión o la ambigüedad, es más conveniente utilizar otras herramientas algorítmicas, que describan con mayor exactitud la secuencia de acciones y el orden en el que han de ejecutarse.

A pesar de su flexibilidad, el pseudocódigo tiene que atenerse a una serie de normas para que los algoritmos construidos resulten legibles, claros y fácilmente codificables, por lo que debe verificarse que los identificadores usados tengan un significado de acuerdo con su contenido y que el conjunto de sentencias sea completo; es decir, que permita a los estudiantes especificar cualquier tarea a realizar con suficiente detalle. También deben contener un conjunto de palabras reservadas, además de quedar bien reflejado el flujo de control del algoritmo, que es el orden temporal en el cual se deben ejecutar los pasos individuales del algoritmo.

Sin embargo el control sintáctico computacional no resulta categoría suficiente para lograr una generalización pseudocodificada, sino es en relación con la confirmación semántica de la representación computacional, como proceso que permite al estudiante garantizar la pertinencia de la representación computacional realizada a partir de pseudocódigos; lo que logra mediante la precisión de su efectividad, en correspondencia con la intencionalidad deseada para dar respuesta al problema computacional.

Es en este estadio del proceso donde el estudiante debe verificar la validez de la solución computacional que propone, lo que le permite ratificar si la estructuración del pseudocódigo es correcta en términos del contenido. Este análisis semántico está alejado de cualquier herramienta formal, desde el punto de vista computacional, pues se sustenta principalmente en la verificación de que el algoritmo cumple con su función generalizadora a partir de análisis progresivos de corridas no automatizadas y cada vez más abarcadoras.

Es así que, una vez que el estudiante ha escrito el algoritmo utilizando una herramienta adecuada como el pseudocódigo, se hace necesario comprobar que realiza las tareas para las que fue diseñado y que produce los resultados correctos y esperados a partir de la información de entrada. Este proceso se conoce como prueba del algoritmo y consiste básicamente en recorrer todos los caminos posibles del algoritmo, comprobando, en cada

caso, que se obtienen los resultados esperados. Para lo cual será conveniente realizar una ejecución manual del algoritmo, con datos significativos que abarquen todo el posible rango de valores, así como comprobar que en cada caso la salida coincide con la esperada.

La aparición de errores puede conducir a tener que rediseñar determinadas partes del algoritmo que no han funcionado correctamente y tener que aplicar de nuevo el proceso de localización de errores, definiendo nuevos datos de prueba y recorriendo de nuevo el algoritmo con estos.

Por consiguiente, se debe enfatizar la formación del estudiante en cuanto a la necesaria corroboración de la pertinencia de la representación computacional, a partir del fundamento de que, aun cuando se haya realizado correctamente la modelación matemática y computacional del problema, resulta imprescindible optimizar el pseudocódigo propuesto, con el objetivo de ganar en eficiencia y eficacia de los resultados, lo que redundará en la apropiación de una cultura de perfeccionamiento de los algoritmos computacionales.

Ambas configuraciones devienen en unidad dialéctica, ya que el control sintáctico permite una evaluación constante de las estructuras computacionales, garantizando así la confirmación semántica de la representación computacional realizada. De manera que un error desde el punto de vista sintáctico puede conducir a una falsa interpretación semántica, por lo que, en la misma medida en que se comprueba la intencionalidad y pertinencia del algoritmo, se ratifica y transforma su significado desde la coherencia algorítmica alcanzada. A su vez esta transformación y corroboración de la validez del contenido computacional obtenido, conduce inevitablemente a la necesidad de evaluar y modificar continuamente la sintaxis del pseudocódigo, con lo que se transforma la confirmación semántica anterior.

