

Competir + Motivar + Hornero = aprender programación

Claudia Carina Fracchia¹, Pablo Kogan², Silvia Amaro³

¹ Departamento Computación Aplicada, Facultad de Informática, Universidad Nacional del Comahue, Neuquén, Argentina

² Departamento Teoría de la Computación, Facultad de Informática, Universidad Nacional del Comahue, Neuquén, Argentina

³ Departamento Programación, Facultad de Informática, Universidad Nacional del Comahue, Neuquén, Argentina

{carina.fracchia, pablo.kogan, silvia.amaro@fi.uncoma.edu.ar}

Resumen

La búsqueda de estrategias que permitan mejorar la enseñanza y aprendizaje de programación en estudiantes es una tarea indispensable tanto en el nivel secundario como en el universitario, donde se detectan problemas de motivación y de adecuación personal al nivel de dificultad. En este contexto se propone emplear como estrategia el desarrollo de torneos de programación para lograr fomentar la creatividad, el trabajo en equipo y la innovación en la construcción de nuevos programas de software.

En el año 2014 se ha desarrollado en la Facultad de Informática de la Universidad Nacional del Comahue un juez en línea denominado HORNERO para la gestión de torneos de programación, lo que ha permitido generar nuevos ambientes y situaciones de aprendizaje, que buscan potenciar la creatividad, la imaginación, el trabajo en equipo y el desarrollo del pensamiento computacional.

Se han desarrollado distintas experiencias utilizando HORNERO en el marco de tres proyectos de extensión realizados entre la Universidad y la Escuela Media, y en una instancia en el marco de los IX Juegos Olímpicos del Comahue realizados en octubre de 2015. En este trabajo se describen las experiencias y presentan algunos resultados obtenidos.

Palabras clave: Programación, HORNERO, Torneos de Programación, Aprendizaje Colaborativo, Software Libre.

Abstract

Searching for strategies to improve programming learning and teaching is an essential task in both the secondary and university level. Students have motivation problems and other types of problems as personal adaptation to the level of difficulty. In this

context the development of programming contests seems to be an excellent strategy for fostering creativity, teamwork and innovation in the construction of new software programs.

In 2014 it has been developed a judge online called HORNERO at Faculty of Informatic, Comahue University. HORNERO manages programming contests. It has enable the generation of new environments and learning situations, for enhancing creativity, imagination, teamwork and development of computational thinking.

It has been developed different experiences using HORNERO within three extension projects. The projects involved the University and Secondary School. It has also been used in the IX Olympic Games of Comahue, in October 2015. This work describes the experiences and presents some results.

Keywords: Programming, HORNERO, Programming Tournaments, Collaborative Learning, Free Software.

1. Introducción

Las Tecnologías de la Información y la Comunicación (TIC) han contribuido a motivar a jóvenes que se encuentran en plena experimentación de nuevas formas de socialización y de adquisición de conocimientos. Mediante la educación se les proporciona valores y actitudes sociales, sumado a experiencias constructivas que favorezcan la adquisición de habilidades y competencias relacionadas con la gestión del conocimiento. Es necesario que los gobiernos hagan un esfuerzo para identificar y conceptualizar correctamente el conjunto de habilidades y competencias requeridas (según estándares educativos) que cada estudiante debe ser capaz de alcanzar al final de la educación obligatoria. Esto se ve reflejado en un informe de la

OECD (Organisation for Economic Co-operation and Development) publicado en 2009 [1].

Dentro de las habilidades que se destacan se identifican las de orden superior, tales como las lógicas y el pensamiento crítico, las cuales pueden favorecerse mediante el uso de tecnología, por ejemplo a través de la enseñanza y el aprendizaje de la programación. En las Ciencias de la Computación, la programación se orienta al desarrollo de una serie de habilidades que permitan dar soluciones generales aplicables a distintos problemas del mundo real, resaltando la creatividad y la innovación. Interesa principalmente alcanzar el pensamiento algorítmico [2]. Por este motivo en los primeros cursos de carreras informáticas se toma como eje la resolución de problemas, se enseñan distintas formas de representar la información dada, como así también estrategias de resolución tales como analogía, abstracción, generalización, entre otras.

A nivel país, el proyecto Program.AR, iniciado a mediados del año 2013, tiene como propósito llevar la enseñanza y el aprendizaje de las Ciencias de la Computación a la escuela argentina. Además de la formación de docentes, en este proyecto se facilitó material didáctico que ofrece una guía a los docentes sobre los principios básicos de la programación. Los recursos utilizados, además de favorecer el aprendizaje de los conceptos básicos de programación, buscan motivar y favorecer la interacción mediante la programación de juegos [3].

En otras iniciativas, se trabaja con programación competitiva, en la bibliografía consultada se ha encontrado información respecto a distintas experiencias llevadas a cabo dentro y fuera del país [4, 5, 11].

La contribución específica de los torneos en el nivel medio, en materias donde se enseña programación, implica un cambio de la educación en informática, dirigida a estudiantes de edades comprendidas entre 16 y 18 años, donde se desarrollan tareas que favorecen la puesta en marcha de habilidades informáticas relegando los aspectos técnicos del lenguaje de programación utilizado a un papel secundario. En los cursos introductorios de Ciencias de la Computación a menudo se da el abandono y fracaso de los estudiantes, muchos se sienten frustrados, y esto deriva en un grado de frustración que afecta tanto al aprendizaje como a la enseñanza. Por otra parte, tanto en el nivel secundario como en el universitario se detectan problemas de motivación y de adecuación personal al nivel de dificultad, creemos que por este motivo los estudiantes se sienten atraídos y fuertemente motivados por competiciones, donde pueden ponerse a prueba en tareas que demanden más creación, que las que generalmente se asignan en el trabajo escolar normal.

