

Desarrollo de Pensamiento Computacional en Estudiantes de Ingeniería

EJE 4. INNOVACIONES

Enfoques y estrategias innovadoras en la enseñanza universitaria en distintos campos de conocimiento.

Jiménez Rey, Elizabeth¹
Servetto, Arturo C.¹
Calvo, Patricia M.¹

¹ Facultad de Ingeniería UBA, Argentina, ejimenezrey, aservetto, pmcalvo@fi.uba.ar

RESUMEN

Se describe estrategias de enseñanza para el desarrollo del pensamiento computacional en alumnos de carreras de Ingeniería en la FIUBA, en cursos de una asignatura denominada Computación correspondientes a los autores, y con diversos condicionamientos desfavorables. Se trata de una asignatura de formación básica que los alumnos de carreras de Ingeniería pueden cursar al ingresar a la facultad luego de aprobar el Ciclo Básico Común, en simultaneidad con otras asignaturas de Ciencias Básicas muy demandantes de tiempo y esfuerzo como Álgebra, Análisis Matemático y Física. El pensamiento computacional es un paradigma para resolver problemas que en

general resulta extraño o está poco o mal desarrollado en los estudiantes, que además consideran a la asignatura como ajena a sus intereses profesionales, por lo que resulta un desafío para los docentes despertar y mantener la motivación de los alumnos y evitar que deserten. El objetivo de las estrategias de enseñanza y las modalidades de desarrollo y evaluación de actividades formativas que se describen en este trabajo es desarrollar las competencias específicas y genéricas que tiene la asignatura como objetivos propios, propiciando el interés de los alumnos y su permanencia en los cursos.

PALABRAS CLAVE: Algoritmia y programación, Pensamiento computacional, Actividades formativas, Evaluación continua.

1. INTRODUCCIÓN

El aprendizaje de los contenidos de la asignatura Computación como ciencia básica en la Facultad de Ingeniería de la Universidad de Buenos Aires FIUBA se focaliza en contribuir a desarrollar en los estudiantes de las carreras de Ingeniería en general la capacidad de construir la solución de problemas con la computadora (mediante el ingenio de algoritmos y su codificación como programas). El mayor desafío cognitivo al que se enfrentan los alumnos es el descubrimiento de un

algoritmo efectivo (eficaz y eficiente) que solucione el problema planteado. (Jiménez Rey, Aveleyra, Barranquero, 2019)

En términos generales, sus objetivos son:

- Que el alumno comprenda los fundamentos de la tecnología actual y la forma en que influye en el diseño de computadoras.
- Que el alumno comprenda cómo el software de sistema controla el funcionamiento de una computadora y establece las rutas de comunicación básica entre una máquina y las personas que la usan, y cómo pueden conectarse computadoras para compartir información y recursos.
- Capacitar al alumno en el descubrimiento y representación de algoritmos como programas aplicando el principio de dividir y conquistar y refinamientos sucesivos, e inculcarle la búsqueda de eficiencia y la comprobación de corrección y completitud.
- Capacitar al alumno en la comunicación de algoritmos a las computadoras mediante el proceso de programación en lenguaje Python, en el diseño de datos para representar información y manipularlos con la mayor eficiencia, y en la presentación efectiva de información al usuario.

1.1 Caracterización de la asignatura

En la FIUBA es obligatoria para los estudiantes de las ingenierías no informáticas: Electricista, Industrial, Química, Mecánica, Naval, de Alimentos, en Agrimensura, en Petróleo. Es cuatrimestral, es decir, se desarrolla en 16 clases, 1 por semana, de 4 horas de duración. Tiene como asignatura correlativa posterior Análisis Numérico I, y ninguna correlativa anterior. La cursan los estudiantes que recién ingresan a la FIUBA, en general, luego de haber aprobado todas las asignaturas del Ciclo Básico Común CBC, y en un contexto curricular de alta concentración de asignaturas de ciencias básicas.

