

Los Alumnos de la Licenciatura en Ciencias de la Computación como Científicos y Educadores en Formación

Francisco Bavera

Departamento de Computación,
FCEFQyN, Universidad Nacional de Río Cuarto,
Ruta 36 km 601, Río Cuarto, Argentina
pancho@dc.exa.unrc.edu.ar

Resumen

Nuestra principal meta como docentes es formar a nuestros alumnos como científicos (personas que disfruten jugando con ideas) y como educadores (personas que buscan transmitir ideas y puedan hacerlo de manera efectiva). Creemos que lograr estas metas es muy importante en el desarrollo personal y profesional de nuestros alumnos. Buscamos brindarles a los alumnos la experiencia de jugar y disfrutar con la resolución creativa y original de problemas complejos. Consideramos muy importante ayudar a los alumnos a vislumbrar que lo que están aprendiendo es importante e interesante, y tienen el deber de compartirlo con los demás. Para que los alumnos afiancen firmemente sus conocimientos del método científico, deben experimentar directamente con la formulación de hipótesis, el diseño experimental, la verificación de la hipótesis y

Introducción

De estudios informales (nuestra propia experiencia, encuestas y de discusiones con estudiantes y docentes) hemos determinado que los alumnos de cuarto año de la Licenciatura en Ciencias de la Computación de la Facultad de Ciencias Exactas Físico-Químicas y Naturales de la Universidad Nacional de Río Cuarto no consideran sus actividades como “hacer ciencia”. Esta apreciación se debe fundamentalmente a su desconocimiento de que es “ciencia” y el “hacer científico”. Esta sensación de los alumnos proviene (1) de la ausencia de actividades en las que se recalque el proceso

el análisis de los datos para su posterior presentación. La currícula debe proveer esta experiencia con el objetivo vital de que los estudiantes hagan ciencia, no solo lean sobre ciencia. En este trabajo presentamos el enfoque que estamos implementando con el fin de que los alumnos logren usar sus conocimientos en la resolución creativa y original de problemas complejos y en su capacidad para transmitir estas resoluciones. Se busca que los alumnos logren un pensamiento científico y creativo, que los lleve a una mejor comprensión asumiendo una actitud crítica y participativa.

Palabras Clave: Ciencias de la Computación, Autómatas y Lenguajes, Compiladores, Innovación Pedagógica.

científico que realizan constantemente; (2) del escaso contacto que tienen con investigaciones científicas y eventos científicos (artículos, revistas, seminarios, charlas, workshops, congresos, entre otros); y (3) del escaso requerimiento que tienen para redactar, discutir y/o presentar resultados científicos (tanto propios como ajenos).

Estos alumnos, en su gran mayoría (más de un 90%), demuestran todos los años una gran apatía y poca motivación para ser pro-activos en su formación con una marcada ausencia de conciencia crítica y curiosidad para con los conocimientos impartidos y por adquirir nuevos conocimientos.

Hemos concluido, que lo más importante de sus carencias no es que sepan más o menos contenidos relativos a Ciencias de la Computación, sino que sus carencias provienen de una falta de capacidad de razonamiento, de comprensión del lenguaje propio de las Ciencias de la Computación, de entrenamiento en el estudio constante y metódico. Es válido afirmar que se aprende a hacer mientras se practique el hacer. Significa que los estudiantes no aprenderán a pensar críticamente, analizar la información, comunicar las ideas y resultados, alcanzar conclusiones lógicas, colaborar con otras personas, o crear soluciones alternativas a menos que les den el tiempo y oportunidad de desarrollar y de practicar estas habilidades.

Los diferentes temas involucrados en el área de Teoría de Autómatas y Lenguajes Formales, como así también, los relacionados con el diseño e implementación de Compiladores conforman una parte muy relevante en la currícula de Ciencias de la Computación. El nivel de abstracción de estos conocimientos hace que se presenten dificultades tanto a la hora de enseñarlos como a la hora de aprenderlos. Nuestra experiencia en el dictado de estas materias muestra que muchos alumnos no sienten ninguna motivación por aprender (y comprender) estos temas como nosotros esperamos. Estas dificultades se ven reflejadas, en mayor medida, en la instancia de evaluación final de las materias. En esta instancia, se desnudan falencias en la formación que nos resultan, por lo menos, desconcertantes. Los actuales alumnos parecen no tener suficientemente desarrolladas las destrezas vinculadas con la capacidad de abstracción, considerada esencial para el progreso científico [KOZEN05]. Inmersos en una cultura de la imagen, ya no consideran al libro como el emblema del conocimiento y la cultura, y ha sido reemplazado por la televisión y la computadora. Además, existe un proceso social por el cual los nuevos alumnos han sido

