

# Diseño e implementación de un software educativo en Cálculo Numérico <sup>1</sup>

**Pizarro, Rubén A. - Ascheri, María E.**

Facultad de Ciencias Exactas y Naturales – Universidad Nacional de La Pampa

Uruguay 151 – (6300) Santa Rosa – La Pampa – Argentina

[ruben@exactas.unlpam.edu.ar](mailto:ruben@exactas.unlpam.edu.ar)

[mavacheri@exactas.unlpam.edu.ar](mailto:mavacheri@exactas.unlpam.edu.ar)

## Resumen

En este trabajo presentamos las características de un software educativo que hemos desarrollado con MATLAB. Hemos diseñado este software para implementar los diferentes métodos de resolución de ecuaciones no lineales que se estudian en Cálculo Numérico. Esta materia corresponde al Profesorado en Matemática, Licenciatura en Física e Ingeniería Civil, carreras que se siguen en la Facultad de Ciencias Exactas y Naturales de la Universidad Nacional de La Pampa.

Además, detallamos las motivaciones que nos llevaron a su diseño e implementación, como así también las modificaciones que se introdujeron tanto en las clases prácticas como teóricas. Estas modificaciones se fueron produciendo a partir del uso de este software y del análisis a-posteriori de la información obtenida por medio de encuestas a alumnos, y de la observación de clases.

Finalmente, luego de utilizar este software durante tres años (versiones anteriores) y a partir de los resultados alcanzados, realizamos un análisis sobre las mejoras / correcciones necesarias para elaborar una versión final del mismo.

**Palabras clave:** software educativo, cálculo numérico, resolución de ecuaciones no lineales.

## Introducción

Durante el desarrollo de Cálculo Numérico, se estudian diferentes métodos para la resolución, en forma aproximada, de ecuaciones no

lineales, sistemas de ecuaciones lineales y no lineales, aproximación e interpolación, derivación e integración numérica. El desarrollo de estos temas demanda a los alumnos el aprendizaje de una gran cantidad de contenidos compuestos de métodos y fórmulas. Esto hace que frecuentemente los alumnos realicen los cálculos matemáticos aplicando las fórmulas sin efectuar un análisis detallado del comportamiento de cada método, según la situación problemática abordada y los resultados obtenidos. Este análisis es de gran importancia para facilitar la comprensión y permitir un aprendizaje significativo [1], en beneficio del uso de estas temáticas en aplicaciones futuras.

Considerando la unidad curricular “Resolución de ecuaciones no lineales”, y con los objetivos de facilitar y mejorar la comprensión de los métodos numéricos que en ella se estudian y de contar con una nueva herramienta para su enseñanza, nos propusimos desarrollar un software educativo que implemente cada una de estas técnicas numéricas. En este software abordamos, principalmente, la interpretación gráfica de los métodos numéricos y las condiciones requeridas para su aplicación. Para su diseño hemos tenido en cuenta la opinión de Cuevas [2], según el cual al tratar de enseñar a los alumnos un concepto matemático se debe presentar la reunión de varios mundos, contextos o registros de representación semiótica, lo que hace más compleja la actividad de desarrollar un software educativo para Matemática.

Este software permite visualizar cómo los distintos métodos numéricos se van acercando o no a la solución buscada, ofreciendo la posibilidad de modificar fácilmente las

opciones a partir de las cuales se implementan cada uno de ellos.

## Software educativo

Existen diversas definiciones de software educativo y son muchos los estudios realizados sobre los mismos. Una de las definiciones que hemos seleccionado [3], indica que un software educativo debe ser:

- Pensado y desarrollado con finalidad didáctica desde el momento de su elaboración.
- Elaborado de manera tal que se utilice el ordenador como soporte en el que los alumnos realizan las actividades que ellos proponen.
- Interactivo, es decir, debe contestar inmediatamente las acciones de los alumnos y permitir un diálogo y un intercambio de informaciones entre el ordenador y los alumnos.
- Desarrollado de forma que individualice el trabajo de los alumnos o que se adapte al ritmo de trabajo de cada uno, y que puedan adaptarse sus actividades según las actuaciones de los alumnos.
- Fácil de usar. Esto es, los conocimientos informáticos necesarios para utilizar la mayoría de estos programas deben ser similares a los conocimientos de electrónica necesarios para usar un video, es decir, mínimos, aunque cada programa tenga reglas de funcionamiento que sea necesario conocer.

