

Perfil del alumno de Computación para el diseño de un Sistema Tutor

Computer student profile for the design of the Tutor System

Guillermina Sánchez Román¹, Josefina Guerrero García¹, Erika A. Martínez Mirón¹

¹ Benemérita Universidad Autónoma de Puebla, México

guille.sroman@correo.buap.mx , jguerrero@cs.buap.mx , erika.mtzm@correo.buap.mx

RESUMEN. Desde hace varias décadas ha existido un interés en conocer los factores que se involucran en el aprendizaje de programación y los retos que existen en el proceso de desarrollo de habilidades algorítmicas con el fin de minimizar las estadísticas de reprobación en esta área. Por lo tanto, es necesario contemplar estrategias didácticas con las que el estudiante aprenda de forma autónoma. Se requiere que los estudiantes de nuevo ingreso a nivel superior desarrollen habilidades algorítmicas como parte de sus competencias genéricas primordiales. El presente trabajo muestra las características del estudiante, a partir de tres aspectos: el conocimiento previo, el estilo de aprendizaje, y nivel de razonamiento, para desarrollar habilidades algorítmicas en estudiantes de nuevo ingreso a la Facultad de Computación. La metodología utilizada es cuantitativa, se aplicaron tres test: estilos de aprendizaje, razonamiento científico y pensamiento computacional. La muestra son estudiantes de nuevo ingreso a licenciatura en computación de México y Colombia. El proceso estadístico es a partir de estadística no inferencial, los resultados indican que en ambas regiones los estudiantes presentan tendencias similares en el estilo de aprendizaje, nivel de razonamiento y pensamiento computacional. Principalmente los datos obtenidos del test de pensamiento computacional aplicados a estudiantes se comparan para determinar las tendencias de sus habilidades algorítmicas, los resultados abonan en la caracterización para el modelo de estudiante y al diseño de actividades para el prototipo del STI (Sistema Tutor Inteligente).

ABSTRACT. Since several decades there has been an interest in knowing the factors that involve programming learning and the challenges that exist in the process of developing algorithmic skills in order to minimize the low learning outcomes in this area. Therefore, it is necessary to consider didactic strategies to help students to learn autonomously. As part of their primary generic competencies, undergraduate students are required to develop algorithmic skills. This paper reports the results of reviewing some student' characteristics (prior knowledge, learning style, and level of reasoning), needed to develop algorithmic skills in recent students at the Computing Faculty. The methodology used is quantitative, three tests were applied: learning styles, scientific reasoning and computational thinking. The sample comprehends new entrance undergraduates in Computer Science from Mexico and Colombia. The statistical process is based on non-inferential statistics, the results indicate that in both regions students have similar trends in learning style, reasoning level and computational thinking. These results will inform the characterization for the student model and the design of activities for the ITS prototype (Intelligent Tutoring System).

PALABRAS CLAVE: Habilidades algorítmicas, Estilo de aprendizaje, Educación superior, Sistema tutor inteligente, Modelo del estudiante, Razonamiento.

KEYWORDS: Algorithmic skills, Learning style, Higher education, Intelligent tutor System, Student model.

1. Introducción

Hoy en día es importante integrar en la educación las tecnologías de la información y comunicación (TIC) que se van generando. El área de ciencias se ha caracterizado por tener poca demanda y a su vez mayor dificultad para su aprendizaje; en particular, en el área de ciencias de la computación los estudiantes tienen dificultades en los cursos iniciales de programación, lo que da lugar a altas tasas de reprobación (Moroni & Señas, 2005, Rosanigo & Paur, 2006).

Estudios realizados muestran que las habilidades algorítmicas de los estudiantes universitarios que ingresan a carreras del área de Ciencias de la Computación son todavía deficientes para las características que se debe tener en este nivel de estudios. Además, existe ausencia del dominio de los aspectos cognitivos e instrumentales relacionados con estas habilidades, tanto para su enseñanza como para su evaluación (Moroni & Señas, 2005; Rosanigo & Paur, 2006).