socializados hacia la abulia y la ausencia, mientras que la escuela requiere dedicación y atención. En este contexto, nuestros métodos “tradicionales” de enseñanza mostraron cierta ineficacia y percibimos que somos incapaces de activar las formas de inteligencia propias del nuevo estilo cognoscitivo de los alumnos que se muestran cada vez más desmotivados y apáticos [CAVADINI05].

En la materia Taller de Diseño de Software, se enseñan un número considerable de técnicas (tradicionales en los cursos de construcción de compiladores). Muchas de estas técnicas son casos particulares de técnicas más fundamentales. Pero, se deja librado al alumno: (1) asociar la aplicación de determinada técnica en otros contextos; (2) determinar, o no, la generalización de la técnica; y (3) investigar, o no, otras alternativas de diseño e implementación. La práctica actual conlleva a dos riesgos: (1) que el alumno no pueda vislumbrar que el campo de aplicación de las técnicas dadas es más amplio; y (2) que el alumno aprenda que la “única” forma de diseñar e implementar un compilador es la aplicación de las técnicas vistas en clase.

Los problemas detectados se reflejan en:

- Una dificultad generalizada para usar de manera fructífera la bibliografía de las materias.
- No muestran interés en leer la bibliografía recomendada. Esperan que el docente resuma los conocimientos teóricos.
- Tienen poca experiencia en la lectura de material de divulgación científica (artículos, revistas, etc.) y ninguna en la transmisión de resultados científicos (escritura y/o exposición). Por este motivo creemos que no se sienten motivados a contrastar, comparar y evaluar con una mirada crítica diversos enfoques que resuelven un mismo problema.

- La relación entre los conceptos teóricos, los trabajos prácticos y los talleres de implementación no se enfatiza suficientemente en el dictado de las materias (y en algunos casos no es requerido formalmente). Lo que se traduce, en que los estudiantes, en general, no consideran a la teoría como una ayuda y/o una fundamentación a la hora de resolver problemas.
- Un aislamiento conceptual entre los temas tratados en diferentes materias e incluso en la misma materia. No tienen desarrollada una gran capacidad de asociar y aplicar los conocimientos a nuevos problemas o nuevos contextos.
- No disfrutaban el proceso involucrado en resolver problemas.
- Extrema falta de independencia para desarrollar soluciones propias a los problemas planteados, poca creatividad para diseñar y en general “exigen” resoluciones “modelo” de los problemas a resolver.
- Los alumnos no se sienten “científicos”, ni tienen en claro que significa ser “científicos”.

El objetivo principal de las acciones elegidas y presentadas en este proyecto es lograr que, al finalizar el cuarto año de la Lic. en Cs. de la Computación, los alumnos adquieran una sólida base de los conocimientos de la Teoría de Autómatas y Lenguajes y en el desarrollo de Compiladores. Esta base sólida se debe ver reflejada no solo en las evaluaciones finales sino en la capacidad de los alumnos de usar estos conocimientos en la resolución creativa y original de problemas complejos y en su capacidad para transmitir estas resoluciones. Se busca que los alumnos logren un pensamiento científico y creativo, que los lleve a una mejor comprensión asumiendo una actitud crítica y participativa. Por último, pretendemos darle continuidad a la innovación a lo largo de la carrera. En una segunda

instancia, articulando entre materias de segundo, cuarto y quinto año.

Marco Institucional

Universidad Nacional de Río Cuarto, Facultad de Ciencias Exactas Físico-Químicas y Naturales, Licenciatura en Ciencias de la Computación. Asignaturas involucradas: “Autómatas y Lenguajes” y “Taller de Diseño de Software”, correspondientes a cuarto año. Estas asignaturas tienen régimen cuatrimestral, dictándose en el primer y segundo cuatrimestre, respectivamente. En los últimos 6 años, cursaron cada una de estas asignaturas entre 13 y 22 alumnos.