Además de las definiciones existentes de software educativo, existen clasificaciones de los mismos que consideran, entre otros aspectos, los contenidos, los destinatarios, los procesos cognitivos que integra, la función del aprendizaje y la estrategia didáctica [4].

### Elaboración de programas en el curso de Cálculo Numérico: caso particular de una unidad curricular

Durante el desarrollo de la unidad curricular “Resolución de ecuaciones no lineales”, los

alumnos elaboran programas para implementar los diferentes métodos que resuelven ecuaciones no lineales en forma numérica. Estos programas ofrecen a los alumnos información valiosa, evitando la realización de cálculos engorrosos, tediosos y a gran escala. Es la computadora, una vez realizado el programa pertinente, la que se encarga de obtener los resultados a una gran velocidad y con el grado de exactitud deseado. Pero el inconveniente es que sólo obtienen valores numéricos o alguna leyenda alusiva a la convergencia o no del método, relacionando cota de error con cantidad de iteraciones y sin ningún otro tipo de análisis. Es decir que no realizan ningún otro estudio o exploración relativo a las características de las funciones (como, por ejemplo, sus gráficas, qué parámetros iniciales son convenientes tomar, cuáles son las razones por las que un método converge o no). Además, en la mayoría de los casos, los alumnos cuentan con escasos conocimientos de programación, limitándolos para el desarrollo de programas de mayor complejidad.

### Elaboración de un software educativo para la resolución de ecuaciones no lineales

Con la elaboración de este software educativo (al cual clasificamos, fundamentalmente, como una herramienta de ejercitación y práctica [5]), pretendemos sumarle a los resultados numéricos obtenidos, la interpretación gráfica de los métodos aplicados a diversas situaciones problemáticas. Por medio de sucesivos gráficos, el software mostrará cómo las distintas iteraciones se van acercando o no a la aproximación de la solución buscada, y por qué un método es más conveniente que otro según la situación problemática abordada. También aspiramos a que este software permita la interacción con el alumno, lo que influirá positivamente en su aprendizaje.

La ventaja que presenta la elaboración e implementación de un software educativo, radica principalmente en la posibilidad de contar con una herramienta que se adapte exactamente a las necesidades del curso [6].

Entre estas necesidades están las referidas, en nuestro caso, a:

- La comprensión profunda de los contenidos conceptuales a desarrollar.
- El logro de los objetivos generales del curso y de los objetivos específicos relativos a esta unidad curricular.
- La posibilidad de mantener una estrecha relación entre teoría y práctica.
- La velocidad para el cálculo científico y obtención de resultados.
- La visualización de los procesos y resultados.
- La adquisición de habilidad y destreza en el procesamiento de información.

Adicionalmente, la implementación del software permite que cada alumno al interactuar con él, pueda desarrollar las actividades propuestas siguiendo su propio ritmo de aprendizaje.

### Programación del software educativo

Hemos programado este software en MATLAB, ya que este lenguaje posee diversas herramientas matemáticas y posibilita la creación de GUI (interfaz gráfica de usuario). Estas características nos permiten alcanzar los objetivos propuestos sin demasiada dificultad. Además, su programación es sencilla y brinda un entorno agradable al momento de editar los programas correspondientes.

Para la ejecución de este software, se debe copiar el conjunto de archivos que lo componen en el directorio de trabajo definido para MATLAB. Actualmente, se está trabajando en la definición de la portabilidad del software para permitir mayor versatilidad al momento de implementarlo en diferentes equipos.

