

## JUEZ AUTOMÁTICO DE CÓDIGO, UNA HERRAMIENTA PARA MEJORAR LAS HABILIDADES DE PROGRAMACIÓN

D. Morales Orozco<sup>1</sup>  
L. G. Gutiérrez Torres<sup>2</sup>  
P. Vega Flores<sup>3</sup>

### RESUMEN

El presente documento describe como a raíz de la experiencia de los docentes al involucrar a un grupo de estudiantes en distintos concursos de programación en los que se ha llegado a estar dentro de los primeros 10 lugares a nivel México y Centroamérica se llegó a la conclusión de que el juez automático de código que es uno de los componentes principales de estos concursos podría ser de gran utilidad durante la impartición de materias de programación del Instituto Tecnológico Superior del Sur de Guanajuato, con esto en mente se comenzó a revisar la mejor manera de aplicar este tipo de juez al aula, teniendo un doble objetivo en primer lugar facilitar el aprendizaje de la programación y en segundo lugar y como consecuencia natural de este primer objetivo el poder aumentar la participación de los alumnos en los concursos de programación y así poder ser más competitivos en los concursos de programación en los que participa el instituto.

### ANTECEDENTES

Los concursos de programación tienen mucho que ofrecer a la educación (Verhoeff, 1997) y además pueden despertar el interés en suficientes personas como para ser un incentivo en la formación de los alumnos (Cernuda del Río & Gayo Avello, 2003) y es con base a estas dos premisas que el Instituto Tecnológico Superior del Sur de Guanajuato (ITSUR), desde el año 2010 comenzó a trabajar con estudiantes y dos profesores en el "International Collegiate Programming Contest" (ICPC), el cual es un concurso en el cual se involucran a universidades del todo el mundo más de 2534 universidades de 101 países que mediante competencias regionales clasifican a los equipos a la final mundial (International Collegiate Programming Contest) comenzando con un lugar muy modesto en el 2010 apenas se logró el lugar 131, en 2011 la posición 65, en 2012 se ganó el lugar 26, en 2013 se obtuvo el octavo lugar (Gutiérrez Torres, Morales Orozco, & Martínez López, 2014) y para el 2014 se logró mantener en los primeros 10 lugares de la región de México y Centro América.

Con estos resultados, se notó a inicios de 2014 que a pesar de los esfuerzos por involucrar a las dos carreras afines a este tipo de concursos las cuales son ingeniería informática e ingeniería en sistemas computacionales la participación era únicamente de alumnos de la carrera de ingeniería sistemas, fue por este motivo que se puso como objetivo el incentivar la participación de equipos de informática en los concursos de programación y por consecuencia mejorasen sus habilidades de programación.

También se pensó en una manera de reducir el tiempo que les lleva a los alumnos a ser competitivos en los concursos del ACM-ICPC, ya que para estar en los primeros diez lugares fue un proceso de cuatro años trabajando con más o menos los mismos alumnos.

Para lograr estos dos objetivos se planteó la siguiente pregunta ¿Puede la aplicación de un juez automático de código en el aula de clase ayudar a mejorar las habilidades de

<sup>1</sup> Profesor de tiempo completo. Instituto Tecnológico Superior del Sur de Guanajuato. [d.morales@itsur.edu.mx](mailto:d.morales@itsur.edu.mx).

<sup>2</sup> Profesor de tiempo completo. Instituto Tecnológico Superior del Sur de Guanajuato. [licgerman@itsur.edu.mx](mailto:licgerman@itsur.edu.mx).

<sup>3</sup> Profesor de tiempo completo. Instituto Tecnológico Superior del Sur de Guanajuato. [p.vega@itsur.edu.mx](mailto:p.vega@itsur.edu.mx).

programación de los alumnos? Para responder a esta pregunta se decidió aplicar toda la experiencia de los profesores obtenida como entrenadores de los concursos de programación y llevar esta experiencia a las clases centrándose en la utilización de un juez automático de código.

El juez automático de código es el principal componente de los concursos de programación, ya que un juez de este tipo es objetivo e imparcial (Petit & Roura, 2009) porque lo que hace el juez automático es recibir un código, lo compila, lo ejecuta contra casos de entrada preestablecidos y compara los resultados del programa para verificar si el resultado es el esperado, esto se puede observar en las figuras Figura 1 y Figura 2 donde se muestra primeramente como es un problema y ejemplos de los resultados que puede regresar el juez automático.