Durante el primer cuatrimestre se dicta la materia Autómatas y Lenguajes. Esta materia involucra un gran contenido teórico y la aplicación de conocimientos avanzados de matemática y álgebra. En esta materia se relacionan conocimientos provenientes de distintas áreas: teoría de autómatas, lingüística, estudios matemáticos sobre la decibilidad y complejidad de los algoritmos y teoría de reconocimiento de lenguajes formales. En el segundo cuatrimestre se dicta la materia Taller de Diseño de Software. Esta materia, se centra en el diseño e implementación de compiladores, para lo cual, se basa en los conocimientos abordados en Autómatas y Lenguajes. Los diferentes temas involucrados en el área de Teoría de Autómatas y Lenguajes Formales, como así también, los relacionados con el diseño e implementación de Compiladores conforman una parte muy relevante en la currícula de Ciencias de la Computación.

También se pretende involucrar (en una segunda instancia) las asignaturas de “Programación Avanzada” y “Estructuras de Datos y Algoritmos” correspondientes a segundo año de las carreras de Analista en Computación, Profesorado y Licenciatura en Ciencias de la Computación. Estas asignaturas

tienen régimen cuatrimestral, dictándose en el primer y segundo cuatrimestre, respectivamente. En los últimos años, cursaron en promedio 50 alumnos cada una de estas asignaturas. Además, se involucra a la asignatura “*Computabilidad y Complejidad*” correspondiente a quinto año de la carrera Licenciatura en Ciencias de la Computación. Esta asignatura tiene régimen cuatrimestral, dictándose en el segundo cuatrimestre. En los últimos años, cursaron en promedio 10 alumnos. La materia “*Autómatas y Lenguajes*” y la materia “*Computabilidad y Complejidad*” abarcan los temas del área Teoría de la Computación.

Proyecto Innovador

Fundamentación de la Innovación

Nuestra principal meta como docentes es formar a nuestros alumnos como científicos (personas que disfruten “jugando” con ideas) y como educadores (personas que buscan transmitir ideas y puedan hacerlo de manera efectiva). Creemos que lograr estas metas, aunque sea parcialmente, es muy importante en el desarrollo personal y profesional de nuestros estudiantes. Para lograr esto, buscamos romper con la práctica tradicional de transmitir a los alumnos la sensación de que los problemas planteados ya han sido resueltos por otras personas. Esta situación no es motivadora y además puede ser desmoralizante. Nosotros buscamos brindarles a los estudiantes la experiencia de “jugar” y “disfrutar” con la resolución creativa y original de problemas complejos. También, consideramos muy importante ayudar a los alumnos a darse cuenta de que lo que están aprendiendo es importante e interesante, y tienen el deber de compartirlo con los demás.

Para que los alumnos afiancen firmemente sus conocimientos del método científico, deben experimentar directamente con la formulación de hipótesis, el diseño experimental, la verificación de la hipótesis y el análisis de los datos para su posterior presentación. La currícula debe proveer esta experiencia con el objetivo vital de que los estudiantes “hagan ciencia”, no solo “lean acerca de ciencia”.

La propuesta curricular en el área de Ciencias de la Computación de la ACM y la IEEE [CC01], a la que adhirió la Propuesta de Currícula de la Red de Universidades Nacionales con Carreras en Informática (RedUNCI) recalca que: “Los estudiantes deben desarrollar una comprensión del método científico y adquirir experiencia en investigación”.

Además, la ACM y la IEEE definen como una de las líneas de trabajo de los profesionales en ciencias de la computación: el desarrollo efectivo de caminos para resolver problemas de computación. Esto se refiere a la aplicación o desarrollo de teoría en ciencias de la computación y conocimiento de algoritmos para asegurar la mejor solución posible para problemas computacionales [CC05].

Adquirir estas capacidades no es simplemente comprender los nuevos conceptos. Es, además, desarrollar la capacidad de resolver problemas usando los conocimientos adquiridos y disfrutar haciéndolo. Los métodos que aprendimos en la escuela como el de la división con resto o las operaciones con fracciones aún descriptos por un enunciado interesante, no deberían entrar en la categoría de problemas. Los problemas verdaderos son aquellos que nos apasionan, que llaman nuestra atención y ponen en funcionamiento toda nuestra capacidad de razonamiento. Un problema, debe permitirnos visualizar lo importante y gratificante que es aplicar nuestros conocimientos y

razonamientos para entender fenómenos no triviales.