### Interacción entre el software educativo y el usuario

Al iniciar el software, luego de la primera ventana de presentación, aparecerá la pantalla que se muestra en la figura 1.

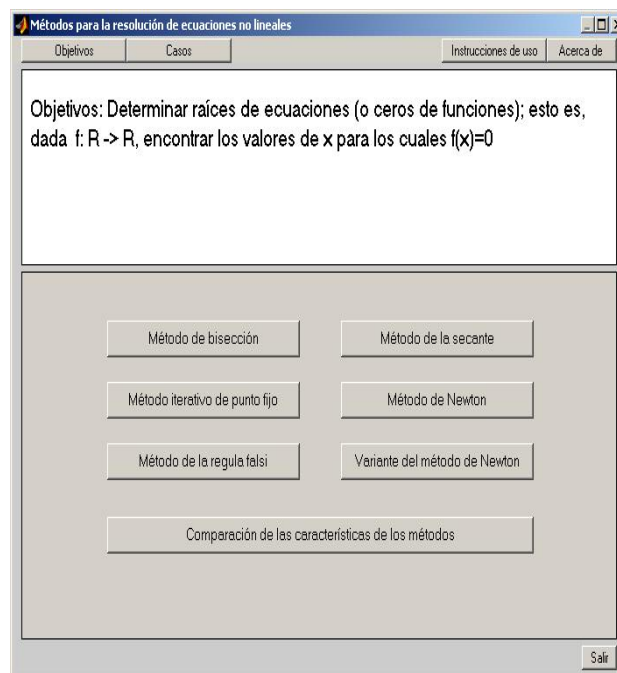


Figura 1

En la misma, el alumno puede explorar las características distribuidas en la barra superior de la ventana, seleccionando los distintos botones (*Objetivos*, *Casos*, *Instrucciones de uso*, *Acerca de*). La característica elegida se describirá en la parte superior de la ventana. En caso contrario, puede elegir el método que desea utilizar para hallar las raíces de las ecuaciones que quiera resolver o podrá ver una comparación entre los diferentes métodos numéricos que se estudian en el curso de Cálculo Numérico.

Optando por alguno de los métodos, el alumno se encuentra con una pantalla, figura 2, en la cual debe ingresar ciertos datos (dependiendo del método), tales como:

- Intervalo en el cual graficará la función que representa la ecuación a resolver.
- Valor/es inicial/es.
- Cantidad de iteraciones máximas.
- Error admisible.
- Pausa entre iteraciones.
- La ecuación a resolver.

Una vez definidos estos valores puede:

- Graficar la función.
- Aproximar la solución.
- Detener / Continuar la ejecución.

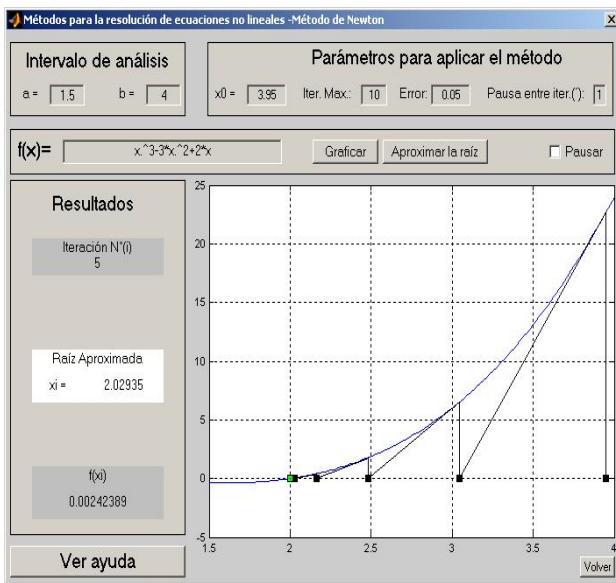


Figura 2. Implementación del método de Newton

Con la opción *Pausar*, se detiene la ejecución permitiendo analizar el desarrollo del programa, considerando principalmente:

- Número de iteraciones realizadas hasta ese momento.
- Intervalo de análisis actual.
- Valor al que se aproxima la solución.
- Valor que adquiere la función en ese punto.