**0 Resta basica**

Puntos	Límite de memoria	32MB
Límite de tiempo (caso)	1s	Límite de tiempo (total)
		60s

**Descripción**  
Restando !!!

**Entrada**  
Pide dos numeros enteros

**Salida**  
mostrar el resultado de la resta de los numeros pedidos

**Ejemplo**

5	2	3	Al primer entero le resto el segundo, y muestro la diferencia
5	10	-5	puede haber resultados negativos

**Límites**

- Usar enteros
- Programa basico

**Figura 1. Redacción de un problema en el juez automático de código OmegaUp**

En la Figura 1 se muestra cómo se presenta un problema en los concursos de programación en los que hay una descripción del problema, una descripción de la entrada que recibirá el problema, y una especificación de cómo se debe mostrar la salida de la ejecución del problema al igual que ejemplos de la entrada y salida.

Envíos

ID	Lenguaje	Tiempo	Memoria	Enviado	Status	Porcentaje	Penalty	Código
c8c9d	java	0.14	4.29	2014-02-25 17:39:55	AC	100.00%	0	ver
57538	py	0.02	4.84	2014-08-06 12:16:37	RTE	0.00%	0	ver
e73d6	py	0.01	4.89	2014-08-06 13:04:24	AC	100.00%	0	ver
9bb54	py	0.01	4.89	2014-08-06 13:30:26	AC	100.00%	0	ver
Nuevo envío								

**Figura 2. Resultados de envíos al juez automático de código**

Con todo lo anterior en mente se planteó la siguiente interrogante ¿Es posible usar un juez automático a las clase de programación del ITSUR?

Para responder esta pregunta se comenzó primeramente con la clase de Programación Orientada a Objetos de ingeniería informática del ITSUR a introducir el uso de un juez automático en línea específicamente con la plataforma OmegaUp que es una plataforma para concursos en línea basado en la nube y de código abierto que es utilizado para la Olimpiada Mexicana de Informática (Chávez, González, & Ponce, 2014), y tomando las ventajas que mencionan Petit & Roura(2009) de una retroalimentación inmediata a los alumnos de si su solución es correcto o incorrecta, seguimiento de los avances de los alumnos, y sobre todo que el alumno es evaluado precisamente en aquello que debe aprender lo cual es resolver problemas de programación.

## METODOLOGÍA

Para lograr incorporar el juez automático en primer lugar se realizó un análisis para elegir la mejor plataforma a ser utilizada y su mejor forma de hacer uso de ella.

Primeramente se analizaron las opciones de utilizar una plataforma en línea o una plataforma local, se optó dar más peso a una plataforma en línea porque ya hay una gran cantidad de problemas listos para ser tomados y ser aplicados en clase como se demuestra que a marzo 2015 hay alrededor de 1500 problemas registrados en la plataforma OmegaUp y en el Caribbean Online Judge (COJ) el cual es una plataforma de juez automático que tiene como uno de sus objetivos, obtener capacitación para participar en competiciones de programación el cual surgió de la Universidad de las Ciencias Informáticas (Caribbean Online Judge, 2010) y esta plataforma tiene alrededor de 2100 problemas al 20 de marzo de 2015.

A continuación se revisaron las opciones de distintos jueces disponibles en línea, para centrarnos finalmente en dos opciones la plataforma del Caribbean Online Judge (COJ) y OmegaUp principalmente por el tiempo de uso que los profesores han invertido en la utilización de ambas plataformas y de este uso han demostrado tener una estabilidad bastante buena, con algunas caídas esporádicas del servidor del COJ pero los encargados de la plataforma siempre han demostrado una gran disposición a restablecer el servicio lo más pronto posible cuando esto ocurre.

De estas dos plataformas se eligió OmegaUp debido a que la base de problemas que tiene esta en español, ya que normalmente se ve que el idioma inglés representa una barrera para muchos de los estudiantes, pero fuera de esta ventaja lo que inclino la balanza a usar OmegaUp fue que su herramienta para crear sus propios concursos y subir problemas propios a la plataforma es mucho más fácil de realizar.

Una vez que se tuvo decidida la plataforma a ser utilizada el proceso para ir utilizando la plataforma en clase se resume en los pasos que se detallan en los siguientes párrafos.