En la moderna concepción pedagógica, el docente es mediador, tiene que ayudar al alumno a adquirir conocimiento, habilidades y estrategias para satisfacer la necesidad de progresar, aprender cada día más y resolver problemas que surgen ante demandas impredecibles. El docente mediador interactúa entre mensaje y aprendiz, siendo éste un participante activo.

Según Williams y Burden [WB97]: El docente tiene que ayudar al alumno a interactuar con los materiales de distintas maneras hasta que se conviertan en verdaderos aprendices auto-dirigidos y, en este caso, lo logran a través de herramientas que potencialmente serán su fuente e instrumento del manejo de información en su futura vida profesional.

Consideramos que es fundamental que, como docentes, tomemos parte activa y busquemos alternativas para lograr que los alumnos afiancen aquellos conocimientos que juegan un rol central en ciencias de la computación y fortalezcan las habilidades necesarias para aplicarlos y relacionarlos en distintas situaciones. Puntos fundamentales en el conjunto de herramientas que deben poseer los profesionales en Ciencias de la Computación.

Objetivos de la innovación

El objetivo principal del proyecto es lograr que, al finalizar el cuarto año de la carrera licenciatura en Ciencias de la Computación, los alumnos adquieran una sólida base de los conocimientos centrales de la Teoría de Autómatas y Lenguajes y en el diseño e implementación de Compiladores. Esta base sólida se debe ver reflejada no solo en las evaluaciones finales sino en la capacidad de los alumnos de usar estos

conocimientos en la resolución creativa y original de problemas complejos y en su capacidad para transmitir estas resoluciones.

Para llevar esto a cabo proponemos los siguientes objetivos específicos en lo referente a nociones y conceptos que esperamos que los estudiantes hayan adquirido al final de las materias:

- Dominio de los fundamentos de autómatas, gramáticas y lenguajes formales. Su relación, formas de representación y algoritmos relacionados.
- Dominio de los conceptos involucrados en las distintas etapas del diseño e implementación de compiladores.
- Fortalecer los conocimientos lógico-matemáticos necesarios, los conocimientos de diseño de sistemas complejos y los conocimientos necesarios para la implementación y manipulación eficiente de estructuras de datos avanzadas.
- Aplicación de estos fundamentos en casos prácticos, como por ejemplo, su uso en *parsing* o su aplicación a la bioinformática. Generalización de las técnicas dadas.
- Generar espacios, dentro de la currícula, para que los alumnos “hagan ciencia” y así desarrollen experiencia con el proceso científico. Se busca enseñarlos a construir un pensamiento científico y creativo, que los lleve a una mejor comprensión asumiendo una actitud crítica y participativa.
- Conocimiento de herramientas de software variadas, su instalación y uso. Estas herramientas pueden no estar relacionadas necesariamente a los contenidos de las materias.
- Generar mayor articulación entre las materias involucradas en el proyecto y las materias de tercer año.
- Darle continuidad a la innovación a lo largo de la carrera. En esta primer instancia, articulando entre materias de segundo, cuarto y quinto año.

- Evaluar la experiencia y formular recomendaciones que permitan implementar y mejorar nuevas experiencias de innovación pedagógica.
- Esperamos que los estudiantes adquieran y mejoren ciertas habilidades tanto en lo específico de la disciplina como de los hábitos de estudio y en los métodos para la resolución de problemas:
- Independencia y creatividad para generar soluciones a problemas nuevos usando las herramientas aprendidas.
- Desarrollo del hábito de lectura (no solo de la bibliografía obligatoria).
- Experiencia en la presentación (oral y escrita) de resultados.
- Uso de herramientas de comunicación, como foros de discusión, correo electrónico y pizarras virtuales, lo cual generará una mayor interacción alumno-alumno y alumno-docente.
- Mayor motivación, curiosidad, pro-actividad y conciencia crítica durante el proceso de aprendizaje.

Descripción de la Innovación:

La innovación se centra en las actividades a realizar en las materias “Autómatas y Lenguajes” y “Taller de Diseño de Software” pertenecientes a cuarto año de la Licenciatura en Ciencias de la Computación. Pero, se pretende dar continuidad a las actividades implementadas en las materias de “Programación Avanzada” y “Estructura de Datos” correspondientes al segundo año. Además, las actividades en cuarto año, tienen su continuidad en la materia “Computabilidad y Complejidad” correspondiente a quinto año.