Al desactivar la opción *Pausar*, se reanuda la ejecución continuando con la realización de las restantes iteraciones.

Durante la utilización del software, la interacción entre éste y el alumno se da, fundamentalmente, en las etapas de:

- Selección de los datos iniciales para implementar el método.
- Ejecución del método.
- Obtención de los resultados.

Cuando el software se está ejecutando con los datos seleccionados, interactúa con el usuario mostrando los resultados obtenidos o informando sobre las diferentes alternativas que surgen a partir de la aplicación del método.

Una de las acciones fundamentales a desarrollar por el usuario es el ingreso de la función de la cual desea conocer sus ceros o raíces. Si durante el ingreso de la misma se produjera algún error que hace que la

expresión no sea válida, aparecerá el mensaje de la figura 3, en el cual se indica al usuario como están definidos los principales operadores y funciones.

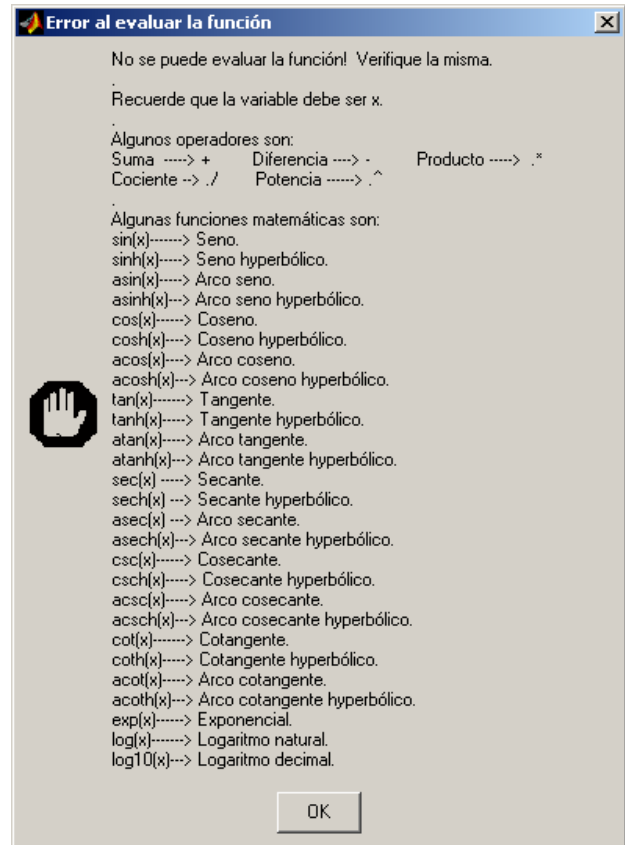


Figura 3

De forma similar, el software controlará que los valores asignados por el usuario (cantidad de iteraciones, cota de error, valor inicial o intervalo de análisis) sean aceptables.

Otra característica relacionada con el funcionamiento del software, es que éste exige que el usuario grafique la función previamente al cálculo de las raíces. De esta manera, se podrá realizar el ajuste de los datos ingresados según la gráfica obtenida. Si se intentan calcular los ceros de la función sin realizar la gráfica, aparecerá un mensaje como el que se muestra en la figura 4.



Figura 4

En cualquier momento de la ejecución del software, el alumno puede recurrir a la ayuda (ver figura 2), que incluye las principales características teóricas de cada método.