Algunas iniciativas hacen hincapié en la resolución de problemas como una habilidad preliminar para la programación. El gran problema radica en cómo presentar este tipo de problemas de manera atractiva y motivadora, no sólo como otro tipo de ejercicios abstractos completamente desconectados de la realidad. En la iniciación a la programación se enseña a escribir programas elegantes y correctos (elección significativa de identificadores, claridad de la estructura del programa, modularización adecuada, entre otros aspectos), intentando que los estudiantes logren, ante programas defectuosos que contienen errores, establecer las causas del funcionamiento incorrecto, lo que requiere un esfuerzo de introspección por parte de los mismos. Obviamente, el objetivo no es hacer de cada estudiante un programador profesional. El lenguaje de programación se presenta como una herramienta para probar soluciones, priorizando una etapa preliminar que implica el uso de lápiz y papel, u otras herramientas, mostrando la necesidad de pensar antes de escribir, ejecutar y probar el programa diseñado y codificado debe seguir siendo el objetivo final.

Desde el año 2014 se han desarrollado diferentes iniciativas que emplean torneos de programación, tales como proyectos de extensión con escuelas que tienen en su currícula la materia programación, la Olimpiada Informática del Comahue, entre otros, donde se fomenta la interacción dentro y fuera de las escuelas, y con la Facultad de Informática (FAIF). Esto se realiza a lo largo de todo el curso escolar, con frecuentes sesiones de entrenamiento, torneos de preparación mensuales, apoyado por el entorno web HORNERO. En este trabajo se presentan resultados obtenidos, como así también consideraciones para el armado de torneos mediante la aplicación HORNERO.

El trabajo se enmarca en la investigación realizada dentro de los proyectos de investigación “Simulación y Métodos Computacionales En Ciencias y Educación”, en la línea “Desarrollo y Uso de TIC”, “Aspectos Avanzados de la Programación de Computadoras” y “Agentes inteligentes en ambientes dinámicos”, los tres pertenecientes a la Facultad de Informática, de la Universidad Nacional del Comahue (UNCo).

2. El rol de la competencia en educación

Los niños y/o jóvenes buscan espontáneamente competir contra sus pares, un deseo que pareciera innato los lleva a desear compararse con los demás. En la historia pueden descubrirse distintos tipos de competencias, donde una de las más conocidas son las deportivas, por ejemplo los Juegos Olímpicos griegos. Los ganadores en ese entonces podían obtener premios simbólicos (una guirnalda o corona) o más importantes.

La palabra torneo se utiliza como un término general para una determinada forma de organización de competiciones deportivas, extendiéndose el concepto de competencias al mundo científico. Por ejemplo, además de reuniones y publicaciones, en muchas instituciones educativas se han organizado torneos, con premios llamativos y/o tentadores, con el fin de fomentar la solución de importantes problemas matemáticos y científicos. En algunos casos no competían por el premio sino por una cuestión de honor.

A partir de un concurso de matemáticas para estudiantes universitarios norteamericanos, realizado en el año 1938, surgieron las Olimpiadas Matemáticas Internacionales (OMI), que fueron organizadas la primera vez por Rumania en el año 1959. Otras disciplinas posteriormente fueron estableciendo sus propias olimpiadas internacionales, por ejemplo la informática en el año 1989.

Teóricos de la educación no acuerdan si se debe fomentar o no la competencia en educación. Estudios mostraron que resultados logrados por los participantes en su actuación en la OMI no están correlacionados con logros posteriores en matemáticas, sin embargo eso no significa que logren un buen desempeño en su vida social posterior [5]. De la misma manera, grandes matemáticos contemporáneos no han tenido buenos resultados en competencias matemáticas. Esto sucede en las distintas disciplinas. Se encontraron diferentes posturas en cuanto al uso de competiciones en educación, algunos autores resaltan que resulta beneficioso al contemplarse en un plan de estudio porque le quita la rigidez al mismo, sobretodo si no es lo suficientemente desafiante, otros hacen referencia a que debería ser algo externo. En cuanto a su efectividad, un argumento que sobresale es el de resultar una forma efectiva de motivar a los estudiantes, permitiendo proporcionarles una retroalimentación en base a su desempeño. En algunos casos ayuda a los estudiantes a incrementar su nivel de confianza, y en otros casos, buenos estudiantes no querrán competir por una emergente necesidad de controlar su destino [4 -6].

En la realización de un torneo son necesarias tres fases principales: preparación, ejecución y seguimiento. La primera refiere a la preparación de normas de competencia, tareas, procedimientos de competencia para juzgar, etc. Las reglas o reglamento debe ser lo más completo y transparente posible, a fin de no confundir a los participantes. En la fase de ejecución es donde se lleva a cabo la competencia real, los participantes compiten y son evaluados. La última fase corresponde al análisis y presentación a los participantes de los resultados obtenidos, además de la premiación.

Existen en la actualidad diferentes olimpiadas de informática, competencias tales como la gestionada por la ACM (Association for Computing Machinery),

ACM/ICPC¹ (International College Programming Contest) que data del año 1970, jueces automáticos en línea que disponen de cientos de problemas que pueden ser usados por los docentes y estudiantes para ejercitarse y poner en práctica sus habilidades relacionadas a la programación [6]. Dado que el trabajo con estos jueces demanda conocimientos de programación en algunos casos avanzados, y manejo de estructuras de datos, los problemas a resolver pueden resultar complejos y desalentadores para aquellos estudiantes que recién se inician en la programación. Tal es el caso de los estudiantes de nivel medio que han participado de los diferentes proyectos de extensión desarrollados en la FAIF y los que cursan el primer año de las carreras informáticas ofrecidas en dicha institución. Esto ha motivado el desarrollo de HORNERO², cuyas características se presentan en un apartado más adelante. La competencia existe “cuando los participantes trabajan contra los demás para alcanzar un objetivo que sólo uno o unos pocos conseguirán”, esta definición está basada en la combinación de la interdependencia de resultado (relación objetivos perseguidos/recompensas) y de medios (acciones para lograr los objetivos). Con respecto a ésta última, hace referencia a que una tarea sólo puede terminarse con la ayuda del otro. En una competencia es necesaria la interdependencia negativa de objetivos, esto refiere a que los estudiantes que mejor se desempeñen serán reconocidos como exitosos, debe existir al menos un ganador (individuo o equipo), comparaciones sociales sobre el desempeño y criterios bien establecidos para determinar al ganador (reglamento). De esa manera se logra una competencia constructiva, divertida y motivadora, ayudando en la enseñanza de los valores del juego limpio. Si la competencia se vuelve estresante y provoca mucha ansiedad, puede ocasionar un efecto contrario al buscado, siendo una estrategia educativa destructiva para los estudiantes [7].

