

PROPUESTA DE UNA ESTRATEGIA PARA LA ENSEÑANZA DE TÓPICOS DE COMPUTACIÓN A ESTUDIANTES DE LICENCIATURA EN MATEMÁTICA

Jorge Rey Díaz Silvera, Larisa Zamora Matamoros
Universidad de Oriente. (Cuba)

jdiaz@csd.uo.edu.cu, larisa@csd.uo.edu.cu

Campo de investigación: educación continua. Nivel educativo: superior

Palabras clave: programación, TIC, matemática, enseñanza

Resumen

El presente trabajo tiene como propósito hacer una propuesta de estructuración de contenidos a enseñar sobre tecnologías de la informática y las comunicaciones (TIC) al futuro profesional matemático de perfil amplio, más versátil y adaptable a las condiciones reales donde realizará su actividad una vez graduado y apto para resolver una serie de problemas comunes a distintas esferas de actuación, ya sea en el campo de las investigaciones aplicadas o puras, o en el campo de la enseñanza de la matemática, de tal forma que tenga una sólida base para emplear estas técnicas en su futuro desempeño profesional.

Introducción

Los perfiles actuales de trabajo del profesional matemático exigen cada vez más que éste deba realizar labores muy vinculadas a la computación y/o emplear las nuevas tecnologías de la información y las comunicaciones en su trabajo diario, por otro lado, han ido apareciendo paquetes profesionales que cada vez hacen más dinámico y efectivo el proceso de enseñanza-aprendizaje de los estudiantes, todo lo cual exige continuos cambios en el currículo del Matemático.

La actividad profesional del matemático se caracteriza, entre otros aspectos, por la aplicación de los métodos y modelos matemáticos ya conocidos a la resolución de problemas reales surgidos en las diferentes esferas de actuación, la elaboración de nuevos métodos, cuando los ya conocidos no sean aplicables, la modelación matemática de situaciones diversas que forman parte del objeto de otras profesiones, la utilización de los algoritmos de cálculo que posibiliten la aplicación de las programotecas existentes, la asesoría a otros profesionales sobre estas materias y la enseñanza en el nivel superior de educación.

El uso de herramientas computacionales favorece el perfeccionamiento de categorías importantes dentro del razonamiento matemático como el análisis, la síntesis, la abstracción, la inducción y la deducción, entre otras, evidenciando que las mismas no son sólo un instrumento para resolver problemas, sino también un modelo de razonamiento. De hecho, la relación entre matemática y computación es natural y está dada en un doble sentido: la matemática ha permitido desarrollar y formalizar las teorías informáticas y de otra parte, el empleo de la computación favorece la formación matemática.

El matemático hace uso de las técnicas computacionales básicamente como herramienta de trabajo y/o para construir programas de mediana o poca complejidad, usando, además de un lenguaje de programación de alto nivel, las capacidades de programación que brindan los paquetes matemáticos, como en el Mathematica, el Statistica, etc. Por ello el entrenamiento de los estudiantes en las técnicas computacionales pasa por tres momentos importantes en la carrera:

- Aprendizaje de un lenguaje de programación de alto nivel, preferentemente dentro del paradigma de orientación a objetos.
- Aprendizaje de otras TIC, que complementen los conocimientos básicos que debe tener en tales técnicas.

- Familiarización con el empleo de paquetes profesionales orientados a la solución de problemas matemáticos y otros de propósitos generales, como los de ofimática.

Necesidad del aprendizaje por parte de los estudiantes de la carrera de Matemática de un lenguaje de programación de alto nivel

Existen ciertos aspectos que refuerzan esta necesidad, entre los que podemos citar:

- Las habilidades que aprenden en la programación con un lenguaje de alto nivel le facilitan la comprensión de cómo programar en paquetes matemáticos que brindan tal posibilidad y poder emplear estos de forma atractiva en la solución de problemas matemáticos que actualmente muchos profesores orientan resolver con técnicas no interactivas, aumentando así la capacidad de representación de los estudiantes, habilidad de suma importancia en esta profesión.
- La posibilidad de programar rutinas matemáticas para obtener mejores prestaciones, por ejemplo, en cuanto a la precisión de los cálculos, que los que pueden brindar las rutinas incorporadas en paquetes profesionales, cuando esto resulte necesario.