En la figura 5, se muestra, por ejemplo, la descripción teórica del método de la secante. Esta descripción está dividida en etapas en las cuales el texto y los objetos a los que éste hace referencia van apareciendo en el gráfico en forma conjunta. Cada vez que el usuario hace clic en el botón *Continuar*, se pasa a una nueva etapa. De esta manera, cada alumno emplea el tiempo que cree necesario para el análisis de los textos, fórmulas y gráficos existentes, facilitando su comprensión. Si el usuario no necesita o no quiere visualizar todas las etapas de la ayuda, puede salir de la misma y regresar a la aplicación del método seleccionado.

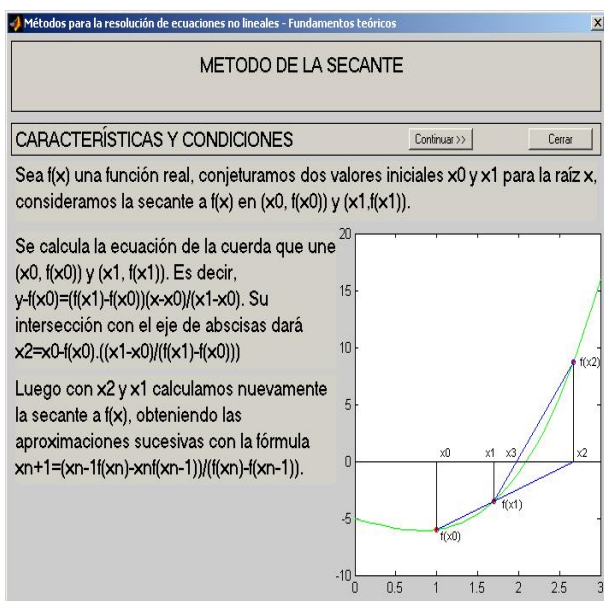


Figura 5

Otro aspecto importante durante la implementación de los diferentes métodos, es que la última aproximación que se ha obtenido aparece en el gráfico de color verde (figura 2), distinguiéndose de las anteriores que aparecen de color negro. Esto permite un mejor seguimiento de las últimas aproximaciones obtenidas.

En la parte izquierda de la pantalla se visualizan los resultados (figura 2): cantidad de iteraciones realizadas hasta el momento, última aproximación obtenida de la raíz y el

valor de la función en ese punto, que permite relacionarlo con el error ingresado. Podemos ver también que el cuadro de texto en el que se indica la solución es de fondo blanco, aunque durante la búsqueda de la misma presenta igual color que el resto de los cuadros de texto. Este cambio de color (de gris a blanco) pretende resaltar el evento ocurrido: finalización del método de forma satisfactoria con los parámetros ingresados. Éste no se produce, si el método no encuentra una solución. En este caso, aparecerá algún mensaje mostrando la condición por la cual se detuvo la ejecución del método seleccionado. Este mensaje indicará lo siguiente, según el método: inexistencia de raíces en el intervalo indicado, insuficiente cantidad de iteraciones, error admisible inadecuado o no convergencia del método. En las figuras 6, 7 y 8 se pueden observar los mensajes de este tipo.

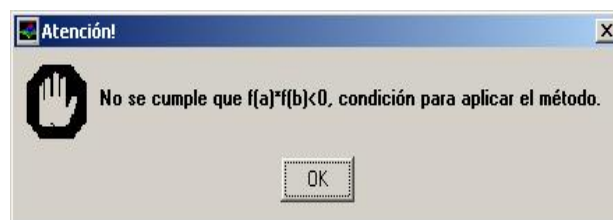


Figura 6



Figura 7

## Otras opciones

A partir de la primera pantalla (figura 1), el alumno podrá acceder a la información relacionada con la comparación de los diferentes métodos numéricos. Esta aparecerá al hacer clic sobre el botón *Comparación de la características de los métodos*. De esta forma, tendrá acceso a la pantalla que se muestra en la figura 8.