A partir de la inferencia del perfil del estudiante cada alumno estará en una categoría según su estilo de aprendizaje, nivel de razonamiento y el conocimiento previo (Moroni & Señas, 2005). El análisis de las habilidades algorítmicas en la programación básica se integrará posteriormente en el diseño del Sistema Tutor Inteligente (STI). El STI generará un conjunto de estrategias pedagógicas y didácticas que personalizarán el proceso de aprendizaje de acuerdo a las necesidades particulares de cada alumno, esperando mejorar sus habilidades algorítmicas para la resolución de problemas, evitando la deserción escolar (Rosanigo & Paur, 2006). En el módulo de estudiante se propone contemplar las características del estudiante a partir de tres dominios: el razonamiento, estilo de aprendizaje y habilidades algorítmicas; integrando las características en los procesos de aprendizaje para el diseño de actividades lúdicas y de acuerdo a las habilidades requeridas para la programación.

El objetivo de esta investigación es caracterizar a los estudiantes del área de computación en México y Colombia a partir de tres vertientes: las habilidades algorítmicas, el estilo de aprendizaje y el nivel de razonamiento que tienen los estudiantes de computación con una muestra a comparar los resultados encontrados en los grupos explorados en las dos regiones, se trata de analizar cuáles son las habilidades que debe desarrollar en el proceso de solución de problemas y contrastar los datos obtenidos.

2. Estado del Arte

De acuerdo al enfoque cognitivista y a la teoría instruccional de Gagné se requiere saber cómo el estudiante obtiene el conocimiento previo, cómo adquiere las representaciones del mundo, cómo almacenan y se recuperan los datos en la memoria. El modelo para determinar el estilo de aprendizaje que da mayor información en el área de ciencias de acuerdo a la literatura (Franzoni-Velázquez & Cervantes-Pérez, 2012) y ha dado mejores resultados para identificar el estilo preferente, es el propuesto por Felder y Silverman, por lo que se considera este modelo para evaluar a los estudiantes de ciencias de la computación para identificar cuál es su estilo preferente de aprender. Se aplica el instrumento de estilos de aprendizaje (Felder & Silverman, 1998) que evalúa cómo el estudiante procesa, percibe, representa y comprende la información, se definen cuatro pares dicotómicos a identificar: activo-reflexivo, sensorial-intuitivo, visual-verbal, secuencial-global que permiten saber la predominancia de su estilo de aprendizaje.

El perfil de estudiante es un aspecto muy importante que determina las características que tiene este en un momento dado. Algunos de los aspectos que se pueden considerar son: datos personales, estilo de aprendizaje, estado anímico, personalidad, psicológica, académico, emocional, ambiental, entre otros. Normalmente se utiliza el término "estilo de aprendizaje" para referirse al hecho de que cada persona utiliza su propio método o estrategias para aprender, aunque las estrategias varían de acuerdo a lo que se requiera aprender, por lo tanto, cada individuo debe desarrollar sus propias preferencias que identifique y defina su estilo de aprendizaje.

Para conceptualizar las habilidades algorítmicas es necesario comenzar con la definición de Pensamiento Computacional, que es la capacidad de formular y solucionar problemas apoyándose en los conceptos



fundamentales de la computación, y usando la lógica sintaxis de los lenguajes informáticos de programación: secuencias básicas, bucles, iteraciones, condicionales, funciones y variables (Román-González, Pérez-González & Jiménez-Fernández, 2015).

El pensamiento algorítmico es, de alguna manera, un conjunto de habilidades que están conectadas a la construcción y comprensión de algoritmos. Según Futschek (2006), este pensamiento incluye las siguientes capacidades:

- a) Analizar problemas dados.
- b) Especificar un problema de manera precisa.
- c) Encontrar las acciones básicas que son adecuadas para resolver el problema dado.
- d) Construir un algoritmo correcto para resolver un problema determinado, utilizando las acciones básicas.
- e) Pensar en todos los posibles casos tanto especiales como normales de un problema.

El pensamiento computacional se refiere a diferentes dimensiones que van desde las habilidades genéricas hasta la implementación de algoritmos en algún lenguaje. Sin embargo, es posible rescatar que antes de que se piense en programar se deben de desarrollar las habilidades básicas de pensamiento lógico y pensamiento algorítmico para el planteamiento y solución de problemas. La resolución de problemas está fundamentada en activar la curiosidad, descubrimiento y exploración en el estudiante para resolver el problema, por otra parte, se define como la adquisición de modelos de pensamiento a partir de la construcción de pensamiento declarativo y procedimental. Piaget y Vergnaud proponen que el planteamiento de problemas cotidianos o del área de dominio son formas de motivar el entendimiento e interés por resolverlos, particularmente en el área de programación básica.

