

DESARROLLO E IMPLEMENTACIÓN DE UN MARCO DE TRABAJO PARA EL ENTRENAMIENTO EN PROGRAMACIÓN COMPETITIVA

Gerson Yesid Lázaro Carrillo, Angie Melissa Delgado León, Fredy Humberto Vera Rivera
Universidad Francisco de Paula Santander, Colombia

Recibido (21/03/16), aceptado (21/06/16).

Resumen: El rol de las maratones de programación en la formación académica de los estudiantes de ingeniería es claramente reflejado en un mejor desenvolvimiento durante la vida laboral. Para fortalecer el proceso de preparación de los equipos de estudiantes en las distintas competencias de programación, con énfasis en la Maratón Regional Latinoamericana y la Maratón Mundial de Programación, proponemos un marco de trabajo basado en Tecnologías de Información y Comunicación (TIC), donde se plantean y describen las actividades a desarrollar, las herramientas a utilizar, la metodología y los cronogramas. Así mismo, desarrollamos una serie de recursos que forman parte del marco del trabajo: un sitio web, un repositorio con explicaciones teórico-prácticas, ejercicios propuestos y resueltos, así como manuales de ayuda en C++ y JAVA. El marco de trabajo propuesto ha mejorado significativamente los resultados de la participación de los equipos de la Universidad Francisco de Paula Santander – Cúcuta (Colombia) en las competencias a nivel local, regional e internacional.

Palabras Clave: Maratones de Programación, Programación Competitiva, Marco de trabajo.

DEVELOPMENT AND IMPLEMENTATION OF A FRAMEWORK FOR TRAINING IN COMPETITIVE PROGRAMMING

Abstract: The role of programming marathons in engineering student's academic training is clearly reflected in a better development during working life. In order to strengthen the preparation process for the different programming contests for which students of Francisco de Paula Santander University take part, we developed a framework to describe and planify the activities, the tools, the methodology and the schedule that guides the process. In addition, we generated a set of resources for contribute to the framework: a repository with theoretical-practical explanations, proposed and solved exercises, as well as, help manuals in C++ and JAVA with a list of different algorithms to be used during competitions. With the application of the framework proposed, we improved the results of the teams that represent the university in different local, regional and international level programming contests.

Key words: Programming Contest, Competitive Programming.

I. INTRODUCCION

Con el fin de brindar una educación integral y formar a los estudiantes en las habilidades que requieren en su vida laboral, muchas universidades incentivan la participación en actividades extracurriculares que complementen su formación académica. En carreras afines a las ciencias de la computación y el desarrollo de software, es común la participación en diferentes tipos de competencias o maratones de programación, espacios donde los estudiantes se enfrentan a un set de problemas con unas reglas preestablecidas, desarrollando soluciones algorítmicas que resuelvan dichos problemas, pudiendo interpretarse como un campo de juego entretenido para la exploración de la

inteligencia de los programadores [1]. Las maratones de programación fortalecen en los estudiantes su capacidad de razonamiento lógico, creación y análisis de algoritmos y uso de estructuras de datos, fomentando la creatividad, el trabajo en equipo, la innovación y la creación de nuevos programas de software poniendo a prueba la habilidad de los estudiantes para actuar bajo presión, investigar por su propia cuenta temas no cubiertos en clase, y resolver problemas sin la guía de un instructor [2].

Como afirma Halim [3] las universidades como instituciones de educación superior e investigación tienen un papel primordial en la consolidación de desarrolladores de software hábiles e informáticos para las generaciones futuras. Por este motivo, desde el

primer semestre del 2015, la Universidad Francisco de Paula Santander (UFPS) cuenta con el grupo de estudio en programación competitiva, adscrito al semillero de investigación en Linux y Desarrollo de Software libre (SILUX), el cual se encarga de realizar un proceso de preparación a los estudiantes que participan en competencias de programación. Este proyecto se planteó con el fin de realizar una metodología de trabajo para este grupo de estudio, la cual puede ser replicada por cualquier Universidad, que permita mejorar los resultados en las competencias oficiales donde participen estudiantes del programa. Competencias de este tipo impulsan el aprendizaje, ya que está en la naturaleza del ser humano intentar ser el mejor, sobre todo si su calificación se pone en juego. Del mismo modo, estas actividades generan entusiasmo hacia la computación y aportan reconocimiento positivo para las instituciones que las acogen [4].