Métodos para la resolución de ecuaciones no lineales							
COMPARACIÓN DE LOS DIFERENTES MÉTODOS							
Método	Cantidad de valores iniciales	Convergencia relativa	Estabilidad	Exactitud	Alcance de aplicación	Esfuerzo de programación	Comentarios
Bisección	2	Lenta	Siempre converge	Buena	General	Fácil	Permite el conocimiento previo de las iteraciones necesarias.
Regla Falsa	2	Media	Siempre converge	Buena	General	Fácil	
Iteración de punto fijo	1	Lenta	Puede no converger	Buena	General	Fácil	
Newton	1	Rápida	Puede no converger	Buena	Limitada si $f'(x) \neq 0$	Fácil	Requiere la evaluación de $f'(x)$ . Aplicable a raíces complejas.
Newton Modificado	1	Rápida	Puede no converger	Buena	Limitada si $f'(x) \neq 0$	Fácil	Requiere la evaluación de $f'(x)$ . Aplicable a raíces complejas.
Secante	2	Entre lenta y media	Puede no converger	Buena	General	Fácil	Los valores iniciales no tienen que encerrar la raíz. Aplicable a raíces complejas.

Figura 8

Aquí también el usuario podrá pausar la ejecución mientras aparecen los diferentes aspectos que se están comparando, para realizar, si lo desea, un análisis más profundo de la información.

## Impacto del software en el desarrollo de las clases

Incorporamos el software educativo que hemos diseñado cuando comenzamos a desarrollar la unidad correspondiente a la resolución de ecuaciones no lineales. Su uso modifica la planificación de las actividades tanto teóricas como prácticas, ya que constituye una herramienta sumamente adecuada al momento de ejemplificar los desarrollos teóricos de los métodos correspondientes a esta temática.

Los métodos implementados son los siguientes:

- Iterativo de punto fijo
- Bisección
- Regula falsi
- Secante
- Newton
- Variante del método de Newton

Con la aplicación en el curso de Cálculo Numérico de versiones previas del software y a través de encuestas realizadas en cursos de años anteriores, pudimos obtener la siguiente información:

- Los alumnos del ciclo lectivo 2004 utilizaron el software en una versión anterior que sólo implementaba dos de los métodos (bisección y regla falsi), y que se podía aplicar a un conjunto de ejemplos. Al experimentar con el software y luego de la encuesta realizada, se pudo observar claramente que el 100% de los alumnos rescató la experiencia como positiva. El uso del software facilitó la comprensión de los métodos numéricos abordados en este caso, ya que se pudieron visualizar en forma clara y sencilla sus características. Además, consultados sobre las modificaciones que realizarían al software, el 40% de los alumnos mencionó la posibilidad de poder incorporar todos los métodos vistos. El 60 % restante no emitió opinión alguna.
- Los alumnos del ciclo lectivo 2005 utilizaron el software que implementaba los seis métodos para la resolución de ecuaciones no lineales. La mayoría de los alumnos (el 70%), señaló que la utilización del software facilitaba la resolución de ejercicios. Le adjudicaron una utilidad netamente práctica, relativizando el apoyo que pretendíamos que el software constituyera para facilitar la comprensión de los aspectos teóricos. Por otro lado, una de las modificaciones más solicitadas por los alumnos fue la de poder aplicar los métodos a cualquier ecuación y no sólo a un conjunto de ecuaciones propuestas. En consecuencia, luego de las diferentes etapas de prueba del software con los alumnos y considerando los distintos datos recabados por medio de las encuestas, fuimos modificando algunos aspectos del mismo. El software actualmente implementa los seis métodos estudiados en la unidad curricular correspondiente, y se puede aplicar para cualquier función que ingrese el alumno.

Además ofrece al usuario la posibilidad de acceder a ayudas teóricas sobre el funcionamiento del método.