El constructivismo es una teoría de aprendizaje que pregona que el conocimiento es construido activamente por los estudiantes, y no es absorbido pasivamente de exposiciones o textos [ARI01]. Esta teoría afirma que la construcción de nuevos conocimientos se realiza recursivamente, basándose en el

conocimiento previo. Este proceso es activamente guiado por el docente y su meta es que el estudiante pueda construir conocimientos significativos. El alumno selecciona y transforma información, construye hipótesis, y toma decisiones basándose en una estructura cognitiva que provee significado y organización para sus experiencias. Los docentes y los estudiantes deben participar de un continuo dialogo donde una de las tareas del docente es transformar la información a un formato apropiado según el estado actual de entendimiento del alumno.

Los principales puntos que rigen el aprendizaje significativo son los siguientes: (1) Aprender es una actividad social y contextual. (2) Aprender es un proceso activo. (3) El conocimiento previo juega un rol muy importante. (4) Actividades de refuerzo y de extensión promueven un aprendizaje significativo [BRU90].

Siguiendo estos principios y basándonos en la experiencia presentada en [CGM04] se presentan las actividades propuestas. Estas estrategias se combinaron y sumaron a las ya tradicionales en el dictado de las materias (clases expositivas, trabajos prácticos y talleres de implementación).

1. Contextualización histórica de Ciencias de la Computación. Se incluyó explícitamente detalles históricos de la evolución de los tópicos introducidos y su posible relación con el contexto histórico en que se desarrolló. Entender la evolución de los conceptos es muy enriquecedor y motivante para nuestros estudiantes. Esto ayuda al alumno a vislumbrar el proceso científico inmerso en las actividades que se llevaron a cabo para obtener el resultado estudiado. Con el fin de integrar la historia de Ciencias de la Computación con los tópicos presentados, además de los comentarios en clase se adicionan links a notas biográficas y material adicional en las

presentaciones y en las páginas web de las materias.

2. Aprendiendo activamente con herramientas interactivas y de simulación. Se complementa el uso tradicional de papel y lápiz en la resolución de ejercicios prácticos con el uso de herramientas interactivas (JFlap [JFLAP] y The Java Computability Toolkit [JCT]) y de simulación (ASSIST [ASSIST]). Estos programas educativos nos permiten experimentar con muchos de los conceptos introducidos. Además, los alumnos cuentan con un mecanismo para validar los resultados obtenidos adquiriendo independencia y cierto grado de autocorrección. Además, pueden resolver ejercicios de mayor grado de dificultad y/o tediosos de resolver en papel, recibiendo un rápido feedback en el proceso de resolución. Estas herramientas motivaron a indagar nuevos problemas y sus posibles soluciones, como así también, a contrastar distintas soluciones. El uso de simuladores de algoritmos visuales facilitó a los alumnos el proceso de comprensión del comportamiento de dichos algoritmos. Además, “ver” el comportamiento de los algoritmos (por ejemplo usando estos simuladores) motivará (o por lo menos es lo que esperamos se produzca) a los alumnos a proponer optimizaciones de los algoritmos conocidos y, en los mejores casos a intentar proponer nuevos algoritmos o variantes de los ya conocidos.

3. Usando el conocimiento previo de los alumnos. En este punto nuestras actividades apuntan a mostrarle a los alumnos la relación entre los contenidos de las materias y su relación con los lenguajes de programación actuales (como JAVA, C, C++, Eiffel) los cuales son familiares para los alumnos. No es muy común, por ejemplo, encontrar referencias o ejemplos concretos que relacionen los conceptos concernientes al núcleo central de la Teoría de la Computación con los lenguajes de

programación actuales. Creemos que no recalcar y usar este tipo de relaciones es causa de la pérdida de interés y motivación de muchos estudiantes que perciben que los aspectos fundamentales de ciencias de la computación son “muy matemáticos” y no tienen ninguna relación directa y palpable con sus futuras actividades profesionales. Creemos que estas relaciones son puntos de motivación para introducir y relacionar muchos conceptos abstractos de la teoría.

4. El estudiante como un investigador en formación. Para ayudar a los estudiantes a ser científicos nos enfocamos en generar experiencias que les permitan disfrutar de descubrir soluciones creativas y originales a problemas complejos. Por ejemplo, utilizamos problemas que inicialmente, los estudiantes, piensan que son imposibles de resolver. Entonces una vez que obtuvieron la solución se sienten “invencibles”. Hemos notado que este proceso despierta cierta pasión por resolver problemas y “jugar” con ideas. Para solucionar algunos de estos problemas es necesario que los alumnos realicen previamente una revisión bibliográfica y luego se les solicita un reporte técnico comentando la solución desarrollada y el proceso realizado para obtenerla.