Asimismo, para la ejecución de la competencia y la preparación para la misma se propone aplicar la estrategia del aprendizaje cooperativo, de forma de lograr mayores avances a partir de aprovechar las habilidades interpersonales y grupales. En el contexto de la enseñanza de la programación el presentar tareas que requieren múltiples habilidades y la organización de los estudiantes con asignación de roles, enfocándose en la responsabilidad del grupo, conduce a lograr un mejor desempeño de los estudiantes y un aprendizaje productivo [8, 9].

¹ <http://icpc.baylor.edu/>

² <http://hornero.fi.uncoma.edu.ar/>

2.1.Aspectos claves en la organización de torneos

La educación y la competencia se pueden combinar de muchas maneras, desprendiéndose atributos y dimensiones que pueden ser de ayuda para clasificar los tipos de torneos a realizar, o diseñar uno nuevo. En la figura 1 pueden observarse distintos aspectos y elementos a tener en cuenta al momento de diseñar un torneo.



Figura 1. Elementos a tener en cuenta en la organización de torneos.

El empleo de metodologías que favorezcan el trabajo grupal como el cooperativo, puede ayudar a promover no sólo conocimiento del tipo conceptual sino además habilidades del tipo social, afectivo y profesional. Un grupo de estudiantes no va a cooperar adecuadamente sólo porque se les diga que trabajen en forma grupal o porque se los siente en un mismo espacio físico, se necesita “construir relaciones y crear confianza entre ellos antes de que puedan compartir ideas y ayudarse de manera eficaz para producir un verdadero esfuerzo grupal” [3].

2.2.Comunidad de Práctica: facilitando el abordaje de la enseñanza y el aprendizaje de programación

La FAIF ofrece un espacio de socialización clave para desarrollar el diálogo entre los distintos grupos de interesados en la temática programación. Se cuenta con una comunidad de práctica (CP) que ha iniciado en el año 2014, creada con el fin de abordar la enseñanza y aprendizaje de programación, para la cual se han tenido en cuenta los siete principios de diseño definidos en [9] para facilitar su gestión de una manera flexible, evitando caer en la improvisación. Está basada en comunidades preexistentes de redes personales, por

ejemplo varios de los docentes de nivel Medio con los que se ha trabajado en proyectos de extensión han sido alumnos de la carrera Profesorado en Informática, dictada desde la FAIF. Se dispone de un espacio virtual que permite brindar material bibliográfico, canales para favorecer la comunicación síncrona y/o asíncrona entre los diferentes miembros, enlaces a recursos TIC y a otras comunidades que investigan y trabajan temas de programación.

Un equipo coordinador es el responsable de ayudar y agilizar el desarrollo de reuniones, favoreciendo el trabajo en un clima y ambiente sano, favoreciendo un diálogo abierto y fluido. Se cuenta con espacios públicos y privados, de acuerdo a la temática de trabajo e intereses de los participantes. Dada que la participación en la mayoría de las comunidades es de carácter voluntario, se persigue contemplar los problemas y las necesidades de los miembros actuales de la comunidad, los cuales pueden cubrirse a través del desarrollo de conferencias, talleres y torneos.

La vida de una comunidad depende de la red de relaciones duraderas entre los miembros, el ritmo de sus interacciones fuertemente influenciado por el ritmo de los acontecimientos de la comunidad, por ejemplo reuniones presenciales, teleconferencias, la actividad en HORNERO, etc. El ritmo que adquiera, indicador de su vitalidad, dependerá además de la conexión lograda por sus miembros, del sentido de colaboración y trabajo colectivo.

3. HORNERO: aplicación Multi-Paradigma, Multi-Plataforma, Multi-Lenguaje para la gestión de torneos de programación

Existen en la actualidad varios jueces online que permiten la generación de torneos, tales como Uva online judge³ (España), Sphere online judge⁴, TopCoder⁵, Code forces⁶, Caribbean Online Judge⁷ (Cuba), OmegaUp⁸ (México) entre otros, muchos de ellos están limitados a unos pocos lenguajes y requieren conocimientos avanzados de programación.

Esto motivó el desarrollo de la herramienta denominada “HORNERO”, diseñada teniendo como meta dos premisas, competir empleando cualquier lenguaje de programación y dar la posibilidad de jugar a todas las

³ <http://uva.onlinejudge.org>

⁴ <http://www.spoj.com>

⁵ <http://www.topcoder.com>

⁶ <http://www.codeforces.com>

⁷ <http://coi.uci.cu/>

⁸ <http://omegaup.com>

personas interesadas. La figura 2 muestra la pantalla inicial.



Figura 2. Pantalla Inicial de la aplicación HORNERO.

En el ámbito educativo esto significa poder llegar a todos los estudiantes y programadores que quieran jugar, y a todos los docentes que quieran utilizar la herramienta sin importar el lenguaje que se esté utilizando de acuerdo a la planificación curricular.

La herramienta HORNERO se desarrolló en el lenguaje PHP, con base de datos Mysql y servidor de Web Apache. Para la misma se desarrolló una API de comunicación vía web service REST[10] lo que permite la interacción con lenguajes con interfaz de socket TCP. Con el fin de no excluir a los lenguajes que no tienen interfaz de socket TCP, se desarrolló un wrapper en C que encapsula la comunicación con el servidor y llama al archivo desarrollado por el programador enviando y recibiendo los parámetros por entrada y salida estándar, logrando de esta manera extender la comunicación a todos los lenguajes de programación.

En la actualidad HORNERO permite trabajar con los lenguajes: java, python, php, c, c++, pascal, javascript, c#, ciao-prolog, perl, bash, lisp y ruby. Para facilitar el desarrollo de la experiencia en cursos iniciales, se ha incorporado además la posibilidad de trabajar con pseudocódigo, a partir del uso de la herramienta PSeInt⁹. Se sigue trabajando para adicionar nuevos lenguajes, como es el caso de Visual Basic y Smalltalk.