Por medio de la observación de las clases, desde la Cátedra se detectó que si bien en las clases prácticas se utilizaba el software, como éste no era incluido directamente en actividades relacionadas con las evaluaciones parciales, existía cierta resistencia a dedicarle suficiente tiempo. Hasta ese momento, los alumnos percibían la utilización del software como una actividad meramente ilustrativa. Si bien describía los métodos y su funcionamiento de forma clara, no les significaba en forma directa ningún aporte al momento de resolver los ejercicios en forma manual en los exámenes parciales. Por ello es que se decidió incluir en estas evaluaciones, ciertas aplicaciones que se resolvieran por medio de la utilización del software. De esta manera, se logró una mayor valoración por parte de los alumnos de aquellas actividades que incluían su uso.

También se detectó que los alumnos, durante la utilización del software, dedicaban la mayor parte del tiempo a buscar ejemplos para su aplicación. En muchos casos recurrían a los ya resueltos en las actividades prácticas (con calculadora y en papel). Por esta razón, se decidió implementar junto al software un conjunto de ejercicios para que los alumnos pudieran aplicar los diferentes métodos. En estos ejercicios, se sugieren los posibles valores iniciales, intervalos de análisis, cota de error y cantidad de iteraciones. Con estos datos los alumnos comienzan a trabajar con el software. Según las respuestas obtenidas, deberán modificar los datos a efectos de mejorar los resultados.

## **Acciones futuras**

Considerando las implementaciones del software en sus distintas versiones y los resultados alcanzados, se fueron definiendo las actividades futuras a desarrollar.

## **Actividades relacionadas con el software**

Dado que el desarrollo del software se encuentra avanzado, llegamos a una instancia en la cual surge la necesidad de abocarse a la portabilidad del mismo. Es por ello que trabajando con versiones de MATLAB superiores a la 5.1, que es con la que elaboramos el software educativo, trataremos de que el mismo se pueda ejecutar en la mayoría de las PC con los menores requerimientos posibles.

## **Actividades de planificación**

Las actividades relacionadas a la introducción del software en el proceso de enseñanza y aprendizaje que ya se han implementado, como describimos anteriormente, se deben afianzar y ampliar. Así, se espera que las potencialidades del software se pueden aprovechar y no se reduzca a una actividad ilustrativa de los conceptos desarrollados tanto en las clases teóricas como en las prácticas.

Es importante establecer con claridad el contrato didáctico en el curso. Es decir, precisar en que forma y tiempo intervendrán:

- La computadora y/o software en el curso.
- El profesor con las explicaciones pertinentes y el alumno con sus dudas e inquietudes. Es necesario aclarar el rol de cada uno.

Se continuará con la planificación de actividades tendientes a la utilización del software desarrollado en las clases teóricas, ya que permite ejemplificar mucho más fácilmente los diferentes conceptos teóricos.

También se buscará que por medio de las actividades implementadas durante las clases prácticas, el alumno no sólo logre una mayor comprensión de los fundamentos de cada método numérico, sino que además pueda aplicar el mismo en la resolución de sus actividades prácticas y de las evaluaciones parciales.

Finalmente, nos resta analizar la relación entre la utilización del software y el rendimiento académico de los alumnos. Desarrollaremos actividades que permitan registrar cómo y en

qué grado la utilización del software modificó el rendimiento académico de los alumnos.

## Conclusiones

Luego del desarrollo y utilización de este software, se han obtenido conclusiones relacionadas principalmente con:

- Las actividades necesarias al momento de desarrollar un software educativo para matemática.
- Las expectativas que genera en los alumnos la posibilidad de utilizar esta herramienta.
- Las modificaciones que se deben incluir en la planificación desde la Cátedra.

Considerando el primero de los aspectos, podemos decir que es amplio el esfuerzo que demanda la realización de un software educativo para temas de Matemática, ya que es necesario que el alumno logre relacionar una gran cantidad de conceptos y posibilidades. No obstante ello, este esfuerzo se ve compensado cuando los alumnos logran combinar la comprensión y la profundización teórica con las actividades prácticas, a lo que contribuye en gran medida la inclusión del software educativo [7].