5. El estudiante como un educador. Para ayudar a los estudiantes a transmitir conocimientos nos enfocamos en mostrarles que lo que aprenden es importante e interesante y que tienen el deber de compartirlo con otras personas. Entre las actividades que deben realizar los alumnos mencionamos: presentación de soluciones a problemas complejos, realización de correcciones a sus compañeros, proposición de mejoras a las soluciones de sus compañeros, realización de autocorrecciones.

Es de notar que fue necesario fomentar el uso de distintas formas de comunicación y participación además de la interacción en el

aula. Para esto se generó un foro de discusión usando listas de correo electrónico para comentar, consultar y realizar correcciones de las actividades propuestas. También, se pretende utilizar en un futuro pizarras virtuales en donde cada alumno o grupo de alumnos publicaran la solución del problema que obtuvieron (todas las resoluciones se publicaran simultáneamente) los alumnos comentarán, discutirán las distintas soluciones y de ser necesario propondrán modificaciones a los restantes alumnos o grupos. Las correcciones de cada solución solo la podrá realizar el alumno (o grupo) que la publicó.

La experiencia nos ha demostrado que los prácticos de laboratorio tradicionales son necesarios para lograr una mejor asimilación de los conceptos teóricos. Creemos que los prácticos de laboratorio complementan muy bien los conceptos teóricos con prácticas dirigidas, orientadas hacia la enseñanza del buen uso de las herramientas y sus aplicaciones.

Evaluación de los Resultados

La inclusión de mecanismos que permitan el seguimiento del alumno y del proceso de innovación son las herramientas que permitirán verificar el desarrollo del proyecto y la corrección de falencias en la ejecución del mismo. La motivación es doble: por un lado, ayudar a los docentes en la localización de lagunas cognitivas; por otro lado, poder utilizar los resultados como parte de la valoración general del conocimiento de los alumnos y en la valoración del desarrollo del proyecto. Esto genera información que retroalimentara el proyecto y permite revisar y reformular los objetivos y practicas. Adaptarlos a las necesidades y a la realidad.

El mecanismo de evaluación nos debe permitir comprobar los resultados y logros obtenidos por los alumnos en diferentes ámbitos y los beneficios de los cambios en la

práctica docente. Estas evidencias se obtienen de las evaluaciones formales y de los datos cuantitativos que demuestren un cambio (positivo o negativo); por ejemplo, aumento de las notas obtenidas, disminución de las brechas en la consecución de los objetivos entre grupos, aumento de asistencia en asignaturas relacionadas con la experiencia, disminución en la tasa de deserción, disminución en la tasa de recursantes, entre otros factores) o datos cualitativos específicos, tales como, un análisis sistemático del resultado de los alumnos comparado con resultados previos.

Dado que se trata de una experiencia de innovación, la evaluación nos debe permitir determinar si la experiencia tiene factibilidad de ser potencialmente transferible y/o sostenible en el tiempo.

Objetivos de la Evaluación

Con este proyecto no solo se pretende apreciar mejoras sustanciales en las calificaciones de los alumnos sino poder evaluar, también, mejoras integrales en los alumnos. Es decir, queremos medir el impacto de los cambios implementados en las materias involucradas en el proyecto, en las restantes materias de la Licenciatura y en las actividades realizadas por los alumnos para elaborar su tesina de grado.

Los objetivos específicos de la evaluación del proyecto son:

- Analizar el impacto real de la innovación pedagógica ¿Cumplió con las expectativas? ¿Falencias? ¿Virtudes?
- Determinar la relación entre el proceso implementado y el proyecto formulado. Determinar posibles causas y consecuencias. ¿Realmente se innovó la práctica docente?
- Determinar la mejora de los alumnos en su desempeño en actividades científicas (tanto

de adquisición, replicación y generación de conocimiento científico).

- Determinar la mejora de los alumnos en su desempeño en actividades de comunicación de información científica.
- Medir el grado de motivación, participación, pro-actividad, curiosidad, interés y conciencia crítica de los alumnos.
- Determinar el nivel de aprovechamiento de los alumnos de los recursos brindados.
- Evaluar la experiencia y formular recomendaciones que permitan implementar y mejorar nuevas experiencias de innovación pedagógica.