Una tecnología por sí sola no asegura el éxito en su aplicación, incide de manera diferente en los estudiantes, y factores como la calidad de conexión, el ancho de banda y la velocidad pueden afectar el ánimo y desempeño de los mismos. Se debe tener en claro los objetivos perseguidos al incorporar un recurso al aula y sus implicancias en el proceso de enseñanza-aprendizaje de la programación. Por esta razón en el momento que se utilice HORNERO es importante conocer respecto a los estudiantes las competencias o capacidades, y respecto al uso académico de las nuevas tecnologías, las

actitudes que se adoptan en relación a su uso en el proceso de su formación, la valoración que realizan como herramientas de apoyo para el proceso de aprendizaje y sus necesidades formativas.

3.1. Configuración y dinámica de los torneos

El objetivo principal de un torneo consiste en resolver correctamente todos los problemas que se presentan, dentro del plazo de tiempo máximo establecido y empleando para ello el menor tiempo posible.

Los participantes (estudiantes, programadores o docentes, según sea el caso) se agrupan en equipos, entre todos eligen un nombre y diseñan el logo que los identificará, esto último es opcional. Esto debe realizarse en forma previa al inicio del torneo, y nuestra sugerencia es conformar grupos de no más de 4 integrantes de manera tal de asegurar la interacción y colaboración de cada uno de ellos. Se les presenta la lista de roles a trabajar dentro del equipo (lector de enunciado, diseñador, codificador y verificador de solución), y se les pide a cada uno de ellos que elija el rol inicial. Se sugiere que estos roles vayan rotando ante cada nuevo ejercicio a resolver.

Cada equipo tendrá acceso a una computadora personal equipada con el software, lenguaje o entorno de programación elegido para participar. Los participantes pueden comunicarse sólo con integrantes de su equipo y con el personal encargado del torneo, pudiendo además hacer uso de tutoriales o manuales de referencia para el trabajo con la aplicación HORNERO y del lenguaje elegido para participar.

Dado que se espera un ambiente de competencia sano, a cada equipo se le entrega un reglamento del torneo, se les pide su lectura y se aclaran las dudas que puedan presentarse. El equipo que desacate estas reglas será sancionado, pudiendo en el peor de los casos quedar descalificado.

Cada equipo debe registrarse en la aplicación Web, y a continuación inscribirse en el torneo que desee, entre los que estén habilitados. A partir de la hora de inicio del torneo los participantes inscriptos podrán visualizar los enunciados de los ejercicios que lo conforman.

Por defecto se presentan 10 problemas por torneo, aunque esa cantidad puede modificarse en el momento en que se crea y configura el torneo. Hasta el momento se cuenta con 77 ejercicios para seleccionar que han sido utilizados en el marco de diferentes experiencias, aunque la base de datos de problemas se mantiene en constante expansión.

Previo al torneo, se debe descargar desde la aplicación el Stub del lenguaje de preferencia, el cual debe ser modificado en función de los problemas a resolver. El creador de un torneo decide el orden en que se presentan los ejercicios, el cual puede ser por complejidad,

⁹ <http://pseint.sourceforge.net/>

aleatorio, o respetando alguna otra característica, por ejemplo por estructura de control involucrada. Independientemente de ello los ejercicios pueden resolverse en cualquier orden. La resolución realizada se ejecuta en la máquina del programador, recibiendo los parámetros y enviando la respuesta desde y hasta el servidor. Cada equipo se entera al instante si la solución enviada es correcta, de manera que puede corregirla y reenviarla tantas veces como sea necesario. Durante el desarrollo de la competencia se visualiza el avance del tiempo y una tabla de resultados le permitirá conocer el número de problemas resueltos correctamente, el puntaje y el tiempo empleado en la resolución. Esto permitirá observar el desempeño de los equipos participantes, resultando ganador el equipo que resuelve más ejercicios en el menor tiempo.

Algunos de los logros más relevantes con HORNERO son la abstracción del ejercicio-resultado del lenguaje, la reutilización de ejercicios (por ej. entre diferentes cátedras y lenguajes) y la generación distribuida de ejercicios por retroalimentación de la misma comunidad de docentes.

Las características consideradas relevantes y que se han tenido en cuenta hacen referencia a: buenas prácticas de diseño de ejercicios, establecimiento y cumplimiento de pautas a contemplar en las etapas de diseño de algoritmos y codificación, clasificación de ejercicios según complejidad y tema a desarrollar, reglamento simple.

3.2. Escritura de problemas

Se buscó conformar un conjunto de problemas con la más amplia cobertura de áreas de conocimiento, con una distribución justa de los niveles de complejidad para promover la participación y el buen desempeño de nuevos participantes. Se utilizan etiquetas, sobretodo para una correcta clasificación de problemas que abordan diferentes temas al mismo tiempo. Al agregar un nuevo problema a la base de problemas existentes se tiene en cuenta:

- *evitar el uso de grandes conjuntos de datos:* esto es importante dado que puede aumentar considerablemente la diferencia de tiempo de ejecución entre algunos lenguajes de programación
- *establecer formato de entradas válidas y los límites para todos los parámetros*
- *analizar mecanismos de entrada y salida de datos:* además de esto se analizan los necesarios para la validación de los resultados enviados.

Nuestra base de problemas se ha conformado con problemas recopilados y adaptados de diversas fuentes:

materias de primer año que abordan temas de programación en carreras dictadas en la FAIF, bases de datos de problemas que se utilizan en sitios que promueven torneos de programación tales como Caribbean Online Judge, ACM, OmegaUP y del sitio del proyecto Euler¹⁰.