Referido al segundo de los puntos mencionados, cabe destacar que durante las primeras implementaciones del software observamos que los alumnos que realizaban las actividades con el software, priorizaban aquellas que serían evaluadas en los exámenes parciales (entrega de ejercicios resueltos con calculadora y presentados en papel). Esta actitud se modificó cuando desde la Cátedra se decidió cambiar las condiciones y características de las evaluaciones parciales, incluyendo en las mismas la resolución de ejercicios utilizando el software.

Indudablemente existen riesgos de utilizar un software y creer que su sola implementación en algunas clases, durante el desarrollo del cursado, tendrá efectos positivos en los resultados obtenidos con los alumnos. Esto

motiva la necesidad de trabajar el tercero de los puntos planteados anteriormente (las modificaciones que se deben incluir en la planificación desde la Cátedra). Si bien las ventajas de utilizar software educativos son claras (aumenta la motivación, facilita la realización de cálculos y la visualización, se adapta a los ritmos de cada alumno, entre otras), dependen ampliamente de la modalidad con que se implemente el mismo. La implementación del software debe ser acompañada por una minuciosa planificación de las actividades a desarrollar antes, durante y después de su uso. Entre estas podemos mencionar: las guías a implementar para la utilización del software, las actividades que se desarrollarán posteriormente, en qué instancias se considerarán las mismas y cómo se evaluará el desempeño de los usuarios. El éxito de un software educativo, en lo referido a la mejora de procesos de enseñanza y aprendizaje, dependerá en gran parte del entorno educativo diseñado por el docente.

## Referencias

- [1] Ausubel, D. P. y Novak, J. D. 1978. *Educational Psychology: "A Cognitive View"*. Holt, Rinerhart and Winston. New York.
- [2] Cuevas Vallejos, C. 2000. *¿Qué es Software Educativo o Software para la Enseñanza?*  
<http://www.matedu.cinvestav.mx/~ccuevas/SoftwareEducativo.htm>
- [3] Marquès, P. 1996. *El Software Educativo*. Universidad Autónoma de Barcelona. Disponible en: [www.doe.d5.ub.es](http://www.doe.d5.ub.es)
- [4] Galvis Panqueva, A. 1992. *Ingeniería de Software Educativo*. Facultad de Ingeniería. Universidad de Los Andes. Bogotá. Colombia.
- [5] Cataldi, Z. 2000. *Una Metodología para el Diseño, Desarrollo y Evaluación de Software Educativo*. Universidad Nacional de La Plata. Facultad de Informática. Argentina.



- [6] Organización para la Cooperación y Desarrollo Económicos (OCDE). 2001. *Los Desafíos de las Tecnologías de la Información y las Comunicaciones en la Educación*. Ministerio de Cultura y Educación. España.
- [7] Soler Selva, V. 2003. *La Experimentación Asistida con Calculadora: Una vía para la educación científico-tecnológica*. Revista Iberoamericana de Educación. Disponible en:  
[www.campusoei.org/revista/deloslectores/553Soler.pdf](http://www.campusoei.org/revista/deloslectores/553Soler.pdf)
- [8] Mathews, J. y Fink, K. 2000. *Métodos Numéricos con MATLAB*. Prentice-Hall. España.
- [9] Nakamura, S. 1992. *Métodos Numéricos Aplicados con Software*. Prentice-Hall Hispanoamericana, S.A. México.
- [10] Nakamura, S. 1997. *Análisis Numérico y Visualización Gráfica con MATLAB*. Pearson Educación. México.
- [11] The Math Works. 1993. *MATLAB. Building a Graphical User Interface. Reference Guide. User's Guide. Version 5.1*.

---

<sup>1</sup> Trabajo desarrollado en el Proyecto de Tesis de la Maestría en Tecnología Informática Aplicada en Educación, UNLP, del Prof. Pizarro R., dirigida por el Ing. De Giusti, A. y co-dirigida por la MSc. Ascheri, M.