El grupo de estudio de programación competitiva busca reforzar los conocimientos de sus integrantes con las temáticas típicas de una maratón de programación: algoritmos de grafos, Backtracking, procesamiento de Strings, geometría computacional, programación dinámica y conceptos básicos de teoría de números, entre otros. La construcción de un marco de trabajo para el grupo de estudio es necesaria para formalizar el proceso de entrenamiento a través de la aplicación y ajuste de técnicas y metodologías que se adapten al nivel en el que se encuentran los estudiantes. Se busca además crear un sitio web donde se comparte la información del marco de trabajo, se crea también un banco documental y de material que sirva de base para el trabajo de los diferentes equipos que participan en maratones de programación, también se desarrolla una plataforma de entrenamiento que permite fortalecer las actividades y llevar un seguimiento del trabajo realizado.

Muchas Universidades a lo largo del mundo han creado sus propias metodologías, programas, estrategias y técnicas para preparar a sus equipos en maratones de programación. Algunos de los ejemplos más notorios están en la Universidad de Stanford con su grupo de estudio SLPC (“Stanford Local Programan Contest”) donde se realizan regularmente maratones de preparación para sus estudiantes que pueden ser accedidas también por cualquier agente externo y la Universidad Nacional de Singapur, donde dirigidos por Steven Halim han desarrollado una metodología que gira en torno al libro *Competitive Programming 3* y la herramienta Uva Hunting para preparar adecuadamente a sus equipos colocando como objetivo primordial de

la programación competitiva lo siguiente: "Dado un conjunto de problemas conocidos de ciencias de la computación, resolverlos lo más pronto posible" [5]. En Colombia, el grupo de investigación GUIAME de la Universidad Nacional sede Medellín ha creado la herramienta “Programming Practice Center” (PPC) donde los estudiantes de la Universidad realizan sus entrenamientos, y la Universidad EAFIT también en Medellín ha desarrollado un completo programa de trabajo para el estudio de programación competitiva. Teniendo en mente todos estos trabajos realizados, se analiza la forma de diseñar e integrar un marco de trabajo efectivo que se adapte a las necesidades específicas del grupo de estudio en Programación Competitiva de la Universidad Francisco de Paula Santander, que pueda ser referente en la región y a cualquier equipo de maratones de programación.

En esta introducción se ha mostrado la importancia de la realización de este proyecto, y un pequeño resumen del marco teórico sobre el cual hemos trabajado. A continuación, se detalla el desarrollo de la investigación, indicando en métodos y materiales la forma en que este proceso se ha llevado a cabo, como se ha construido en marco de trabajo y como se ha aplicado en el ambiente en cuestión. Posteriormente se realiza un análisis de los resultados obtenidos a raíz de la implementación del proyecto. Luego se propone una discusión centrada en los resultados obtenidos y cómo interpretar los resultados en casos de replicar este proceso en un ambiente diferente. Finalmente, se detallan las conclusiones a las cuales hemos podido llegar a raíz de esta investigación.

II. DESARROLLO

1. Métodos y Materiales

El grupo de estudio en Programación Competitiva de la UFPS tiene como principal objetivo incentivar en los estudiantes un gusto particular por la programación y la competencia en un espacio en el que puedan mejorar sus habilidades algorítmicas, prepararse para las competencias y conocer, compartir y discutir con otras personas interesadas en el tema de manera tal que puedan tener una formación sólida y puedan llegar a representar a la Universidad en competencias oficiales como el International Collegiate Programming Contest ICPC. Con esto en mente se ha definido un marco de trabajo que consta de 5 componentes: Sesiones de entrenamiento, material de apoyo, guías imprimibles, repositorio de soluciones, metodología de trabajo y plataforma de entrenamiento.