Sistema Tutor Inteligente

En los cursos, la mayoría de los docentes se han visto rebasados el proceso de acompañamiento y asesoría en relación uno a uno con la cantidad de los estudiantes, además de una variedad en sus niveles de conocimientos previos que tienen los estudiantes. Los Sistemas Tutores Inteligentes (STI) surgen como una evolución de la enseñanza asistida por computadoras, al combinar conocimientos de pedagogía, psicología, ciencias cognitivas con avances en la ciencia de la computación. Usan diversas técnicas de IA en la generación de rutas de aprendizaje, selección de actividades, soporte durante las actividades, evaluación, presentación de los contenidos, entre otras. “Los sistemas inteligentes deberían proveer algunas características en función de los propósitos por los que el estudiante recurre a él, tales como: la perspectiva desde la que debe impartir los conocimientos a los alumnos, la forma de adaptación del tutor a los conocimientos previos de los alumnos y la selección de la estrategia de enseñanza más adecuada para el alumno que lo consulta” (Cataldi, Salgueiro, Lage & García-Martínez, 2005). Estos sistemas han demostrado ser efectivos en la enseñanza uno-a-uno (Murray, 1999; Chi et al., 2001).

Los STI permiten la emulación de un tutor humano para determinar qué enseñar, cómo enseñar y a quién enseñar, se plantea el diseño del STI (figura 1) a través de los siguientes módulos:

- Módulo del dominio: que define el dominio del conocimiento, proporciona los conocimientos que serán presentados en forma adecuada para que el alumno pueda adquirir las habilidades y conceptos requeridos.
- Módulo del estudiante: que es capaz de realizar un diagnóstico cognitivo del estudiante, además de actualizar el conocimiento del estudiante en cada punto durante la sesión de trabajo.
- Módulo del tutor: que genera las interacciones de aprendizaje basadas en las discrepancias entre el especialista del área de dominio y el estudiante, posee el conocimiento experto en los ámbitos de los procesos de evaluación, enseñanza-aprendizaje.
- Módulo de interface con el usuario: que permite la interacción del estudiante con el STI de una manera eficiente (conocimiento sobre cómo presentar los contenidos) (Cataldi & Lage, 2009).

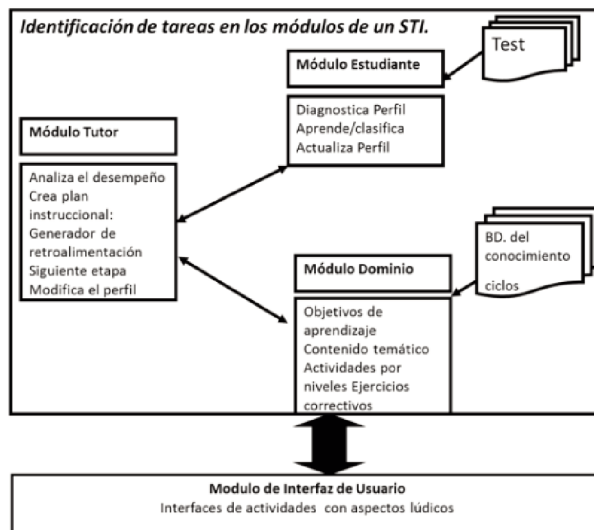


Figura 1. Diseño del Sistema Tutor. Fuente: Elaboración propia.

Se han elegido los elementos principales que alinean al proceso de desarrollo de habilidades algorítmicas en estudiantes de computación, en la propuesta del desarrollo de un STI. Los principales factores en el diseño del módulo de estudiante son el estilo de aprendizaje, las habilidades algorítmicas y la resolución de problemas como pilares fundamentales conceptualmente.

3. Metodología

Se realizó un estudio de enfoque cuantitativo, el proceso de análisis es a partir de estadística no inferencial. Se pretende conocer la caracterización del estudiante que servirá de información al STI para hacer el diseño de las actividades de aprendizaje a personalizar de acuerdo al perfil del estudiante y desarrolle las habilidades que sea necesario reforzar.