Primeramente generar un concurso Inicial para que se familiaricen con la plataforma en el cual el objetivo sea que los estudiantes conozcan cómo serán presentados los problemas que tendrán que resolver y como es la forma correcta de mandar su solución al problema y como obtener una respuesta positiva por parte del evaluador automático, un ejemplo de cómo se muestra un concurso se puede ver en la Figura 3 y un ejemplo de cómo el juez automático da la retroalimentación se puede ver la Figura 2 los resultados que da el juez automático principalmente son los que se muestran en la Tabla 1.

**Tabla 1. Retroalimentación Juez Automático**

Mensaje	Significado
<b>AC</b>	Aceptado el problema paso todos los casos de prueba.
<b>CE</b>	Error de Compilación.
<b>RTE</b>	Error en tiempo de ejecución.
<b>TLE</b>	El tiempo que tiene para que el código resuelva el problema fue excedido

En la Figura 3 los detalles que muestra un concurso de programación es los problemas que se van a resolver durante el concurso, la duración del concurso, el ranking de cómo van los participantes del concurso.

POO - Examen Unidad 2			
00:00:00			
Problemas	Ranking	Envíos	Clarificaciones
Resumen			
A Rifa			
B. Supresores de picos			
C. Arriba del promedio			
Usuario	Total		
1 POOErikEstrada	33	0	
2 mvazquez101010	20	0	
3 caropreso	7	0	

POO - Examen Unidad 2	
Examen unidad 2 Programación Orientada a Objetos ITSUR	
Hora de inicio	2014-03-20 12:55:00
Hora de fin	2014-03-20 14:40:00
Fin de actualización del scoreboard	
Tiempo para resolver los problemas	
Organiza	

**Figura 3. Ejemplo de Examen Tipo Concurso**

La primer materia donde se comenzó a utilizar la herramienta de juez automático fue en programación orientada a objetos, para continuar con estructura de datos, inteligencia artificial y actualmente con la materia de fundamentos de programación, es importante destacar que la selección de materias se dio de acuerdo a la carga académica que reciben los docentes lo cual no está en las manos del docente decidir y por tal motivo se puede apreciar que no hay un orden en el cual se iniciaron a aplicar porque idealmente hubiera sido mejor empezar con la materia de fundamentos de programación.

Fundamentos de programación es la materia más idónea para este tipo de plataforma ya que en programación orientada a objetos adicionalmente a que el alumno solucione el problema se debe revisar que haya seguido correctamente el paradigma orientado a objetos lo cual requiere una revisión del código adicional por parte del profesor, la ventaja es que se puede revisar el código directamente que enviaron los alumnos como se muestra en la Figura 4 y Figura 5 esto también da una ventaja que es el poder indicar a los alumnos de forma relativamente sencilla en que se están equivocando al enviar su código al juez automático.

Envíos									
Veredicto:	Todos	Status:	Todos	Problema:	Todos	Lenguaje:	Todos	Usuario:	
Usuario	Problema	Lenguaje	Status	Puntos	Penalty	Rejuecear		Detalles	
erick.g.a.15	<a href="#">arriba_del_promedio</a>	java	RTE	0.00	0	rejudge	debug	details	
erick.g.a.15	<a href="#">arriba_del_promedio</a>	java	CE	0.00	0	rejudge	debug	details	
poo_benjamin	<a href="#">arriba_del_promedio</a>	java	WA	0.00	0	rejudge	debug	details	
mvazquez101010	<a href="#">Rifa</a>	java	RTE	19.80	0	rejudge	debug	details	
jaimortizdiaz375	<a href="#">Rifa</a>	java	TLE	0.00	0	rejudge	debug	details	
POOErikEstrada	<a href="#">supresores_de_picos</a>	java	AC	33.00	0	rejudge	debug	details	
jaimortizdiaz375	<a href="#">Rifa</a>	java	CE	0.00	0	rejudge	debug	details	
caropreso	<a href="#">Rifa</a>	java	PA	6.60	0	rejudge	debug	details	
caropreso	<a href="#">Rifa</a>	java	PA	6.60	0	rejudge	debug	details	

Figura 4. Resumen de envíos a concurso

```

import java.util.Scanner;
public class Main{

    private int a;
    private int b;
    private int resultado;

    public int suma(){
        resultado =a + b;
        return a+b;
    }

    public int restar(){
        resultado =a - b;
        return a-b;
    }

    public int multiplicar()

```

Figura 5. Vista de Código Enviado al Sistema

Una de las tareas más importantes al momento de usar la plataforma es seleccionar problemas para el tema que se está presentando o evaluando y una vez seleccionados los problemas se debe crear un concurso con los problemas que fueron seleccionados previamente.