Sesiones de entrenamiento: Las sesiones de entrenamiento son los espacios en los cuales los estudiantes se familiarizan con las temáticas tratadas, mejoran sus habilidades y resuelven problemas en un entorno competitivo. Se llevan a cabo dos tipos de sesiones:

En primer lugar, las tutorías, realizadas una vez cada semana con una duración de 2 horas y media, en las cuales se expone un tema nuevo o se refuerza un tema visto previamente con exposiciones magistrales de un docente o estudiante con dominio del tema (1 hora), se resuelven dudas y se genera discusión basada en los ejercicios propuestos en sesiones anteriores (10 a 20 minutos), se proponen ejercicios y se resuelven colaborativamente entre los diferentes integrantes (tiempo restante).

En segundo lugar, están las competencias, realizadas generalmente dos veces al mes con una duración de 5 horas, donde los estudiantes en grupos de 3 personas se enfrentan a estudiantes de otras Universidades en toda Latinoamérica con un conjunto de entre 8 y 12 ejercicios predefinidos escritos en idioma inglés, teniendo a su disposición un solo computador por equipo, buscando resolver el mayor número de problemas en el menor tiempo posible. Estas competencias son organizadas por la Red de Programación Competitiva (RPC) y la Colombian Collegiate Programming League (CCPL), y la Universidad es una sede activa en todas sus competencias.

Material de apoyo: Para el desarrollo de las sesiones se genera gran cantidad de material documental, que es compartido a través de un repositorio de acceso libre para que el entrenamiento no sea exclusivo de las sesiones, sino que pueda continuarse en cualquier momento por los estudiantes. Este material contiene el Syllabus, slides, tips y videos.

Guías imprimibles: En competencias oficiales de programación no se permite el uso de dispositivos

electrónicos ni consultas en internet, pero si es posible llevar impreso todo el material documental que los estudiantes deseen. Por esta razón, se han construido dos guías imprimibles (una para el lenguaje Java y otra para C++) con aquellos algoritmos que los estudiantes consideran que pueden ser útiles durante la competencia. Se encuentra en un repositorio público, para que cualquier estudiante pueda añadir sus aportes, o bien solicitar que alguien realice un aporte determinado. Estas guías están en constante actualización, y son compiladas con la nueva información antes de cada competencia oficial.

Repositorio de soluciones: Una vez finalizadas las competencias, los estudiantes pueden compartir sus soluciones en un repositorio público para ser visualizadas y repasadas por los demás integrantes (y el público en general). Esto permite conocer otras formas de solucionar los problemas, o entender nuevas formas de solucionar un problema determinado. Este repositorio cuenta con una plantilla con los metadatos que debe tener cada solución para ser fácilmente accedido e indexado. Al momento de escribir este artículo, han sido compartidas 145 soluciones.

Metodología de trabajo: La metodología de trabajo en realidad es el punto de unión de los componentes anteriores. Desde esta se modela el syllabus a enseñar y se articula la forma en que cada uno de los componentes trabaja para fortalecer el entrenamiento. En primer lugar, la metodología establece las delimitaciones en las temáticas estudiadas. Fue necesario realizar una selección de temáticas a trabajar durante las sesiones (Ver Tabla I); para esto se estudiaron y eligieron de acuerdo al nivel de los participantes algunos de los contenidos trabajados en la metodología utilizada por Steven Halim en el curso Competitive Programming de la Universidad Nacional de Singapur (NUS) [5]. Se eligió una cantidad de temas acorde a la duración de un semestre académico, teniendo en cuenta la intensidad horaria de las sesiones.