Más que decidir cuál lenguaje debe enseñarse para programar, hay dos cuestiones que debemos tener en cuenta al elaborar una estrategia para la enseñanza de esta materia:

- Determinar cuáles son las invariantes de conocimientos que serán tratadas, que faciliten la posterior profundización en otras características de lenguajes no enseñadas en el curso, la comprensión de algoritmos que pueden ser expresados en pseudocódigos de lenguajes de programación existentes y las formas de programar en paquetes profesionales que brinden esa capacidad.
- Establecer un orden de impartición de los contenidos de la asignatura, que facilite al estudiante apropiarse de los conceptos y constructores del subconjunto del lenguaje que se determine, a tenor de lo planteado en el punto anterior.

Diferentes lenguajes se han usado en un primer curso de programación, entre los que podemos citar Pascal, Ada, Modula, C, C++, C# y Java. De una parte los criterios de selección no se desligan de las necesidades de programación del sector industrial, científico y empresarial; por otra parte son bien conocidos los requerimientos que debe reunir un buen lenguaje orientado a objetos, fundamentalmente si se usa en un primer curso de programación, que se expresan, entre otras fuentes, en (Kolling, 1999). A partir de nuestras propias experiencias (Diaz, 2006), brindamos algunas orientaciones metodológicas que pueden ser tenidas en cuenta al impartirlo, que se detallan a continuación.

1. *No abandonar a lo largo del curso un estilo de enseñanza de la programación basada en la resolución de problemas.* El estudiante debe quedar convencido, desde que comienza la asignatura, de que su eficacia como programador no está precisamente en dominar la herramienta de programación, cosa evidentemente necesaria, sino en saber resolver problemas con la computadora para lo cual se requiere dominar una *metodología de la programación* cuyo eje central es el algoritmo. Consecuentemente, debe dedicarse las primeras semanas de clase al desarrollo de algoritmos, usando para su expresión alguna forma de pseudocódigo.

2. *Enseñar la estructura del programa en el lenguaje de programación seleccionado.* Introducir las declaraciones de constantes, definiciones de variables, características de algunos tipos primitivos y estudiar, con mayor grado de formalización sintáctica y semántica, los operadores y las estructuras de control básicas del lenguaje, así como nociones de alcance y tiempo de vida. Se sugiere usar solamente las operaciones imprescindibles de entrada/salida. Deben dedicarse varias actividades prácticas en computadoras, para que los estudiantes se familiaricen con el ambiente de

trabajo del lenguaje y puedan correr algunos de los problemas, cuya expresión algorítmica dio en clases anteriores.

3. *Introducir los conceptos de procedimiento y/o función desde un punto de vista algorítmico*, definiendo los parámetros de entrada, salida, entrada/salida y formas que eventualmente usará la función para retornar valores que calcula. Debe evitarse la declaración global de constantes y variables.

4. *Introducir el concepto de clase*. El punto de partida es la definición de tipos de datos abstractos que nos ayuden en la representación de un problema, por ejemplo, la definición de alguna clase para datos numéricos (números fraccionarios, números complejos) donde se implementen algunas operaciones apropiadas, luego se presenta el concepto de clase como una herramienta eficaz para codificar en el lenguaje de programación la definición de tal tipo, junto con los conceptos de atributos y métodos y su empleo en el problema.

Deben ser estudiadas las formas de visibilidad que implementa el lenguaje de programación para las componentes de clase, las partes de la clase (interfase e implementación), constructores simples y destructor.

Si usamos el lenguaje C++, debe darse a la variable *this* un tratamiento muy simple, que evite mezclarnos con los punteros y lograr que en este momento sea aceptado por definición que **this* es el objeto actual que envía el mensaje en proceso.