Otra actividad que se tiene que estar realizando es revisar aleatoriamente códigos para verificar que no haya copias o tratar de disminuir esto porque si son concursos que pueden realizar o trabajar extra clase puede darse el caso de que a los estudiantes manden un código que fue previamente aceptado por uno de sus compañeros.

## DISCUSIÓN DE RESULTADOS

Los resultados han sido positivos ya que se ha demostrado que se puede perfectamente utilizar como una herramienta de apoyo durante la impartición de clases de programación, como lo demuestran los 35 concursos organizados en la plataforma para distintas materias como se muestran en la Tabla 2.

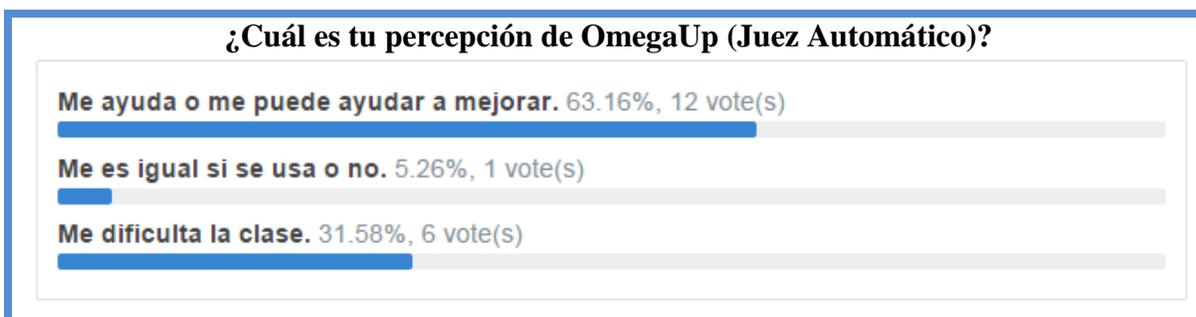
**Tabla 2. Concursos Organizados en plataforma OmegaUp**

Materia	Carrera	No. Concursos.	Semestre
Programación Orientada a Objetos. Estructura de Datos.	Ingeniería Informática.	7	ENERO-JUNIO 2014
	Ingeniería en Sistema.	11	AGOSTO-DICIEMBRE 2014
Estructura de Datos.	Ingeniería Informática.	9	AGOSTO-DICIEMBRE 2014
Fundamentos de Programación.	Ingeniería Informática.	8	ENERO-JUNIO 2015 (Hasta Marzo)

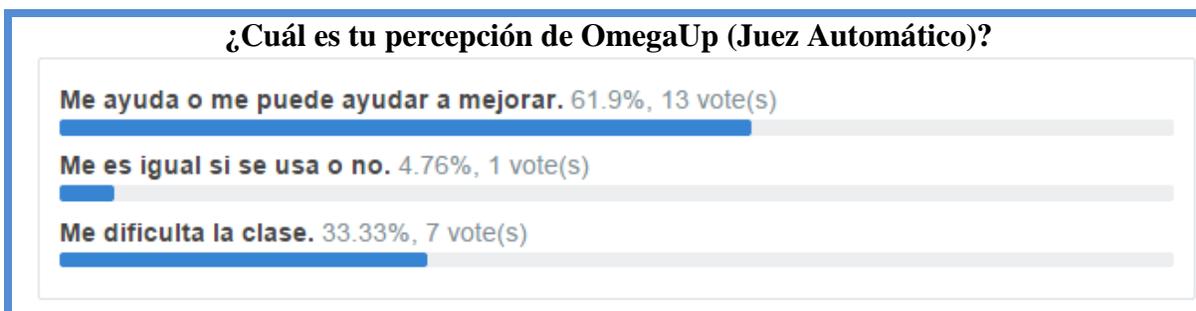
También un resultado importante es que se pudo formar un equipo de ingeniería informática durante el ciclo de participación 2014 en el ACM-ICPC lo que no se había logrado en lo que se llevaba de participar en estos concursos por parte del ITSUR. También se logró incrementar en 2 equipos más de ingeniería en sistemas de 3er semestre y cabe destacar que estos equipos fueron competitivos ya que al menos resolvieron dos problemas en el concurso de clasificación al regional del ACM-ICPC en Octubre de 2014, lo que es significativo si se toma en cuenta que hasta 2014 más del 50% de equipos que resuelven un problema no resolvían ningún problema.