La población de estudio consiste en los estudiantes de nuevo ingreso a carreras afines a Computación. La muestra está constituida de un grupo de 28 sujetos de la Facultad de Ciencias de la Computación (FCC) de la Benemérita Universidad Autónoma de Puebla (BUAP) en México y un grupo de 20 estudiantes de la Facultad de Ingeniería Electrónica y Telecomunicaciones (FIET) de la carrera de Ingeniería en Sistemas de la Universidad del Cauca (UniCauca) en Colombia. La muestra fue por conveniencia debido a que ambos grupos ya estaban constituidos.

La tabla 1 muestra los datos de los estudiantes a los que se aplicó los test su edad es entre 18 y 20 años, con un 81% de hombres y un 19% de mujeres aproximadamente. Se resalta que se aplicó a alumnos de nuevo ingreso en su mayoría de la FIET y alumnos que ya había tomado un curso básico en la FCC.

Edad	México	Colombia
18-20 años	25	18
20-23 años	3	2
Primer curso	8	16
Segundo curso	16	4
Mujeres	15%	19%
Hombres	85%	81%
Total estudiantes	28	20

Tabla 1. Datos de la muestra de estudiantes México-Colombia. Fuente: Elaboración propia.

Instrumentos de recolección de información

En la investigación se aplicaron tres instrumentos: a) el test de estilos de aprendizaje (Felder &

Sánchez Román, G.; Guerrero García, J.; Martínez Mirón, E. A. (2019). Perfil del alumno de Computación para el diseño de un Sistema Tutor. *Certiuni Journal*, (5), 19-26.

Silverman, 1998) b) el test de Razonamiento (Lawson, 1995) y c) test de Pensamiento Computacional (Román-González, Pérez-González & Jiménez-Fernández, 2015). Los dos primeros test ya están validados, el test de pensamiento computacional está en proceso de validación y se está haciendo una adaptación de acuerdo a las necesidades de la investigación analizando las habilidades algorítmicas que el estudiante requiere desarrollar.

A continuación, se muestra una de las preguntas del test de Felder y Silverman con un total de 44 reactivos con opción de respuestas dicotómicas:

- Cuando alguien me da la dirección de un nuevo lugar, prefiero:
 - a) un mapa.
 - b) instrucciones escritas

El test de razonamiento científico consta de 12 reactivos cada pregunta tienen cuatro posibles respuestas. Se evalúan de acuerdo a tres categorías concreto donde ya deben tener desarrollado el pensamiento empírico-inductivo, transición y formal donde su pensamiento ya es hipotético-deductivo (Rodríguez, Mena & Rubio, 2010).

El tipo de pregunta es de la siguiente manera:

- Seis piezas cuadradas de madera se introducen en una bolsa de paño oscuro y se mezclan. Las seis piezas son idénticas en tamaño y forma, pero tres piezas son rojas y el resto son amarillas. Supón que alguien mete la mano dentro de la bolsa y sin mirar saca una de las piezas. ¿Cuál es la probabilidad que la pieza extraída sea roja?

- a) $1/6$ (1 posibilidad cada 6 eventos).
- b) $1/3$ (1 posibilidad cada 3 eventos).
- c) $1/2$ (1 posibilidad cada 2 eventos).
- d) 1 (1 posibilidad por cada evento).
- e) No puede ser determinado con la información dada.



El test de Pensamiento Computacional propuesto se adaptó en las dimensiones y las habilidades algorítmicas que se van a evaluar de acuerdo a la taxonomía de Bloom. Se trabaja con factores importantes en el pensamiento algorítmico (habilidades de recordar, analizar y modelar), y solución de problemas (habilidades de analizar, abstraer, modelar, evaluar), tomando en consideración dos dimensiones: habilidad de abstracción, habilidad de resolución de problemas, las cuales se han estructurado con base a la taxonomía de Bloom.