Las relaciones que se establecen entre las categorías generalización de la representación pseudocodificada, como síntesis y dinamizadora del control sintáctico computacional y la confirmación semántica de la representación computacional, permiten revelar un nuevo movimiento en la lógica

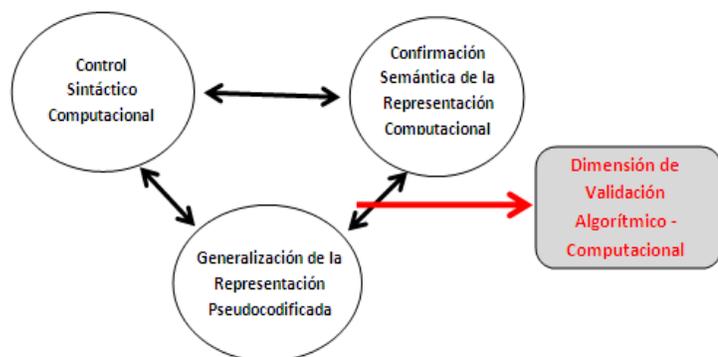


Figura 4: Dimensión de la validación algorítmico-computacional

algorítmica modelada dando lugar a la dimensión de validación algorítmico–computacional (ver Figura 4).

Dicho movimiento se constituye en un segundo nivel de esencialidad algorítmica, expresando la verificación didáctica de la validez y pertinencia de las estructuras computacionales empleadas, a partir de connotar la exactitud en el control, confirmación y generalización de la lógica construida desde la pertinencia y certeza del algoritmo realizado.

Es así que esta dimensión garantiza la corrección, tanto sintáctica como semántica, de la solución propuesta, permitiendo al estudiante diseñar y crear algoritmos computacionales que permitan resolver una determinada problemática de manera eficiente y eficaz, acorde con los requerimientos de la misma, lo que deviene cualidad generalizadora que condiciona la confirmación de la funcionalidad y viabilidad algorítmica de la lógica didáctica desplegada en el proceso de programación computacional.

Ahora bien, la relación que se establece entre las dimensiones de la construcción lógico–matemática, la orientación lógico–algorítmica, la estructuración algorítmico–generalizadora y la validación algorítmico–computacional, se constituye en expresión de un pensamiento algorítmico computacional, estratégico y transformador, que funciona de manera integral y se desarrolla en el proceso de resolución de las situaciones problémicas, en el que los procesos de representación y generalización se constituyen en síntesis sucesivas de interpretación de dichas situaciones, dando lugar a nuevas síntesis reflexivas, cualitativamente superiores (ver Figura 5).

Estos movimientos dan lugar a sucesivas transformaciones en los sujetos implicados en la resolución de las situaciones problémicas, pues durante el desarrollo de la explicada dinámica éstos van alcanzando niveles más esenciales y profundos de síntesis reflexivas, las que les facilitarán la algoritmización a partir de una apropiación de patrones que aporta la Matemática, y que facilitan el análisis y representación de las situaciones problémicas, así como de herramientas computacionales para la estructuración y validación de los algoritmos.

En consecuencia, de la modelación de la dinámica lógico–algorítmica de la resolución de problemas de programación computacional, emerge un sistema de relaciones

esenciales que permite interpretar su comportamiento y transformación. Dicho sistema está integrado por las siguientes relaciones:

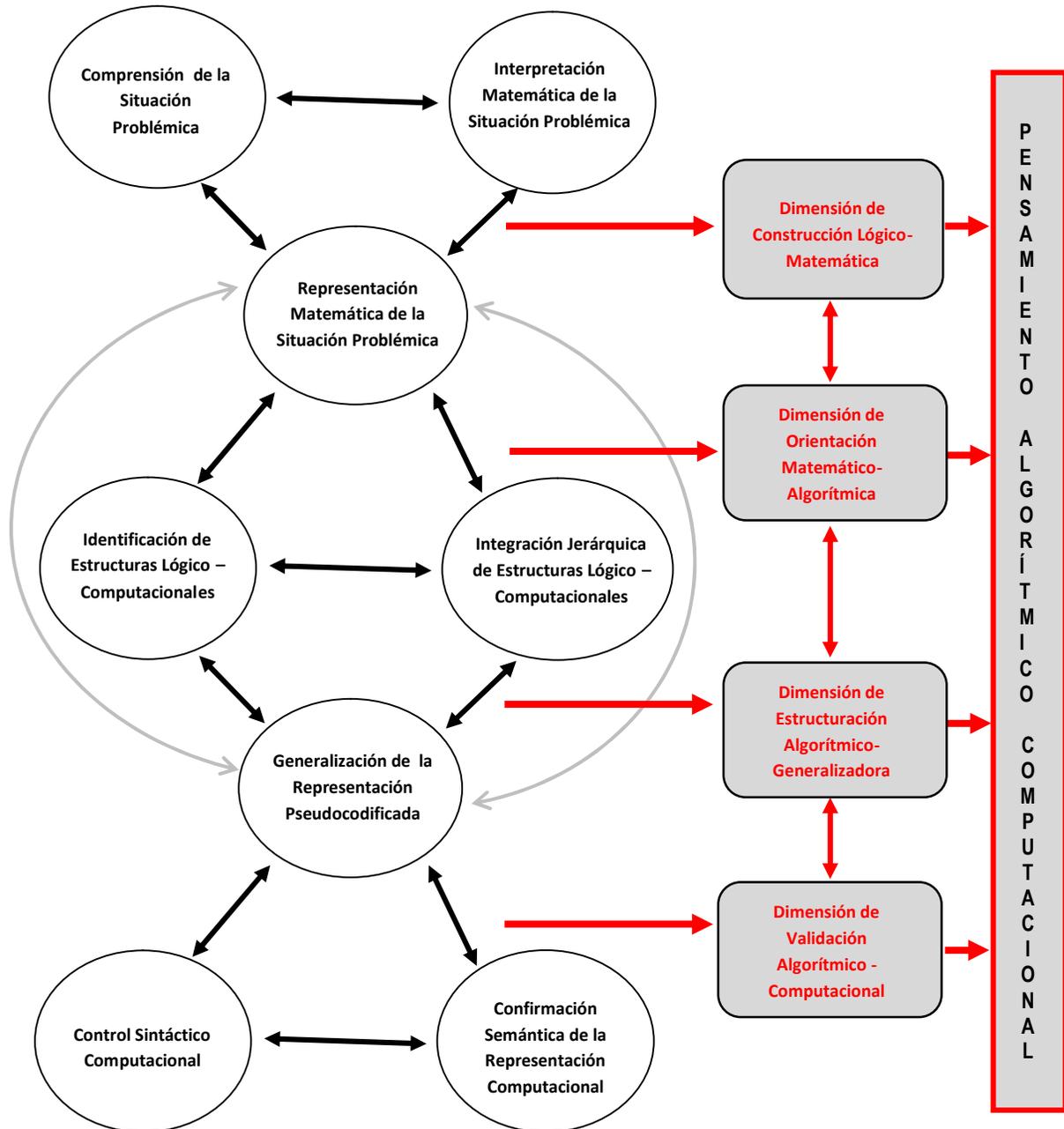


Figura 5: Modelo de la dinámica lógico-algórica de la resolución de problemas de programación computacional.

- a) La representación matemática de la situación problémica, como síntesis de la relación entre la comprensión y la interpretación matemática de dicha situación, que da lugar a un movimiento que genera una construcción lógico -matemática.
- b) La orientación matemático–algorítmica como expresión de una síntesis representativa de la situación problémica, que se origina por la contradicción que emerge entre la identificación y la integración jerárquica de estructuras lógico–computacionales.
- c) La generalización de la representación pseudocodificada, como síntesis de la relación entre la identificación de estructuras lógico-computacionales y la integración jerárquica de éstas, que es generadora del movimiento que conduce a una estructuración algorítmico–generalizadora.
- d) El movimiento que da lugar a la validación algorítmico–computacional, se origina de una generalización de la representación pseudocodificada, que es síntesis de la relación entre el control sintáctico computacional y la confirmación semántica de la representación computacional.

A partir de la profundización en el estudio de estas relaciones se devela entonces como regularidad esencial, la lógica integradora entre la representación matemática de la situación problémica y la generalización de la representación pseudocodificada, que da cuenta de la doble modelación, matemática y computacional, que debe sufrir una situación problémica en el camino algorítmico que conduce a su solución.

Ahora bien, para llevar a la práctica el sistema de relaciones y la regularidad esencial, se requerirá de la elaboración de instrumentos didácticos que orienten a los docentes en la conducción de la formación del pensamiento algorítmico, en aras de elevar la eficacia y eficiencia resolutoria de las situaciones problémicas, con lo que se contribuiría al perfeccionamiento de su formación profesional. En esta dirección, la presente investigación ha avanzado la elaboración de un sistema de procedimientos didácticos, estructurado en correspondencia con las dimensiones del modelo aportado, contenido de objetivos, acciones, criterios evaluativos y patrones de logro que posibilitan llevar a cabo la mencionada orientación. El mismo está actualmente en proceso de validación en la carrera

de Ciencia de la Computación, Universidad de Oriente, Santiago de Cuba, Cuba, para su posterior socialización.