Debe explicarse el concepto de estado de un objeto y a partir de aquí las funciones de acceso y modificación de un objeto (funciones Get y Set) y su importancia.

5. *Introducir los tipos enumerativos y fundamentalmente tipos de arreglos* y introducen las formas en que éstos se operan o transforman (lecturas, escrituras, cálculos, búsquedas, etc.) construyendo clases en que sus atributos sean la cantidad real de elementos que tiene el arreglo y la definición de los elementos del arreglo usando el constructor por defecto de un arreglo.

6. *Introducir el concepto básico de herencia*, los conceptos de clases y superclases, las formas en que se heredan atributos y métodos, la visibilidad protegida o equivalente, todo a través de un ejemplo de entrada y las características de los constructores y destructores de las clases derivadas. Se debe demostrar a partir de dicho ejemplo, desarrollado en laboratorio, la importancia de definir métodos virtuales, enfatizando además que esta definición constituye una forma de polimorfismo.

7. Si se emplea el lenguaje C++ será imperioso *enseñar el concepto de punteros*, como vía para definir variables dinámicas usando las funciones *new* y *dispose* y como un ejemplo la definición de arreglos dinámicos usando punteros, disciplinando a los estudiantes en el chequeo del rango de sus índices y dar herramientas para evitar la aparición de referencias nulas o basura. Deben estudiarse las cadenas de caracteres, usando las bibliotecas más evolucionadas que posea el lenguaje.

8. *Retomar el concepto de herencia*, para enseñar conceptos relativos a la definición de tipos-subtipos en una jerarquía y cómo esto permite definir objetos polimórficos.

9. *Introducir el manejo de estructuras de datos de listas, pilas y colas*, usando las facilidades de definición de estructuras dinámicas que brinda el lenguaje, implementadas como tipos de datos usando el concepto de clases.

Los ejercicios que se desarrollan en la asignatura deben facilitar la definición de tipos de datos abstractos para entidades matemáticas, trabajo con fórmulas matemáticas, aproximaciones, confección de gráficas y similares.

Uso de otras herramientas computacionales

Se hace imperioso que los estudiantes de la carrera adquieran conocimientos adicionales sobre tecnologías informáticas, de tal forma que puedan tener una mejor adaptación a entornos de trabajo cooperativos y al procesamiento de datos, lo mismo de forma centralizada que de forma remota. Es por ello que proponemos una segunda asignatura de perfil computacional, que podemos denominar Complementos de Informática, que contemple, al menos, las siguientes temáticas y sistema de conocimientos que pasamos a detallar:

- Sistemas de bases de datos.

Conceptos fundamentales de bases de datos, sistemas de gestión de bases de datos (SGBD) y sus características y sistemas de bases de datos. El modelo relacional, álgebra y cálculo relacional y lenguajes que permitan expresar demandas sobre la base de datos (SQL), presentando los comandos básicos de sus sublenguajes de manipulación y creación de datos. Formas normales ligadas a dependencias funcionales. Cuestiones esenciales del trabajo con un SGBD (si es en ambiente propietario puede ser SQL Server o Access, si es software libre MySQL), elementos de seguridad e integridad que incluyen y los elementos básicos de una plataforma cliente-servidor para bases de datos y la conectividad con bases de datos a través del lenguaje estudiado en la asignatura de programación, usando componentes ADO.

- Conceptos elementales de redes y su administración.

Propósito de los sistemas y redes de comunicación de datos. Redes locales y globales. Medios de transmisión. Topologías de red. Acceso a los medios de transmisión. Estándares de red. Dispositivos de interconexión. Modems. Tipos de servidores. Correo electrónico. Principales software de red. Protocolos y direcciones. TCP/IP. Seguridad en redes. Configuración de redes. Manejo de usuarios y de grupos. Establecimientos de derechos. Políticas de Administración. Configuración y administración Windows NT.

- Confección de páginas web dinámicas y acceso remoto a bases de datos.

Introducción a los servicios de web e Internet, elementos básicos del lenguaje HTML y del diseño de páginas estáticas, JavaScript, introducción a la programación en Web usando PHP o ASP con JScript y confección de páginas dinámicas y administración remota de bases de datos.