1.2 Competencias que deben desarrollar los estudiantes

Competencias generales de Ingeniería. Conocimiento de tecnologías y métodos básicos que los capaciten para el aprendizaje de nuevos métodos y tecnologías, así como que los doten de versatilidad para adaptarse a nuevas situaciones. Capacidad de resolver problemas con iniciativa, toma de decisiones, creatividad, y de comunicar y transmitir conocimientos, habilidades y destrezas, comprendiendo la responsabilidad ética y profesional de la actividad del Ingeniero.

Competencias generales de Informática. Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la Ingeniería.

Competencias específicas de la asignatura. A continuación, se enuncian las competencias cognitivas (saber), procedimentales o instrumentales (saber hacer) y actitudinales (saber ser o saber actuar), que los autores se proponen que los alumnos desarrollen, y por sinergia también las generales destacadas antes, fundamentalmente mediante la resolución interactiva de problemas en el aula real y la consulta de material de estudio (clases invertidas) y desarrollo de actividades prácticas formativas en el aula virtual (cursos virtuales en plataforma Moodle). Las actividades de resolución de

problemas en forma cooperativa o colaborativa también coadyuvan al desarrollo de las competencias genéricas que se detallan luego.

Cognitivas. Conocimiento de vocabulario técnico que facilite la comunicación con profesionales de la Informática para el trabajo interdisciplinar. Conocimiento general sobre algoritmia como paradigma de resolución de problemas. Conocimientos básicos de la sintaxis de un lenguaje de programación de alto nivel. Conocimiento de estructuras de datos fundamentales. Conocimientos básicos de complejidad computacional.

Procedimentales/Instrumentales. Capacidad para descomponer un problema real para su posterior codificación en un programa. Capacidad para documentar programas con claridad y sencillez. Capacidad para analizar la complejidad de algoritmos. Capacidad para interpretar y utilizar las diferentes estructuras de control y de datos para implementar soluciones a problemas específicos. Capacidad para comprender documentación técnica y reutilizar código desarrollado por terceras partes.

Actitudinales. Capacidad de resolución de problemas mediante algoritmos. Motivación por la claridad, sencillez y eficiencia en la resolución de problemas. Capacidad para debatir y concluir las distintas soluciones a un problema.

Competencias transversales o genéricas. Capacidad para la autoorganización y planificación del trabajo individual y del proceso de aprendizaje. Capacidad para el trabajo en grupo. Capacidad de análisis y síntesis. Motivación por la calidad del resultado.

1.3 Caracterización de los estudiantes

"En general, los alumnos que cursan Computación son estudiantes ingresantes que evidencian serias dificultades para realizar abstracciones, así como hábitos de aprendizaje mecánico y memorístico, y dificultades para analizar problemas y articular estrategias para resolverlos." (Jiménez Rey, 2019)

Invariablemente se verifica una alta deserción en las primeras semanas de cursado, debido a la mencionada concentración de asignaturas de ciencias básicas que los alumnos cursan en simultáneo, que hace que, ante la demanda creciente de tiempo y esfuerzo, prioricen asignaturas que por sus correlatividades posteriores, de no aprobarlas, pueden frenar el avance en sus respectivas carreras.

Por otra parte, al comenzar los cursos se suele observar en los alumnos escasa motivación o interés por los contenidos de la asignatura, por implicar un nuevo paradigma de resolución de problemas que les representa un importante desafío, y por no visualizar o comprender la afinidad con sus campos profesionales.

2. DESCRIPCIÓN DE LA EXPERIENCIA / DESCRIPCIÓN DEL DESARROLLO DE LA INVESTIGACIÓN

Se presenta los enfoques pedagógicos en los distintos cursos.

2.1 Enseñanza

La estrategia pedagógica utilizada para el desarrollo de contenidos en dos cursos de la primera autora se sintetiza en la experiencia de pensar para crear juntos. La acción docente integra enseñar a pensar y enseñar contenidos en aula real (Laboratorio) y en aula virtual (Google Drive Institucional). En aula real, los estudiantes piensan con sus pares y el profesor (gran grupo) para construir el conocimiento nuevo (nueva herramienta de programación), y en aula virtual, los estudiantes piensan con sus compañeros de taller (pequeño grupo) para ingeniar los algoritmos que solucionan los problemas propuestos con las nuevas herramientas de programación acompañados por el profesor quien interviene para modelar el pensamiento.