Capacidad de Abstracción	Reconocer la sintaxis de comandos.	Recordar
	Reconocer la semántica de comandos.	
	Interpretar que cambios han ocurrido cuando modificas una instrucción.	Comprender
	Traducir dado un algoritmo a un diagram de flujo o pseudocódigo.	
	Usar estructuras con un objetivo claro y conciso (decisión, ciclos).	
	Usar rutinas para desarrollar tareas específicas.	Abstraer
Capacidad de Resolución de Problemas	Modificar un algoritmo.	
	Predicción de la salida posible cuando se ejecuta un algoritmo línea por línea.	Analizar
	Detección de posibles errores en un algoritmo.	
	Evaluar los requerimientos a nivel de usuario, sistema, organización y desarrolle un algoritmo coherente que los consideren.	
	Comparar soluciones algorítmicas que resuelvan el problema de diferente manera.	Probar
	Verificar un conjunto de instrucciones a través de pruebas de escritorio.	
	Generar algoritmos completos basados en la especificación del problema.	Crear
	Generalizar un algoritmo.	

Tabla 2. Dimensiones a evaluar de las habilidades algorítmicas con base a la taxonomía de Bloom. Fuente: Elaboración propia.

Se propone utilizar el Test de Pensamiento Computacional (Román-González, Pérez-González & Jiménez-Fernández, 2015) para lograr identificar el nivel de habilidades algorítmicas (tabla 2) sobre su conocimiento previo. El test consta de 28 preguntas gráficas a responder a partir de cuatro opciones que el estudiante debe elegir. Se plantean tres categorías novato, aprendiz y experto para la evaluación de acuerdo al número de preguntas en cada categoría de la taxonomía de Bloom, así, también se identifica exactamente en qué habilidad necesita reforzar su aprendizaje el estudiante. El tipo de preguntas se ve en la figura 2 a continuación:

¿Qué ordenes llevan al Pac-Man hasta el fantasma por el camino señalado?

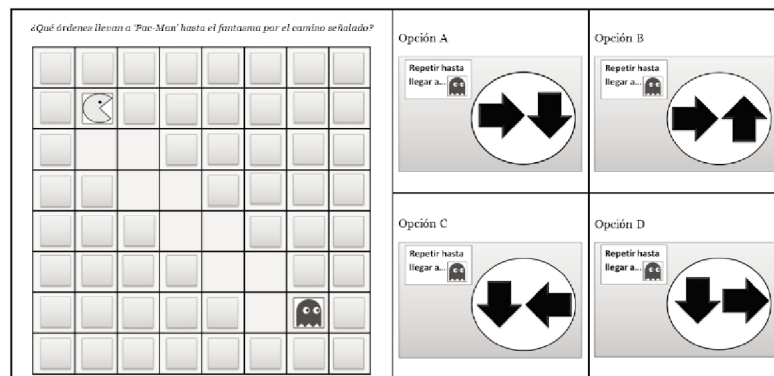


Figura 2. Pregunta del test de Pensamiento Computacional. Fuente: (Román-González, Pérez-González & Jiménez-Fernández, 2015).

La aplicación de los tres instrumentos se realizó en 2 sesiones de 45 minutos, en una sesión se aplicó el test de estilos de aprendizaje y de razonamiento científico, en otra sesión se aplicó el test de pensamiento computacional el cuál se aplicó en el laboratorio a través de un formulario en línea.

4. Resultados del Estudio comparativo

A continuación, se presentan los resultados del test de estilos de aprendizaje y del test de razonamiento; finalmente el análisis de los datos del test de pensamiento computacional. Se realiza la comparación de los datos obtenidos en los dos grupos que nos dan un panorama de las características de los estudiantes de computación en nivel superior.

El test de Felder y Silverman se aplica en línea a los dos grupos de acuerdo a los cuatro pares dicotómicos: activo-reflexivo, sensorial-intuitivo, visual-verbal, secuencial-global como se observa en la figura 3, el grupo de la Facultad de Ciencias de la Computación (México) tienden a ser activos, sensoriales, visuales y globales, y en el grupo de la Facultad de Ingeniería Electrónica y Telecomunicaciones (Colombia) tiene una mayor tendencia ser activos, sensoriales, visuales y secuenciales. La coincidencia de ambos grupos en la forma de aprender es mayormente activa, sensorial y visual. La información obtenida apoya en el diseño del material para integrar al Sistema Tutor que definirá el modelo para desplegar los materiales más adecuados de acuerdo al estilo predominante en que el estudiante procesa, gestiona y recupera la información.