Metodología Usada para Evaluar

En primer lugar hemos de mencionar que la evaluación de este proceso de innovación pedagógica es de naturaleza multimetodológica, es decir cuantitativa y cualitativa. En ella se tratará de comprobar en qué medida el proceso de aprendizaje es influenciado por las actividades planteadas. Se utilizó una serie de encuestas (diseñadas por nosotros) con las que además se registró la situación de partida y permita analizando posteriormente el impacto de la incorporación de la innovación. Las encuestas están destinadas al docente y al alumno. Estas encuestas tratan de captar la experiencia de los docentes y de los estudiantes (tanto de las materias involucradas como de las que no lo están), dudas, expectativas y un análisis de la situación de los alumnos. Nivel de motivación, interés, pro-actividad. Indagan en los preconceptos que tiene los docentes sobre los alumnos y viceversa. Como así también la impresión de la innovación en ambas clases de actores intervinientes. Estas encuestas se realizaron al comenzar y al finalizar cada cuatrimestre lectivo.

Las respuestas referentes al papel del alumno proporcionadas por los docentes (en base a información recibida y percibida)

directamente de los alumnos nos sirven de alguna manera como control.

Conclusiones

Si bien es de reciente implementación estos cambios, los primeros resultados son prometedores y permiten pensar que los alumnos responden favorablemente.

Si bien, en esta primera edición el promedio de las notas obtenidas por los alumnos (en los exámenes parciales) es mayor a la de los últimos años. Como así también el grado de participación y de interés demostrado. Es importante resaltar también que se mostraron más curiosos y preactivos.

Pero, hay que esperar a obtener los resultados en las materias de los primeros años y el impacto en la formación de estos alumnos al llegar al último año de la carrera. Y el desempeño con que realicen la tesina de grado.

También, debemos esperar a obtener los resultados de los próximos años para así contar con una muestra más significativa. Es necesario contar con mayores datos para lograr un análisis concluyente sobre el impacto de estas actividades.

Lo que si queda en claro es que no hay que subestimar el potencial de los alumnos y que su motivación aumenta significativamente ante desafíos y problemas difíciles de resolver.

Esperamos que con el tiempo logremos cambiar la visión de los alumnos con respecto a que significa hacer ciencia y que logren tomar protagonismo en su formación como científicos y transmisores de conocimiento. Obteniendo profesionales con un mayor grado de compromiso hacia la sociedad y la búsqueda de soluciones que la beneficien.

Agradecimientos

Agradecemos las discusiones, aportes y colaboración en el desarrollo de este trabajo a María Marta Novaira, Gastón Scilingo, Germán Regis y Sonia Permigianni.

Bibliografía

[ARI01] M. Ben-Ari. *Constructivism in computer science education*, Journal of Computers in Mathematics and Science Teaching, v.20 n.1, p.45-73, January 2001.

[ASSIST] Simulador ASSIST
<http://www.cs.binghamton.edu/~software/>

[BRU90] J. Bruner. *Act of Meaning*. Harvard Univ. Press, 1990.

[CAVADINI05] Salvador Cavadini. *Compiladores e Intérpretes: En búsqueda de una práctica docente eficaz*. Monográfico final, Especialización en la Enseñanza de la Educación Superior, Universidad Católica de Cuyo.

[CC01] Computing Curricula 2001 ACM-IEEE. http://www.acm.org/education/education/education/curric_vols/cc2001.pdf

[CC05] Computing Curricula 2005: The Overview Report. ACM-IEEE. http://www.acm.org/education/education/curric_vols/CC2005-March06Final.pdf

[CGM] C. Chesñevar, M. Gonzalez y A. Maguitman. *Didactic Strategies for promoting Significant Learning in Formal Languages and Automata Theory*. ITiCSE'04. 2004

[JCT] The Java Computability Toolkit. <http://humboldt.sunyit.edu/jct/default.htm>

[JFLAP] S. H. Rodger. Jflap web site, 2006. <http://www.jflap.org>.

[KOZEN05] D. Kozen. *Automata and Computability*. 1997.

[WB97] M. Williams, R. Burden. *Psychology for Language Teachers: A Social Constructivist Approach*. Cambridge University Press. 1997