Los niveles de dificultad especificados hasta el momento son: Bajo, Medio y Alto, y están relacionados con el uso de las distintas estructuras de control. Por ejemplo en el nivel Bajo se encuentran problemas que se resuelven a través del uso de secuencias simples. En el nivel Medio se encuentran problemas que en su resolución utilizan secuencias combinadas con estructuras alternativas, o problemas que pueden resolverse con una sola estructura repetitiva. En el nivel Alto se encuentran problemas de mayor complejidad, que en su resolución requieren la combinación de las diferentes estructuras de control. Se contempla la adición de nuevos niveles de dificultad para asegurar una mayor cobertura. Debido a las falencias que presentan los estudiantes en la interpretación de consignas se ha prestado especial atención a la redacción de los problemas, complementando el texto de la actividad con el uso de imágenes y ejemplos.

4. Desarrollo de Torneos de Programación en el Nivel Medio: algunos resultados

Desde la Facultad de Informática se han desarrollado diferentes iniciativas cuyo propósito principal fue el de promover la innovación en educación optimizando prácticas pedagógicas entre niveles y constituyendo de esta manera una base sólida para fomentar el desarrollo de instancias de trabajo colectivas, espacios para la discusión y el análisis.

Teniendo como premisa acercar a los estudiantes al pensamiento computacional mediante la enseñanza de la programación, se ha trabajado en la planificación y desarrollo de nuevas experiencias de aprendizaje. Desde la FAIF, uno de los objetivos perseguidos es la vinculación de la educación universitaria con el sistema de educación del nivel medio, atendiendo a favorecer una mayor articulación institucional y pedagógica. Se busca promover la formación de una comunidad de programadores en la región a partir de la realización de torneos análogos, contemplando diferentes niveles de exigencia, acordes a quienes se inician en las tareas de programación.

Hasta la actualidad se han desarrollado dos proyectos de extensión, y un tercero en curso, en los cuales participan docentes del Departamento de Programación, alumnos avanzados y personal no docente de la facultad

¹⁰ <https://projecteuler.net/>

mencionada. Además los docentes responsables de cada escuela de enseñanza media considerada, y los grupos de estudiantes. Proyectos de extensión finalizados y en desarrollo:

- “Entrenamiento en la programación de la computadora a partir de una aplicación para competencias”, período 2014-2015. Participaron estudiantes y docentes de las instituciones de nivel medio Colegio AMEN (Neuquén), CEM 14 (Fernández Oro) y Colegio Almafuerde (Cipolletti). 60 estudiantes, en promedio 20 estudiantes por colegio
- “Olimpiadas de programación para Escuela Media”, período 2015-2016. Participaron estudiantes y docentes de las instituciones de nivel medio Colegio AMEN (Neuquén), CEM 14 (Fernández Oro) y Colegio Almafuerde (Cipolletti), Epet 20 (Neuquen). 80 estudiantes, en promedio 20 estudiantes por colegio
- “La Competencia como Eje Motivador para Acercarnos a la Programación”. Está en desarrollo actualmente, y tiene la particularidad de que en esta nueva etapa, además de las instituciones participantes en los proyectos desarrollados anteriormente, interviene un colegio de enseñanza media de El Bolsón y se prevé comenzar a incorporar otros colegios del interior de las provincias de Río Negro y Neuquén.

4.1. Metodología de trabajo

Previamente al comienzo de cada proyecto se realizan talleres con los docentes de las escuelas del nivel medio participantes con el fin de reforzar y unificar conocimientos de programación y presentar los recursos TIC a utilizar (aula virtual, herramienta PSeInt, HORNERO), facilitando de esta manera su actuación como entrenadores en las competencias de programación planificadas. Para fomentar la integración y colaboración de los docentes y estudiantes de todas las instituciones participantes se realizan diversos encuentros combinando las modalidades presencial y virtual. Mediante un cronograma se pautan día, horario y lugar de realización de los torneos. Algunos torneos se realizan en dependencias de la FAIF, donde los estudiantes y docentes concurren por sus propios medios (concurren con sus netbooks), y en otros se hace desde cada institución mediante el uso de HORNERO. En los casos de aquellas instituciones donde no se cuenta con buena conexión a Internet, los estudiantes utilizan uno de los laboratorios de la FAIF. En la figura 3 se observan los estudiantes trabajando

colaborativamente en uno de los torneos realizados, en dependencias de la FAIF.



Figura 3. Estudiantes en plena competencia.

Las actividades que se proponen inicialmente se focalizan en elementos que puedan resultar atractivos para los estudiantes, por ejemplo juegos y aplicaciones multimedia interactivas, con las cuales se los introduce en la resolución de problemas. A fin de nivelar los conocimientos de los estudiantes se preparó un aula virtual con material bibliográfico, propuestas de ejercicios, acceso a torneos de pruebas y canales de comunicación para que el equipo organizador pudiera evacuar dudas acerca de los contenidos abarcados, complejidad y cuestiones organizativas.

La modalidad de participación aplicada en los torneos es grupal, por lo que en algunos torneos los grupos se conformaron con integrantes de cada institución. Uno de los problemas presentados fue que el tamaño de grupo estuvo en función de la cantidad de netbooks disponibles. Si bien un tamaño de grupo grande puede significar una gama más amplia de aptitudes, habilidades, destrezas y puntos de vista, la interacción y cohesión entre los miembros es menor, posibilitando que algunos estudiantes se oculten y por ende no realicen aportes. Esto motivó a prestar atención en la cantidad de netbooks disponibles, como para asegurar grupos de no más de 6 integrantes. Para lograr una interdependencia positiva se tuvo en cuenta:

- Proporcionar a cada equipo una única copia del material, con lo cual se verían obligados a trabajar juntos.
- Incentivar a cada grupo a establecer su identidad conjunta mediante un nombre, un logo o un lema.
- Asignación de roles: lector de enunciado, diseñador, codificador y verificador de solución. Dependiendo de la cantidad de integrantes algunos de estos roles se repetirán, se deberá rotar el rol al comenzar la resolución de un nuevo ejercicio. De esta manera es posible realizar una distribución de tareas donde cada integrante debe realizar su parte

para avanzar en la búsqueda de una solución al problema dado.

- Establecimiento de recompensas: el grupo festeja el éxito y recibe una recompensa conjunta. En la primera experiencia se les regaló una remera con un logo y mensaje alusivo a la competencia, en la segunda experiencia se entregó un cuello polar con el logo del torneo.