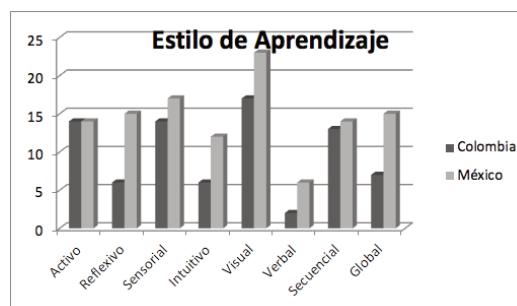


Figura 3. Resultado del test de estilo de aprendizaje. Fuente: Elaboración propia.

En el test de razonamiento que se aplicó al grupo de estudiantes de la FCC (México) obtuvimos que un 73% está en transición, un 8% en el nivel concreto y un 9% en nivel formal (tabla 3). Los resultados del grupo de la FIET (Colombia) se muestran en la tabla 3, donde los datos son muy similares dado que se tiene un 10% en nivel formal, un 20% en nivel concreto y un 70% en transición. Por lo tanto, los estudiantes requieren desarrollar su nivel concreto o transición y pasar al nivel formal, este nivel se caracteriza por que ya pueden abstraer información de forma deductiva a partir de problemas planteados.

Test Razonamiento	Mexico	Colombia
Concreto	18%	20%
Transición	73%	70%
Formal	9%	10%

Tabla 3. Comparación de la aplicación del test de razonamiento. Fuente: Elaboración propia.

Después de aplicar el instrumento de Pensamiento Computacional a los estudiantes de la FCC (México) como se puede observar en la tabla 4, en las primeras tres habilidades recordar, comprender y abstraer el nivel que tienen los estudiantes se refiere en mayor porcentaje al experto, un menor porcentaje se encuentran en nivel de novato; sin embargo, en las dos últimas habilidades de analizar y probar el porcentaje de experto disminuye considerablemente y repunta el aprendiz.

Test Pensamiento Computacional	Experto		Aprendiz		Novato	
	México	Colombia	México	Colombia	México	Colombia
Recordar	40.9%	70%	45%	20%	9%	10%
Comprender	77%	80%	23%	15%	0%	5%
Abstraer	0%	55%	82%	45%	18%	0%
Probar	4.5%	10%	50%	55%	45%	35%

Tabla 4. Comparación de la aplicación del test de Pensamiento Computacional. Fuente: Elaboración propia.

Para el caso de los estudiantes de Colombia, ellos tienen mayor dominio en las habilidades de recordar, comprender y abstraer, e igualmente disminuyen en las habilidades de analizar y probar al nivel de experto, sin embargo, aumenta considerablemente el nivel de aprendiz.

Los resultados obtenidos de la aplicación de los test para la caracterización indican que los tres instrumentos que se proponen nos aportan información acerca de la disposición del estudiante en el aprendizaje y a desarrollar sus habilidades algorítmicas. La ventaja de proponer esta calificación de habilidades da un aporte en qué nivel está y qué habilidad es necesario trabajar de tal manera que en el modelo de estudiante se pueda integrar esta información y dar un seguimiento más preciso.

Existen varios trabajos de STI que modelan al estudiante, pero ninguno ha tomado en consideración el razonamiento científico y las habilidades algorítmicas en el nivel superior inicial. Lo que propone el trabajo es perfilar las características que pueden modelar al estudiante identificando el nivel que tiene el estudiante de nuevo ingreso para adoptar estrategias y actividades de aprendizaje que se integren al sistema tutor a fin de dar un seguimiento personal y autónomo de acuerdo al progreso del estudiante.

5. Conclusiones

Los datos que se vierten en las gráficas presentan la forma en que estudiantes en las dos regiones de Latinoamérica prefieren el estilo de aprendizaje en el área de Computación con mayor predominancia a ser: activos, sensoriales, visuales y secuenciales; en el caso del razonamiento científico ambos grupos mostraron similares porcentajes centrados principalmente en el nivel de transición y que define que hay que reafirmar el conocimiento y resolución de problemas en los estudiantes de nuevo ingreso. Por último, en el test de Pensamiento Computacional las tendencias de los estudiantes para las habilidades: recordar, comprender y abstraer, los estudiantes en ambos grupos definen un nivel de experto, sin embargo, en las habilidades de categorías superiores (analizar y probar) de la taxonomía de Bloom los estudiantes aún siguen en nivel de aprendiz.