Tabla I: Temas propuestos para el grupo de estudio

	TEMA
1	Generalidades de las Maratones Programación: Introducción a las maratones, jueces online, competencias, beneficios y requisitos.
2	Estructuras de Datos: Arreglos, ArrayList, Pilas, Colas, TreeSet, TreeMap.
3	Recursión
4	Búsqueda en profundidad (DFS) y búsqueda en anchura (BFS)
5	Algoritmos Clásicos de Grafos I: Representación de Grafos en java, Algoritmo de Dijkstra, Floyd-Warshall, Árbol de expansión mínima de Kruskal.
6	Algoritmos Clásicos de Grafos II: Ordenamiento topológico, Componentes fuertemente conexas, Algoritmo de Tarjan.
7	Algoritmos Ávidos (Greedy)
8	Combinatoria, Conteo y Probabilidad:
9	Programación Dinámica: Problemas Clásicos
10	Algoritmos de Teoría de Números
11	Backtracking
12	Procesamiento de Cadenas: Algoritmo de Knuth-Morris-Pratt

Con base en estos temas se realizan las tutorías, definiendo desde el inicio de cada semestre un syllabus con los temas, las fechas y enlaces a todos los materiales de apoyo (previamente definidos en los componentes). Los materiales de apoyo son utilizados en cada tutoría: durante la exposición magistral, se utilizan las slides y videos definidos para esa sesión; durante la resolución de ejercicios, los estudiantes pueden acceder al repositorio de soluciones para buscar ejercicios similares, o bien a las guías imprimibles. Cada uno de estos materiales, además de estar disponible online para ser accedidos en cualquier momento, cumple una función central en el desarrollo de cada tutoría. Cada sesión finaliza con un conjunto de ejercicios propuestos sobre el tema, con diferentes niveles de dificultad, donde los estudiantes se miden de manera individual. Estos ejercicios se proponen en diferentes plataformas de entrenamiento online: Codeforces, Universidad de Valladolid Online Judge (Uva), URI Online Judge, etc. las cuales en cuestión de segundos indican al estudiante si su solución es correcta o incorrecta. Estos datos son recolectados semanalmente y con ellos se construye un ranking de estudiantes destacados. Si bien este ranking no aporta calificaciones académicas (el grupo de estudio no se encuentra relacionado con ninguna materia del pensum), si es tenido en cuenta para elegir a aquellos estudiantes que representan a la Universidad en los diferentes eventos de Programación Competitiva, además de ser un factor adicional para motivar a los estudiantes: La satisfacción de ocupar los primeros puestos los mantiene en una constante preparación. Con estas actividades se busca que los estudiantes entiendan y adopten las actitudes necesarias para participar en las competencias. Estas competencias requieren diferentes conocimientos, habilidades de programación, resistencia (al principio es complicado mantener la concentración por las 5 horas que dura una competencia), buen trabajo en equipo, y competitividad [7] [8].

Plataforma de entrenamiento: Con el marco de trabajo funcionando correctamente, el siguiente paso predefinido ha sido integrar todos los componentes en una sola plataforma donde se lleve a cabo todo el entrenamiento. Esta plataforma contendrá todos los materiales de apoyo dotándolos de interactividad, los ejercicios propuestos en un ambiente de calificación automático, y sobre todo permitirá realizar un monitoreo constante y completo sobre cada uno de los aspectos del entrenamiento. Si afirmamos que la metodología constituye el enlace práctico entre todos los componentes, la plataforma constituye el enlace de software entre ellos.

2. Resultados

Durante la aplicación de la metodología de trabajo propuesta en el grupo de estudio se realizaron 10 tutorías y se participó en 8 competencias de entrenamiento oficiales con RPC y CCPL, contando con la participación de 21 estudiantes habituales (asistieron a más del 70% de las sesiones), y 14 estudiantes ocasionales (asistieron solo a sesiones con temáticas específicas).

Los 4 primeros componentes del marco de trabajo han sido completamente creados y documentados (Sesiones de entrenamiento, material de apoyo, guías imprimibles, repositorio de soluciones y metodología de trabajo), y se han colocado en práctica durante todo el semestre I del 2016. Su aplicación ha sido exitosa: si se suman los ejercicios correctamente realizados en las tutorías por los participantes del grupo, tenemos un total de 1225 ejercicios durante el semestre.