Trabajar para incrementar la interdependencia positiva en un grupo de aprendizaje cooperativo, posibilitará a los estudiantes sentir más responsabilidad en realizar sus aportes en pro de lograr un resultado conjunto, y evitando conductas inadecuadas. Factores intrínsecos (objetivos individuales) y extrínsecos (ambientales, entorno de aprendizaje) pueden afectar la motivación y el aprendizaje de los estudiantes en cursos de programación, por esta razón se ha prestado atención a los referentes a la “recompensa y reconocimiento” y a la “presión y competencia social” [11].

Como toda tecnología a ser utilizada en forma online, para las experiencias educativas a realizar con HORNERO se tuvo en cuenta: el lugar de acceso (escuelas de nivel medio y aulas de la universidad), la frecuencia y el tiempo de la conexión, competencias y habilidades de los estudiantes respecto al empleo de tecnología usada en el proyecto (manejo de netbook, Internet, PSeInt, HORNERO), la diferencia entre las habilidades percibidas y las habilidades reales que poseen los estudiantes para diseñar e implementar algoritmos y la actitud de estudiantes y docentes de ambos niveles educativos respecto a la tecnología a utilizar.

En las primeras actividades se contempla una competencia donde un jurado de expertos es el encargado de evaluar el diseño y funcionamiento del algoritmo utilizando un lote de prueba que acompaña a cada problema presentado. Posteriormente esto se realiza en forma automática por la aplicación HORNERO.

Se pauteó el uso de pseudocódigo dado que permite al alumno centrarse en la resolución del problema evitando la rigidez que impone generalmente la sintaxis del lenguaje de programación.

4.2. Resultados y autoevaluación del proceso

Los docentes del nivel medio con los que se ha trabajado presentan una muy buena formación docente, que pueden ser facilitadores en dicho campo, contando con todas las competencias requeridas para cubrir las expectativas del proyecto. Las mayores dificultades y limitaciones presentadas a lo largo de este proceso han

tenido relación con problemas de conectividad y baja relación de computadoras por grupo de estudiantes.

Se ha tomado la competencia como una instancia más de aprendizaje, por lo que se trató que la puntuación (alta/baja) y el resultado (ganador/vencedor) obtenido se tomase como una cuestión secundaria. Previo a los torneos se explicaron las reglas, procedimientos y conductas esperadas, esto favoreció que la competencia en todas las instancias fuera constructiva (contexto cooperativo).

Se han realizado seguimientos de la participación de los estudiantes y su desempeño en los torneos, mediante registros objetivos por estudiante, lo que ha permitido reflejar cómo ha sido la evolución de los mismos, permitiendo realizar valoraciones cuantitativas y cualitativas. Una instancia de evaluación final ha permitido analizar el rendimiento de los alumnos participantes con el fin de determinar si la utilización de la aplicación HORNERO ha realizado un aporte significativo a la enseñanza de programación.

Los resultados obtenidos en los distintos proyectos llevados a cabo han posibilitado: la extensión de la propuesta al currículo, la adecuación y construcción de nuevos y mejores espacios, la adquisición de materiales y recursos, el establecimiento de alianzas, el abordaje de la línea de enseñanza de programación, acciones que no se hubieran podido lograr sin este respaldo y que han fortalecido amplia y profundamente la propuesta.

5. Olimpiadas de Programación Uncoma: extendiendo los límites de la FAIF

En el mes de octubre de 2015 en la Sede Central de la Universidad Nacional del Comahue, de Neuquén Capital, se desarrollaron los "IX Juegos Olímpicos del Comahue". Esta es una actividad organizada por la Secretaría de Extensión Universitaria en el marco de las políticas de extensión, formación, vinculación y articulación entre la Universidad Pública y la enseñanza media. La convocatoria estuvo dirigida a todos los establecimientos educativos de Nivel Medio de las Provincias de Neuquén, Río Negro y San Salvador de Jujuy. Participaron un total de 132 establecimientos. En estos juegos se propone la participación de los estudiantes en disciplinas y temáticas tales como Letras, Física, Historia, Matemática, entre otros. La disciplina Informática fue adicionada por primera vez en dicha oportunidad.

En este tipo de evento al igual que en los proyectos de extensión desarrollados, es muy importante la participación de los estudiantes de la escuela media en la vida universitaria, dado que es paso siguiente en su vida educativa. Se instó a los alumnos a dar el máximo de su esfuerzo durante la competencia como parte del

compromiso que han asumido como grupo, valorando positivamente la respuesta de todos los profesores de escuelas secundarias quienes han trabajado con sus alumnos en el periodo de preparación.

La participación se dividió en tres categorías según la dificultad de los problemas presentados:

- **Categoría 1:** secuencia y estructuras alternativas
- **Categoría 2:** estructuras repetitivas
- **Categoría 3:** arreglos y cadenas

Una reunión pautada previamente con los docentes de los establecimientos, encargados de la disciplina “Informática” permitió acordar la forma de trabajo y preparación para la competencia. Se explicó cómo trabajar con la herramienta de competencias HORNERO. Se habilitó un espacio virtual en la plataforma PEDCO¹¹ para la preparación de los grupos de estudiantes. En el mismo se dejó disponible material de estudio que constó de una guía teórica de resolución de problemas, diseño de algoritmos (en pseudocódigo utilizando la herramienta PSeInt) y programación; y una lista de problemas similares a los seleccionadas para trabajar en la olimpiada. Se habilitaron además canales de comunicación para la evacuación de dudas. Para la competencia se utilizó la herramienta HORNERO, como juez en línea encargado de evaluar los trabajos de los grupos. Cada grupo ejecuta su solución a un problema, y la herramienta verifica si la respuesta presentada es correcta.

La segunda instancia constó de 2 fases: La primera fase fue de carácter preparatorio, llevada a cabo el primer día de los juegos olímpicos. La segunda fase de carácter competitivo se desarrolló el segundo día de los juegos. Los grupos participantes estuvieron conformados por 3 o 4 estudiantes. Cada equipo contaba con una computadora personal equipada con el software, lenguaje o entorno de programación elegido para participar. Durante las competencias se proyectó en una pared visible a todos los equipos, la cuenta regresiva al fin del torneo y la tabla de posiciones que se fue actualizando a medida que los equipos fueron resolviendo los problemas. Ver pasar los segundos y la posición que cada equipo ocupa crea un efecto motivador visible en eufóricos festejos al resolver cada ejercicio.