Una forma específica de pensar, el pensamiento computacional (Zapata Ros, 2015), propicia el análisis y la relación de ideas para la organización y la representación lógica de procedimientos y favorece las competencias computacionales. En el proceso de enseñanza (Jiménez Rey, 2019) se promueve el desarrollo de este pensamiento específico mediante el análisis del problema y el diseño de algoritmos (Fases Análisis y Diseño) durante el proceso evolutivo de construcción de programas como una competencia clave de aprendizaje en la formación de los estudiantes de ingeniería; no se reduce el aprendizaje al hallazgo de una secuencia de instrucciones de programa (Fases Codificación y Evaluación) para solucionar problemas de dificultad progresiva. La Algoritmia y la Programación no se aíslan, sino que se distinguen y se conjugan.

La práctica educativa es activa y reflexiva y está centrada en hacer visible el pensamiento de los alumnos de una manera sistematizada y formalizada mediante el uso de tablas (Jiménez Rey, 2019): Tabla 1 Definición de Recursos, Tabla 2 Análisis del Problema, Tabla 3 Diseño de la Solución, Tabla 4 Modelo de Programa Tipo. Se proporciona a los estudiantes un andamiaje para ingeniar algoritmos y hacer visible el pensamiento algorítmico de manera que el profesor pueda intervenir y guiar el proceso de diseño para el aseguramiento de la calidad de la solución en la construcción de un programa (Jiménez Rey, Aveleyra y Barranquero, 2019).

El pensamiento visible se focaliza en tres prácticas básicas (Proyecto Cero de la Escuela de Educación de Harvard, s. f.): las rutinas de pensamiento, la documentación del pensamiento del estudiante y la práctica profesional reflexiva. Una rutina de pensamiento promueve movimientos de pensamiento específicos e implica un proceso que al ponerse en práctica permite que el pensamiento de los estudiantes se haga visible a medida que expresan sus ideas, debaten y reflexionan en torno a ellas. Las rutinas de pensamiento se consideran desde tres perspectivas: como herramientas, como estructuras y como patrones de comportamiento (Ritchhart, Church, y Morrison, 2014). Tanto en aula real como en aula virtual se utilizan fundamentalmente, entre otras: la rutina Conectar-Ampliar-Desafiar para sintetizar y organizar ideas porque ofrece una estructura en la que el pensamiento de los estudiantes inspirado por una nueva situación problemática se puede hacer visible y porque permite crear oportunidades para que los estudiantes puedan escudriñar con una mirada pensante lo que necesitan comprender, y la rutina Afirmar-Apoyar-Cuestionar para explorar las ideas más profundamente porque ofrece una estructura abarcadora para examinar las ideas y generar nueva comprensión (Jiménez Rey, Aveleyra y Barranquero, 2019).

La herramienta tecnológica Google Drive Institucional soporta el pensamiento visible y el pensamiento computacional que se despliega en el descubrimiento de un algoritmo solución y provee una plataforma estable y flexible que potencia la actividad conjunta desplegada por el profesor

responsable y los estudiantes en torno al contenido de aprendizaje en el aula virtual e impacta sobre el control de la calidad de diseño del algoritmo en el proceso de construcción de un programa (Jiménez Rey, 2019).

En los cursos de los otros autores se desarrollan actividades formativas tanto en el aula real como en el aula virtual (esto último a través de la plataforma Moodle). El énfasis se pone en el desarrollo de la autonomía para la adquisición de conocimientos (por ejemplo, a través de la implementación de aula invertida), en el desarrollo de la proactividad, clave para el futuro desarrollo profesional, y en la autorregulación.

En cuanto a la utilización del aula virtual (a través de la plataforma Moodle), las actividades se plantean colaborativas, y están orientadas a fomentar la metacognición, el autoanálisis, la regulación, reflexión y la concientización para adquirir conocimientos significativos. En este ámbito la tarea docente se orienta a la tutorización, facilitación y provisión de oportunidades a los estudiantes, con un rol de mediador y una intervención pedagógica acotada en función de las necesidades de los estudiantes.