Con todo el trabajo desarrollado en el grupo de estudio se logró la construcción de un repositorio en GitHub (<http://programacioncompetitivaufps.github.io/index.html>) en donde se encuentran los slides correspondientes a 6 de las temáticas trabajadas, el documento con los ejercicios propuestos y las soluciones de todos los ejercicios desarrollados ya sea en competencias o en las sesiones de entrenamiento. Con el fin de complementar todo el trabajo desarrollado y facilitar la comprensión de los temas se crearon 3 versiones de la guía imprimible de algoritmos de la UFPS que cuenta con la implementación de 37 algoritmos de grafos, manejo de cadenas, geometría computacional, matemáticas, programación dinámica y optimización de entradas y salidas; y que se encuentra disponible en lenguaje java y C++.

Las competencias son una parte fundamental del grupo de estudio, mejorar el desempeño de los participantes es el principal objetivo de todo el trabajo realizado, es por esto que se realizó un seguimiento de los grupos que participaron en las competencias de entrenamiento organizadas por RPC y CCPL. En todas las competencias realizadas por RPC en el semestre I del 2016 por lo menos un equipo del grupo de estudio se ubicó entre los 10 primeros a nivel nacional. Del mismo modo, en las competencias realizadas por CCPL, por lo menos un equipo se ubicó en los primeros 15 puestos.

3. Discusión

El grupo de estudio en Programación Competitiva existe desde hace 4 semestres, aunque la construcción del marco de trabajo inició en el semestre II del 2015, y

solo durante el primer semestre del 2016 se implementó correctamente. Esto permite analizar los resultados en perspectiva con respecto a lo realizado en semestres anteriores, cuando no existía el marco de trabajo.

En primer lugar, encontramos que la permanencia de estudiantes en el grupo ha aumentado. 21 estudiantes hicieron parte activa durante todo el proceso (frente a 9 estudiantes en el semestre inmediatamente anterior). Esto es relevante, dado que en ambos semestres se contó con un total de asistentes similar (35 estudiantes el semestre actual vs. 36 el semestre anterior), lo cual demuestra que la permanencia de los estudiantes es mucho mayor con un marco de trabajo establecido que da orden y dinámica al grupo. La permanencia de los estudiantes es un pilar vital de este proyecto, pues de ellos depende el marco, y solo aquellos que demuestren mayor constancia son los que logran buenos resultados. El número de ejercicios realizados también se incrementó considerablemente. En total fueron 1225 ejercicios solucionados contando únicamente los ejercicios propuestos. Esto permite una mayor apropiación de los temas, y mayor práctica a la hora de participar en una competencia.

Además, en cuanto al marco de trabajo, se ha creado un completo set de componentes que buscan ser replicables por otras instituciones que inician su proceso en competencias de programación. Estos componentes han sido base fundamental del grupo de estudio en cuestión, y se siguen nutriendo con nuevos aportes regularmente. La participación en la Maratón Nacional de Programación Universitaria ha mejorado con este tipo de preparación. Los resultados obtenidos comparados con años anteriores han mejorado notablemente, verificando el nivel de éxito que ha tenido la aplicación de este marco de trabajo.

III. CONCLUSIONES

Los resultados de la implementación de este marco de trabajo han sido positivos. Un número significativo de estudiantes se ha vinculado al grupo de estudio, participando activamente de las diferentes actividades planteadas, mejorando sus habilidades algorítmicas, y complementando lo aprendido en el grupo de estudio con lo aprendido en sus clases tradicionales.

En total se impartieron 25 horas de tutorías y 40 horas de competencias, generando una cantidad importante de material de apoyo que se comparte de manera libre para todos los estudiantes que deseen acceder a él posteriormente.

El marco de trabajo se construyó y adaptó según las mismas opiniones de los estudiantes hasta tomar la

forma que finalmente fue compartida en este documento. Y ese marco tuvo su validación al verificar el avance de los estudiantes en las competencias llevadas a cabo a lo largo del semestre. Los resultados de RPC y CCPL demuestran que finalmente, la metodología utilizada ha sido efectiva. Por esta razón, dicha metodología se ha documentado por completo, de manera que pueda ser utilizada y aplicada en el futuro, aun cuando el personal encargado del grupo de estudio haya cambiado, siguiendo el ejemplo de la Universidad EAFIT [6].