Por lo tanto, se considera que los resultados que se encontraron en ambos grupos son similares en cuanto al tipo de estudiantes que se tienen y sus carencias en el dominio de las habilidades algorítmicas necesaria para el área de Ciencias de la Computación. Los datos abonan a definir la caracterización general de los estudiantes de nuevo ingreso al área de Computación, en particular para la generación del modelo de estudiante a partir de tres factores el razonamiento, el estilo de aprendizaje y las habilidades algorítmicas; la clasificación que se maneja permite decir en qué nivel está y cuál es la habilidad que requiere mayor enfoque en un primer diagnóstico que se integra en el modelo de estudiante y dar un seguimiento en el proceso de diseño del STI. La investigación abona al área educativa puntualmente al área de instrucción en el aprendizaje de las habilidades básicas de programación detectando en particular que necesita fortalecer el estudiante.

Cómo citar este artículo / How to cite this paper

Sánchez Román, G.; Guerrero García, J.; Martínez Mirón, E. A. (2019). Perfil del alumno de Computación para el diseño de un Sistema Tutor. *Certiuni Journal*, (5), 19-26. (www.certiunijournal.com)

Referencias

- Cataldi, Z.; Lage, F. (2009). Sistemas Tutores Inteligentes Orientados a la Enseñanza para la comprensión. *EduTec Revista Electrónica de Tecnología*. (http://edutec.rediris.es/Revelec2/revelec28/articulos_n28_pdf/EduTec-E_Cataldi_Lage_n28.pdf)
- Cataldi, Z.; Salgueiro, F.; Lage, F.; García-Martínez, R. (2005) Sistemas Tutores Inteligentes. Los Estilos del Estudiante para Selección del Tutorizado. In *Proceedings del VII Workshop de Investigadores en Ciencias de la Computación. WICC 2005* (pp. 66-70).
- Chi, M.; Siler, S.; Jeong, H.; Yamauch, T.; Hausmann, R. (2001). Learning from human tutoring. *Cognitive Science*, 25, 471-533.
- Felder, M.; Silverman, K. (1988). Learning and Teaching Styles in Engineering Education, Institute for the Study of Advanced Development. *Journal Enginner Education*, 78(7), 674-681.
- Franzoni-Velázquez, A. L.; Cervantes-Pérez, F.; Assar, S. (2012). A Quantitative Analysis of Student Learning Styles and Teacher, Teachings Strategies in a Mexican Higher Education Institution, *Journal of Applied Research and Technology*, 10(3), pp. 289-308.
- Futschek, G. (2006). Algorithmic Thinking: The Key for Understanding Computer Science. In *ISSEP* (159-168).
- Lawson, A. E. (1995). *Science teaching and the development of thinking*. Belmont, CA: Wadsworth.
- Moroni, N.; Señas, P. (2005). Estrategias para la enseñanza de la programación. In *Primeras Jornadas de Educación en Informática Y TICS en Argentina* (pp. 254-258). Bahía Blanca, Argentina.
- Murray, T. (1999). Authoring Intelligent Tutoring Systems: An Analysis of the State of the Art. *Int. J. of Artificial Intelligence in Education*, 19, 98-129.
- Rodríguez, M. D.; Mena, D. A.; Rubio, C. M. (2010). Razonamiento Científico y Conocimientos Conceptuales de Mecánica: Un Diagnóstico de Alumnos de Primer Ingreso a Licenciaturas en Ingeniería. *Formación Universitaria*, 3(5), 37-46. doi: 10.4067/S0718-50062010000500006
- Román-González, M.; Pérez-González, J. C.; Jiménez-Fernández, C. (2015). Test de Pensamiento Computacional : diseño y psicometría general. In *III Congreso Internacional sobre Aprendizaje, Innovación y Competitividad (CINAIC 2015)*. doi: 10.13140/RG.2.1.3056.5521.
- Rosano, Z.; Paur, A. (2006). Estrategias para la enseñanza de Algorítmica y Programación. In *TE&ET'06* (pp. 117-124).