CONCLUSIONES

Al profundizar en el proceso de enseñanza–aprendizaje de la resolución de problemas de programación en las carreras de ciencias computacionales de la Universidad de Oriente, Santiago de Cuba, Cuba, se evidenciaron insuficiencias relativas a la comprensión de las situaciones problemáticas, su modelación desde la programación, la codificación de procedimientos computacionales, así como la verificación y validación de los algoritmos que se conciben e implementan para dar soluciones computacionales a dichas situaciones, todo lo cual dio origen a la necesidad de modelar la dinámica lógico–algorítmica de la resolución de dichos problemas.

El modelo propuesto permitió revelar las configuraciones y relaciones esenciales entre los procesos que lo integran, dando lugar a que emergieran las dimensiones: construcción lógico–matemática, orientación matemático–algorítmica, estructuración algorítmico–generalizadora y validación algorítmico–computacional, que en su relación dialéctica potencian la formación de un pensamiento algorítmico–computacional, imprescindible para la resolución eficaz y eficiente de los problemas de programación computacional.

La doble modelación que caracteriza la resolución de un problema de programación computacional se materializa en el modelo aportado, el que potencia en sus dos primeras dimensiones la formación de habilidades para la modelación matemática de la situación problemática y en las dos siguientes dimensiones da continuidad a dicha formación y enfatiza en la modelación computacional.

La lógica integradora entre la representación matemática de la situación problemática y la generalización de la representación pseudocodificada se constituye en la regularidad esencial de la modelación realizada en esta investigación, la que desde el punto de vista didáctico rige el proceso formativo a desarrollar por el profesor, encaminado al logro de un pensamiento algorítmico–computacional.

El modelo de la dinámica lógico–algorítmica de la resolución de problemas de programación computacional se constituye en un adecuado sustento para la elaboración de

particular de instrumento derivado de dicho modelo es el sistema de procedimientos didácticos que se ha implementado para materializar la formación del pensamiento algorítmico-computacional.

RECOMENDACIONES

Tomando en cuenta la modelación aportada y con base en las conclusiones derivadas de esta, se presentan las siguientes recomendaciones:

- a) Propiciar la capacitación de los profesores de las ciencias computacionales en cuanto a los aspectos relevantes de la dinámica que ha sido modelada, con el propósito de que puedan introducirlos en su práctica docente.
- b) Continuar creando e introduciendo en la docencia de las citadas carreras, instrumentos didácticos derivados del modelo de la dinámica lógico–algorítmica de la resolución de problemas de programación computacional, en aras de reducir las insuficiencias que al respecto se manifiestan en la actualidad.
- c) Perfeccionar el sistema de procedimientos didácticos, que es un caso particular de instrumentos que pueden derivarse del modelo aportado, y que al socializarse puede propiciar su introducción en la dinámica lógico–algorítmica de la resolución de problemas de programación computacional, con lo que se contribuiría al perfeccionamiento de ésta.

REFERENCIAS

- Alonso, I. (2001). *La resolución de problemas matemáticos. Una alternativa didáctica centrada en la representación*. Tesis en Opción al Grado Científico de Doctor en Ciencias Pedagógicas, Universidad de Oriente, Santiago de Cuba, Cuba.
- Ausubel, D. P. (2002). *Adquisición y retención del conocimiento. Una perspectiva cognitiva*. Barcelona: Paidós.
- Ausubel, D. P. (1976). *Psicología educativa. Un punto de vista cognoscitivo*. México: Trillas.
- Ausubel, D. P. (1973). Algunos aspectos psico lógicos de la estructura del conocimiento. En: Elam, S. (Comp.) *La educación y la estructura del conocimiento. Investigaciones sobre el proceso de aprendizaje y la naturaleza de las disciplinas que integran el currículum*, (Págs. 211-239). Buenos Aires: El Ateneo.
- Delgado, J.R. (2000). *Las habilidades generales matemáticas*. Disponible en: <http://www.soarem.org.ar/Documentos/Actas%20de%20la%20VII%20Carem.pdf>