Al finalizar las competencias se realizó un acto de cierre en conjunto con el resto de las disciplinas donde hubo entrega de diplomas y trofeos. En la figura 4 se observan algunos de los resultados obtenidos por los participantes en cada una de las categorías. Se puede observar la localidad de la que proviene la institución, el tiempo promedio logrado por cada grupo, los tiempos

mínimos y máximos empleados en resolver los ejercicios planteados. Si bien son pocos datos que no permiten un análisis más exhaustivo, si se toma el tiempo promedio de cada grupo, se observa en cada categoría grupos de estudiantes que se destacan del resto. Esto es más evidente por ejemplo en la tercera categoría donde el grupo de la localidad El Bolsón obtuvo un tiempo menor al obtenido por los dos grupos de la ciudad de Neuquén.

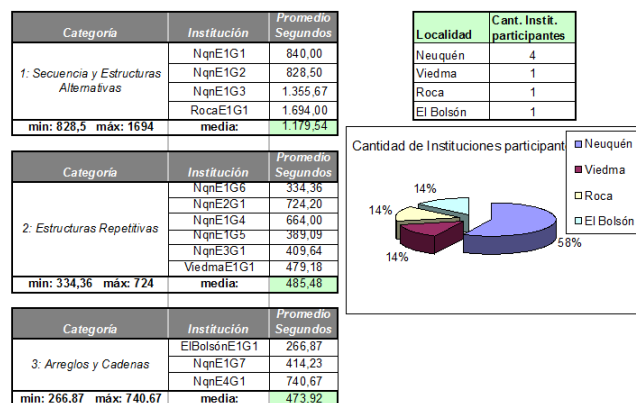


Figura 4. Estadísticas competencias de Informática. IX Juegos Olímpicos del Comahue

Corresponde destacar que el grupo participante de General Roca, proveniente de un colegio sin modalidad Informática recibió una mención especial por el esfuerzo realizado por los estudiantes y su profesor para poder participar. Asimismo la participación de un grupo de estudiantes de El Bolsón fue la motivación necesaria para comenzar a involucrar al colegio que representan en los proyectos de extensión universitaria relacionados con competencias de programación.

6. Aprendizaje competitivo en carreras de la FAIF

Nuestra búsqueda permanente de estrategias que nos permitan mejorar la enseñanza y aprendizaje de programación de estudiantes que cursan los primeros años de carreras informáticas universitarias, nos ha permitido encontrar en el desarrollo de Torneos de Programación una forma de fomentar la creatividad, el trabajo en equipo y la innovación en la construcción de nuevos programas de software. Además de lo mencionado les permite a los estudiantes al mismo tiempo probar su habilidad para trabajar bajo presión, algo que recién ponen en práctica aquellos que deciden participar del ACM, y que por lo general es un número pequeño de estudiantes.

Esto nos motivó en 2015 al armado de un taller para la preparación de los alumnos en una labor extra clase, comúnmente en un horario de la semana establecido

¹¹ <http://pedco.uncoma.edu.ar/>

para no perjudicar las clases normales, con un equipo de trabajo formado por docentes del departamento de Programación. El proceso incluye varias actividades como la capacitación en una aplicación que permite el diseño y ejecución de algoritmos utilizando la metodología pseudocódigo, y la generación de concursos internos periódicos utilizando el juez en línea Hornero.

Se realizaron experiencias desde el año 2014 en las materias Resolución de Problemas y Algoritmos (recursado), y Desarrollo de Algoritmos, dictadas respectivamente en el primer y segundo cuatrimestre del primer año de las carreras Analista en Computación, Licenciatura en Ciencias de la Computación y Profesorado en Informática. Los problemas trabajados incluyeron temas tales como estructuras básicas de control, manejo de cadenas, arreglos unidimensionales y bidimensionales y recursividad.

También se realizaron experiencias con estudiantes de la carrera Tecnicatura Universitaria en Desarrollo Web, en las materias Introducción a la Programación, Programación Web Dinámica, Framework e Interoperabilidad. En las dos primeras materias se trabajó con PHP y en la última, además de PHP se implementaron soluciones utilizando los lenguajes JavaScript, Python y Java.

7. Conclusiones

La integración de las TIC en la enseñanza de programación, como todo proceso innovador, implica una revisión de la acción docente y de las estrategias pedagógicas y didácticas puestas en juego. Se requiere de nuevas destrezas y competencias, no sólo de tipo técnico, sino fundamentalmente didácticas que permitan el aprovechamiento de la potencialidad de la tecnología en el marco de la práctica docente.

Es importante recalcar que los evaluadores automáticos representan una oportunidad enorme en las materias de programación ya que actualmente no son muy utilizados y mucho menos dentro del aula. Con HORNERO ha emergido un nuevo escenario didáctico, que brinda oportunidades para mejorar la forma de trabajo en un contexto que favorece la enseñanza y aprendizaje de programación, y en el cual el trabajo en equipo se lleva a cabo de manera constructiva, divertida y motivadora. Permite la gestión de competencias de programación, utilizando pseudocódigo y una amplia variedad de lenguajes, con requisitos mínimos de conectividad, abierto a todas las personas interesadas, que sepan programar, independientemente de su condición y nivel de estudio.

Tanto en los torneos organizados en el marco de los diferentes proyectos, como en la Olimpiada de Informática, lo importante es resolver los problemas presentados de una manera rápida, lo que a veces se

traduce en escribir programas que no contemplan las consideraciones vistas como buena programación (sin estilo, ineficientes, uso inadecuado de estructuras). Esto motivó a buscar la forma en que los estudiantes interesados, pudieran además enviar los programas realizados (correctos o no) al equipo organizador para obtener una devolución más detallada y que sirviera realmente como retroalimentación. La información relevada ha sido utilizada para realizar mejoras en la aplicación y diseñar nuevas funcionalidades.