Uso de paquetes profesionales

Otras herramientas ampliamente utilizadas en la inserción de la tecnología informática al currículo del matemático son los procesadores simbólicos y los geométricos. El uso de estos procesadores le permitirá, tanto al estudiante -en el proceso de aprendizaje- como al profesor -durante el proceso de enseñanza- acercarse más a una representación de objetos y reglas, que de otra forma quedarían representados de forma abstracta.

Los procesadores simbólicos como Derive, Mathematica, Maple, Mathlab, Mathcad, etc son programas que resuelven, calculan, simplifican, desarrollan series y grafican expresiones del álgebra y el cálculo por medio de símbolos. Ellos permiten proponer situaciones problemáticas para ser resueltas por el estudiante y confirmadas mediante expresiones simbólicas por el sistema.

Resulta de vital importancia que el estudiante se familiarice y logre un dominio de las principales características de algunos paquetes profesionales en el área de la estadística, como el Statistica, el SPSS, etc., de preferencia el primero. Es bueno que el estudiante aprenda también a resolver problemas estadísticos usando el Excel, que aunque de menos alcance que los primeros paquetes

mencionados, brinda posibilidades básicas de cálculo que no son de desdeñar. También se propone el uso de otros editores como el Latex y software para la optimización, como el PLEC++.

Sistematización multidisciplinaria del uso de las técnicas computacionales

Consideramos que la capacidad para algoritmizar la solución de un problema puede ser dinamizada a través de las disciplinas matemáticas que se imparten paralelamente a la enseñanza de la programación, tales como Álgebra, Análisis Matemático y Geometría, y que la propia asignatura de Programación puede dinamizar la capacidad codificadora a través de sus ejercicios, tanto teóricos como prácticos, basándose en problemas provenientes de esas asignaturas. Programar algoritmos para la solución de problemas provenientes de las diferentes disciplinas matemáticas tendrá un influjo muy positivo en la consolidación de los sistemas de conocimientos propios de las mismas. Esto se manifestará también en los años superiores y, por ejemplo, el profesor de Matemática Numérica puede indicar la programación de ciertos métodos, cuya presentación analítica se ha discutido en clases y podrá entonces profundizar en otras cuestiones teóricas relacionadas, por ejemplo, formas de minimizar errores.

La implementación de tipos de datos abstractos para números racionales y complejos, polinomios, matrices, vectores y otros algoritmos desarrollados en la disciplina de Álgebra son ejemplos que deben convertirse en clásicos en la asignatura de Programación. De otra parte las habilidades de programación se verían favorecidas con la utilización en asignaturas propias del perfil de algunos de los paquetes profesionales para la actividad del matemático existentes para el cálculo de límites, derivadas, integrales, etc. El empleo de los paquetes matemáticos profesionales, en general, le dará al profesor en su asignatura la posibilidad de ganar tiempo de cálculo y poder dedicar más tiempo a la discusión de nuevas teorías, de incentivar el espíritu investigativo y el conocimiento productivo en el estudiante, pues le puede orientar trabajos más complejos y de mayor aplicabilidad que el estudiante puede enfrentar con mayor individualidad.

Los conocimientos de bases de datos, redes y tecnologías de Web son empleados en las prácticas laborales investigativas, dentro de la disciplina Práctica Profesional del Matemático (que tiene asignaturas presentes en cada año de la carrera), para dar solución a problemas concretos que emplean el procesamiento de grandes volúmenes de información, incluido el procesamiento remoto de los datos e incluso construcción de hipermedias y sitios web que pueden apoyar, por ejemplo, la gestión del profesor y el aprendizaje autónomo e interactivo del estudiante. Es precisamente esta disciplina el momento idóneo para el empleo de todas las tecnologías informáticas aprendidas con la solución de los problemas propios de la profesión.