Otro resultado exitoso de utilizar los jueces automáticos en clase es que en un periodo de un año se pudo involucrar a un alumno de 4to semestre para que pasará a formar parte del equipo 1 del ITSUR que fue el que obtuvo el 10mo lugar en 2014 y que por que ya egreso uno de los integrantes del equipo ahora este alumno está iniciando competencias en 2015 como parte del equipo 1 y es un alumno que demostró que tiene las habilidades, el interés y el empeño para subir su nivel y el próximo año ser el que impulse a lograr mejores resultados en esta competencia.

Por último como muestran las Figuras 5 y 6, a más del 60% de los alumnos de los dos grupos tanto de ingeniería informática e ingeniería en sistemas computacionales de la materia de informática consideran que el juez automático les ayuda o les puede ayudar a mejorar esto es significativo porque a principio de la materia hubo muchas quejas acerca de que era complicado subir sus códigos a la plataforma, pero una vez que se les explico nuevamente las ventajas de la plataforma y de darles un tiempo para que se acostumbrarán al juez automático la opinión acerca de la herramienta fue mejorando al punto de que algunos de los alumnos expresaron opiniones de que era mucho mejor que los evaluaran de esta manera contraria a la forma tradicional en que los alumnos entregan su código al profesor y este un tiempo después les dice si fue o no correcto lo que entregaron.



**Figura 6. Encuesta percepción OmegaUp EDD Informática**



**Figura 7. Encuesta percepción OmegaUp EDD Sistemas**

## CONCLUSIONES

El preparar mejor a los alumnos que ellos cuenten con herramientas que les faciliten su aprendizaje y su práctica para ser mejores ingenieros no es una tarea fácil ya que es invertir una gran cantidad de tiempo que en su mayoría es adicional a la carga que tiene el docente, ya que es investigar el uso del juez automático, es ir a prueba y error mejorando lo que se va aplicando en clase, pero al final todo el esfuerzo vale la pena cuando vez que los alumnos están mejorando y teniendo un mejor desempeño en las materias.

Para el 2014 se logró la participación en la batalla de programadores que fue la fase de clasificación al regional del 2014 del ACM-ICPC, se logró incluir a 1 equipo de informática y dos equipos más formados por alumnos a los que se les dio clase con apoyo de la plataforma, adicionales a los equipos que normalmente han ido participando logrando

con esto tener la máxima participación de equipos del ITSUR que ha tenido en este tipo de eventos.

Hay que tener especial cuidado de explicar a los alumnos como es que el juez automático los evaluará porque en un principio es complicado que los alumnos se quiten la idea de que su código está bien pero la plataforma no los está evaluando de forma correcta, para esto es muy importante recalcarles que deben probar sus códigos con distintos datos de entrada no solamente con los que vienen en la redacción del problema.

Por último una tarea pendiente es mejorar el sistema de detección de copias de código ya que sin un buen control de esto algunos alumnos se les puede hacer fácil caer en este tipo de plagio con la esperanza de que como el juez automático ya les dio una respuesta positiva con el código de su compañero ya con esto van a tener un aprobatorio en la materia.

### **BIBLIOGRAFÍA**

Caribbean Online Judge. (5 de Junio de 2010). Caribbean Online Judge. Recuperado el 29 de Marzo de 2015, de <http://coj.uci.cu/general/about.xhtml?lang=es>

Cernuda del Río, A., & Gayo Avello, D. (2003). Los concursos de programación como herramienta didáctica. IX Jornadas de Enseñanza universitaria de la Informática.

Chávez, L. H., González, A., & Ponce, J. (2014). OmegaUp: Cloud-Based Contest Management System and Training Platform in the Mexican. IOI Journals. International Olympiad in Informatics.

Gutiérrez Torres, L. G., Morales Orozco, D., & Martínez López, F. J. (2014). Los Concursos de Programación como Detonante del Aprendizaje. XLI Conferencia Nacional de Ingeniería: La Formación de los Ingenieros en México.

International Collegiate Programming Contest. (s.f.). International Collegiate Programming Contest. Recuperado el 25 de Marzo de 2015, de <http://icpc.baylor.edu/>

Petit, J., & Roura, S. (2009). Programación-1: Una asignatura orientada a la resolución de problemas. XV Jornadas de enseñanza Universitaria en Informática. Barcelona.

Verhoeff, T. (Noviembre de 1997). The Role of Competitions in Education, Recuperado el 3 de Marzo de 2015, de <http://olympiads.win.tue.nl/ioi/ioi97/ffutwrlld/competit.html>