En todo momento, los estudiantes estuvieron en el centro del marco de trabajo, pues este proyecto se sale del enfoque metodológico de aula tradicional, y hace parte del aprendizaje colaborativo. Por este motivo sus opiniones se tuvieron en cuenta en todo el proceso. Del mismo modo, los mismos estudiantes compartieron algunas de sus soluciones para hacer parte del repositorio digital que se encuentra disponible de manera abierta en GitHub. Su progreso en cada una de las competencias fue medido, y en futuras iteraciones de este proyecto, una vez que la plataforma de entrenamiento propio sea puesta en funcionamiento, serán ellos los encargados de validar y efectuar retroalimentación. En las 50 horas de competencia realizadas durante el semestre, no hay profesores, ni explicaciones magistrales. Cada estudiante hace uso de sus habilidades de trabajo en equipo con dos compañeros para resolver el mayor número de ejercicios posibles en el menor tiempo. Todas estas características han permitido que los estudiantes continúen en el grupo y mantengan la retroalimentación activa.

IV. REFERENCIAS

- [1] Arefin, A. S. Art of Programming Contest, 2nd edition, Dhaka, Gyankosh Prokashoni, 2006, pp. 14-18.
- [2] Bloomfield, A. and Sotomayor, B. A Programming Contest Strategy Guide, In Proceedings of the 47th ACM Technical Symposium on Computing Science Education (SIGCSE '16), 2016, pp. 609 - 614.
- [3] Halim, S. and Halim, F. Competitive Programming in National University of Singapore, ediciones Sello Editorial SL (Presented at Collaborative Learning Initiative Symposium, Harbin, China, 2010.
- [4] Jackson, S., Ellis, H., and Crouse, R. An analysis of team performance in high school programming contests, In Proceedings of the 15th Annual Conference on Information technology education (SIGITE '14), 2014, pp. 27-32.

[5] Halim, S. and Halim, F. Competitive Programming 3, 3rd. edition, Singapore, Lulu, 2013.

[6] Echavarría Uribe, A. and Cardona Mc'Cormick J. F. Desarrollo e implementación de un programa de trabajo para el Semillero de Programación, 2013, 4p.

[7] Chavey, D., Monrey, T. L., Van Brackle, D. and

Werth, J. Preparing a team for the ACM scholastic programming contest (panel session). In Proceedings of the 19th annual conference on Computer Science (CSC '91). 1991, p. 701.

[8] Acharya, S. Challenges and Benefits of Programming Competitions as Outreach to High School Students, ASEE Annual Conference, 2013, pp. 1-14.

RESUMEN BIOGRÁFICO DE AUTORES



Gerson Yesid Lázaro Carrillo. Estudiante de Ingeniería de Sistemas de la Universidad Francisco de Paula Santander (UFPS) – Cúcuta, Colombia. Coordinador del grupo de estudio en Programación Competitiva UFPS, embajador de campus de Hackerearth y miembro del semillero de investigación en Linux y Software Libre (SILUX UFPS).



Angie Melissa Delgado León. Estudiante de Ingeniería de Sistemas de la Universidad Francisco de Paula Santander (UFPS) – Cúcuta, Colombia. Coordinadora del grupo de estudio en Programación Competitiva UFPS, y miembro del semillero de investigación en Linux y Software Libre (SILUX UFPS).



Fredy Humberto Vera Rivera

Universidad Francisco de Paula Santander (UFPS) - Cúcuta - Colombia

Departamento de Sistemas e Informática.

Grupo de Investigación y desarrollo de Ingeniería del Software - Gidis. Entrenador (Coach) equipos de maratones de programación de la UFPS. Magíster en Ingeniería área informática y ciencias de la computación - UIS. Ingeniero de Sistemas - UIS