Resultados de las encuestas

Con el objetivo de indagar cuáles eran los elementos de programación a tener en cuenta en el currículo del Licenciado en Matemática, así como el uso que se hace de otras tecnologías de la computación y su vínculo con los modos de actuación del profesional matemático, fueron aplicadas dos tipos de encuestas, una dirigida a graduados de la carrera y la otra a estudiantes de tercero a quinto año. Estos resultados, de los cuales a continuación mostramos los principales, fueron tenidos en cuenta al elaborar la estrategia que propone el presente trabajo:

- Se evidencia un mayor empleo de las TIC en las disciplinas de Práctica Profesional del Matemático (87,50%), Matemática Numérica (62,50%) y Estadísticas (37,50%).

- Los graduados, en su etapa estudiantil, usaron, en una clasificación de algunas veces o frecuentemente, la Programación en un 75%, Sistemas de bases de datos en un 62,50% y Estructuras de Datos en un 50%.

Por su parte los estudiantes han opinado en su totalidad que los conocimientos recibidos en la asignatura de Programación son adecuados para su formación como futuros profesionales, que los aplican fundamentalmente en las asignaturas de Matemática Numérica (100% de los estudiantes), Práctica Profesional (40%) y Criptografía (20%) y un 80% de los estudiantes opina que son usadas situaciones relacionadas con temáticas afines a la matemática.

Conclusiones

En el trabajo se ha presentado una estrategia para enseñar un conjunto de conocimientos básicos sobre tecnologías computacionales, haciendo un mayor hincapié en la enseñanza de la programación, además de presentar una propuesta de estructuración de contenidos de esta asignatura y de otra asignatura que complementa los conocimientos de tales tecnologías y las formas en que se pueden introducir los paquetes profesionales matemáticos y de la ofimática.

Es incuestionable el hecho de que el dominio de un conjunto básico de las TIC contribuye a la mejor preparación de los estudiantes, ampliando el espectro de su futura ubicación laboral. De hecho, la mayor parte de los estudiantes que se han graduado en los últimos años han realizado sus labores profesionales vinculados principalmente con aplicaciones de la computación. Para aquellos que se dediquen a la enseñanza de la matemática, el desarrollo de esta cultura informática les permite tempranamente comprender las potencialidades de las TIC en el ámbito educativo y, una vez graduados, hacer un uso eficiente de las potencialidades de las mismas en el perfeccionamiento del proceso de enseñanza-aprendizaje.

Consideramos que con las dos asignaturas, Programación y Complementos de Informática, se pueden lograr los objetivos de dar una formación básica al estudiante de Matemática en técnicas computacionales, a la vez que el estudio de paquetes profesionales matemáticos y estadísticos, como el Mathematica, MatLab, Statística, PLEC++ y otros, pueden ser introducidos a través de las asignaturas básicas o del ejercicio de la profesión, incluyendo la Práctica Laboral e Investigativa. Esta propia disciplina puede propiciar la adquisición de habilidades en el empleo de los paquetes del Office, principalmente Word, PowerPoint y Excel.

Referencias bibliográficas

- Ministerio de Educación Superior (1998). Programa de la disciplina. Programación y algoritmos para la carrera de Matemática. MES, República de Cuba.
- Departamento de Matemática, Universidad de Oriente. (2004). Programa Director de Computación de la carrera de Matemática. Santiago de Cuba, Cuba.
- Díaz, J. (2006). Enseñando programación con C++: una propuesta didáctica. *Revista de Informática Educativa y Medios Audiovisuales*, 3(7), junio 2006, 12-21.
- González, W., Estrada, V. y Martínez, M. (2004). Contribución al desarrollo de la creatividad a través de la enseñanza de la Programación. *Revista Pedagogía Universitaria*, 9(3), 115-148.
- Kölling, M. (1999). The problem of teaching object-oriented programming. Part I: Languages. *Journal of Object-Oriented Programming*, 11(8), 8-15.
- Oteiza, F., Silva, J. y Equipo Comenius (2001). Computadores y comunicaciones en el currículo del matemático. En <http://www.eduteka.org/pdfdir/SilvaMatematicas.pdf> (Citado el 9 de Marzo de 2005).