[Consultado el 25 de junio de 2011]

- Faouzia, B y Mostafa, H. (2007). *Utilisation des NTICs pour l'apprentissage et l'autoévaluation de l'algorithmique*. SETIT 2007, 4th International Conference: Sciences of Electronic, Technologies of Information and Telecommunications. TUNISIA, Marzo 25 - 29, 2007.
- Ferreira, A. y Rojo, G. (2005). *Enseñanza de la programación*. Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología. Disponible en: <http://teyvet-revista.info.unlp.edu.ar/numero-1.htm> [Consultado el 25 de junio de 2011].
- Fuentes, H., Álvarez, I. y Matos, E. (2003). *Teoría Holístico Configuracional de los Procesos Sociales*. En Revista Santiago. Universidad de Oriente, Cuba.
- Gagné, E. D. (1991). *La psicología cognitiva del aprendizaje escolar*. Madrid, España: Ed. Visor Distribuciones, S.A.
- González, W., Estrada, V. y Martínez, M. (2006). *Contribución al desarrollo de la creatividad a través de la enseñanza de la programación*. Revista digital Pedagogía Universitaria. Vol 9. No.3. Disponible en: <http://169.158.24.166/texts/pd/1894/04/3/189404308.pdf> [Consultado el 12 de mayo de 2010].
- Guibert, N., Guittet, L. y Girard, P. (2005). *A study of the efficiency of an alternative programming paradigm to teach the basics of programming*. Disponible en: <http://www.lisi.ensma.fr/fr/equip/es/idd/publications.html> [Consultado el 10 de enero de 2012].
- Hernández, H. (1989). *El perfeccionamiento de la enseñanza de la Matemática en la Enseñanza Superior cubana. Experiencia en el Álgebra Lineal*. Tesis Doctoral. Ciudad de la Habana, Cuba.
- Labarrere, A. (1994). *Pensamiento. Análisis y autorregulación en la actividad cognoscitiva de los alumnos*. México: Angeles Editores.
- Levy, L. (1994). *From Specific Problem Instances to Algorithms in the Introductory Course*. SIGCSE BULLETIN ACM.
- Martínez, Y. (2005). *En busca de una nueva forma de enseñar a programar*. Investigación bibliográfica. Disponible en: [http://www.mty.itesm.mx/rectoria/dda/riee/pdf05/27\(DTIE\).YolandaMtz.pdf](http://www.mty.itesm.mx/rectoria/dda/riee/pdf05/27(DTIE).YolandaMtz.pdf) [Consultado el 4 de abril de 2012]
- Matos, E. C. y Cruz, L. (2011). *La práctica investigativa, una experiencia en la formación doctoral en Ciencia Pedagógica*. Santiago de Cuba, Cuba: Ediciones UO.
- Oviedo, M. y Ortiz, F.G. (2002). *La enseñanza de la programación*. Disponible en: <http://bibliotecadigital.conevyt.org.mx/colecciones/documentos/somece2002/Grupo4/Oviedo.pdf> [Consultado el 25 de abril de 2012]
- Polya, G. (1965). *Como plantear y Resolver problemas*. México: Trillas, 1965.
- Rodríguez, T. (1991). *Enfoque sistémico en la dirección de la asimilación de los*

conceptos básicos de la disciplina Matemática Superior. Tesis Doctoral. Ciudad de la Habana, Cuba.

Salazar, C. y Delrieux, C. (2004). *Asignaturas introductorias a la programación: una discusión acerca de sus objetivos y contenidos programáticos*. Disponible en: <http://lsm.dei.uc.pt/ribie/docfiles/txt20031212151824TCI12.pdf> [Consultado el 20 de febrero de 2011].

Salgado, A., Alonso, I., Gorina, A. y Tardo, Y. (2013). *Lógica algorítmica para la resolución de problemas de programación computacional: una propuesta didáctica*. Vol. IV. Año 2013. Número 1, Enero-Marzo. Disponible en: <http://revistas.ojs.es/index.php/didascalía> [Consultado el 11 de marzo de 2013].

Wiltrock, R. (1990). *Comprensión y representación*. MacMillan Publishing Company.