Los resultados alentadores respecto al aprendizaje de programación, obtenidos en los estudiantes participantes de los distintos proyectos han propiciado la continuidad del proyecto, extendiendo la posibilidad de participar a instituciones alejadas del radio de la UNCo. También dieron origen a la inclusión del trabajo con HORNERO en las olimpiadas. Algunos de los resultados obtenidos han sido difundidos en diferentes actividades de divulgación y promoción.

Una forma de satisfacer el interés natural en Ciencias de la Computación puede ser organizando competencias para aquellos que han aprendido los conceptos básicos a través de auto-estudio, en materias del secundario, etc. En ese caso, la competencia sirve además como vehículo de difusión, ya que puede ayudar a los estudiantes a decidir sobre la elección de una carrera orientada a estas ciencias. Esto se ha podido resolver satisfactoriamente desde HORNERO.

La universidad tiene un fuerte compromiso con el medio, los distintos proyectos de extensión ofrecidos desde la FAIF pretenden dar un apoyo continuo a las instituciones de nivel medio. En este sentido, competencias de programación si bien no son ciertamente los únicos medios para estimular el interés por la informática, pueden ayudar a promover una mejora de la enseñanza de la informática en los distintos niveles educativos.

La participación de los docentes es esencial para la revisión de programas de estudio y la forma en que sus contenidos se aprenden. Los torneos representan una oportunidad única para que los docentes y sus alumnos aprendan juntos, trabajando en tareas concretas de interés común. Nuestro desafío continúa siendo cómo explotar los concursos de programación para el cambio necesario en los sistemas educativos, de todos los niveles educativos.

Referencias

- [1] K. Ananiadou, M. Claro, "21st Century Skills and Competences for New Millennium Learners in OECD Countries", OECD Education Working Papers, No. 41, OECD Publishing, Paris. (2009). DOI: <http://dx.doi.org/10.1787/218525261154>
- [2] E. Milková. "Algorithms: The Base of Programming Skills". 29th International Conference on Information Technology Interfaces, ITI 2007. Croacia. (2007).
- [3] P. M. Factorovich, F. A. Sawady O'Connor. Actividades para aprender a Programar: segundo ciclo de la educación primaria y primero de la secundaria. 1a ed. - Ciudad Autónoma de Buenos Aires: Fundación Sadosky. (2015). E-Book. - (Programar) ISBN 978-987-27416-1-7
- [4] G. Audrito, G. Barbara Demo, E. Giovannetti. "The Role of Contests in Changing Informatics Education: A Local View." Olympiads in Informatics 6 (2012), pp.: 3-20.
- [5] Verhoeff T. 20 Years of IOI Competition Tasks. Olympiads in Informatics. (2009). Vol 3, pp. 149-66.
- [6] L.G. Torres, D. Morales Orozco, F. J. Martínez López. "Los concursos de programación como detonante del aprendizaje." ANFEI Digital 2 (2016).
- [7] D.W. Johnson Y R.T. Jonson. Aprender Juntos Y Solos. Aprendizaje Cooperativo, Competitivo E Individualista.. Ed. Aique. (1999).
- [8] L. Beck, A. Chizhik. "Applying Cooperative Learning Methods in Teaching Computer Programming". 36 ASEE/IEEE, Frontiers in Education Conference. (2006)
- [9] C. Dugas. "No Computers? No Problem! Active and Cooperative Learning in an Introductory Computer Science Course". 38 ASEE/IEEE Frontiers in Education Conference. (2008).
- [9] R.M. Wenger, And M. William. Cultivating Communities Of Practice: A Guide To Managing Knowledge - Seven Principles For Cultivating Communities Of Practice. Snyder. [Http://Hbswk.Hbs.Edu/Archive/2855.Html](http://Hbswk.Hbs.Edu/Archive/2855.Html)
- [10] R. T. Fielding, "Architectural Styles And The Design Of Network-Based Software Architectures" UNIVERSITY OF CALIFORNIA, IRVINE. (2000). ([Http://Www.Ics.Uci.Edu/~Fielding/Pubs/Dissertation/Top.Htm](http://Www.Ics.Uci.Edu/~Fielding/Pubs/Dissertation/Top.Htm))
- [11] C.C. Fracchia, P. Kogan, A. Alonso De Armiño, I. Godoy, L. López. "Realización De Torneos De Programación Como Estrategia Para La Enseñanza Y El Aprendizaje De Programación". XX Congreso Argentino De Ciencias De La Computación. Buenos Aires, Argentina. (2014), pp. 42-51. ISBN 978-987-3806-05-6. [Http://Hdl.Handle.Net/10915/42125](http://Hdl.Handle.Net/10915/42125)

Dirección de Contacto del Autor/es:

Claudia Carina Fracchia
Buenos Aires 1400
Neuquén (8300)
Argentina

e-mail: carina.fracchia@fai.uncoma.edu.ar
sitio web: <http://faiweb.uncoma.edu.ar>

Pablo Kogan
Buenos Aires 1400
Neuquén (8300)
Argentina

e-mail: pablo.kogan@fai.uncoma.edu.ar
sitio web: <http://faiweb.uncoma.edu.ar>

Silvia Amaro
Buenos Aires 1400
Neuquén (8300)
Argentina

e-mail: silvia.amaro@fai.uncoma.edu.ar
sitio web: <http://faiweb.uncoma.edu.ar>

Claudia Carina Fracchia

Docente de la Facultad de Informática de la Universidad Nacional del Comahue. Lic. en Ciencias de la Computación (UNS). Magíster en Educación en Ciencias con orientación matemática.(UNCO)

Pablo Kogan

Docente de la Facultad de Informática de la Universidad Nacional del Comahue. Lic. en Ciencias de la Computación (UNS). Cursando la Maestría en Ciencia, Tecnología e Innovación en la Universidad Nacional de Río Negro.

Silvia Amaro

Profesora Asociada del Dpto. Programación de la Facultad de Informática, UNCo. Participa de proyectos de investigación y extensión. Actualmente se encuentra realizando su postgrado en la Universidad Nacional del Sur.