2.2 Aprendizaje

El aprendizaje en los cursos de los autores atiende no solamente el conocimiento de los alumnos de la programación de computadoras sino también el rol que cumple la enseñanza de la asignatura en la formación en ciencias básicas de los estudiantes y en el desarrollo de la capacidad de resolver problemas en Ingeniería mediante algoritmos y programas que permitan ejecutarlos en la computadora (Jiménez Rey, Calvo y Servetto, 2018). Es necesario enseñar a pensar de forma eficiente. El pensamiento eficiente (Swartz et al, 2008) “se refiere a la aplicación competente y estratégica de destrezas de pensamiento y hábitos de la mente productivos que nos permiten llevar a cabo actos meditados de pensamientos, como tomar decisiones, argumentar y otras acciones analíticas, creativas o críticas”.

El desarrollo de dos cursos de la primera autora se divide en 2 módulos (8 clases por módulo, el primero básico y el segundo integrador). En cada módulo, las 3 primeras clases están destinadas a introducir el conocimiento de una nueva herramienta de programación en aula real (Laboratorio). El profesor y todos los participantes de la clase despliegan juntos el pensamiento computacional utilizando tablas, ingenian un algoritmo que solucione el problema planteado y codifican el algoritmo en lenguaje Python usando otra tabla. Durante las 3 primeras semanas y en correspondencia con cada clase presencial, los alumnos en pequeños grupos de 3 integrantes (talleres propios) desarrollan Actividades Grupales Formativas AGF en el aula virtual (plataforma Google Drive Institucional). Los estudiantes ingenian algoritmos utilizando tablas para solucionar los problemas propuestos orientados por el profesor quien interviene para guiar el pensamiento, y codifican los algoritmos en lenguaje Python usando otra tabla. A partir de la clase 4, los estudiantes desarrollan el Trabajo Grupal Formativo TGF integrador en aula virtual, ingenian algoritmos que solucionan los problemas integradores asistidos por el profesor, los codifican y entregan los programas a través del sitio del curso en el campus virtual FIUBA en clase 7. En el Análisis del Problema y el Diseño de la Solución (fases algorítmicas) los estudiantes utilizan las Tablas 1, 2 y 3 (Un Lugar para Algoritmiar) y, en la Codificación del Algoritmo y la Evaluación del Programa (fases de programación) utilizan la Tabla 4 (Un Lugar para Programar) en el entorno IDLE Python para desarrollar las AGF y los TGF.

En los cursos de los demás autores, durante las primeras diez clases en el aula real se presenta problemas (que se eligen considerando temas que despierten la motivación de los alumnos, asociados, en la medida de lo posible, al 'mundo real', a la resolución de desafíos o a situaciones lúdicas) para cuya resolución se requiere un algoritmo adecuado. El diseño algorítmico implica la modelización de la situación problemática utilizando diversas herramientas (gráficos y representaciones diversas) que pueden ser propuestas tanto por los estudiantes como por los docentes, en una primera etapa de comprensión de la situación.

Luego, generalmente de forma conjunta (alumnos orientados por el docente o trabajando en pequeños grupos), se llega a la invención de una secuencia resolutoria que se traducirá finalmente en un programa. Para el desarrollo del algoritmo hasta llegar a su versión final se aplica una estrategia de refinamientos sucesivos, de modo de reducir la cantidad de detalles a considerar en cada nivel. Así el estudiante enfoca su atención en los aspectos relevantes de cada uno de esos niveles, y adquiere paulatinamente la habilidad de establecer cuáles son esos aspectos. Una vez codificado el algoritmo, el proceso concluye con las pruebas de eficacia y por lo general con una reflexión conjunta sobre el saber incorporado.

Las etapas de motivación inicial, planificación, ejecución de las acciones resolutorias y la posterior autocorrección del proceso de aprendizaje (en cuanto estrategias, y acciones cognitivas), en los procesos grupales son especialmente significativas para lograr la autorregulación del aprendizaje, en conexión con el desarrollo de la autonomía intelectual.

Las siguientes seis clases en aula real se orientan a la resolución grupal de problemas con la asistencia de los docentes, pero con un mayor grado de libertad por parte de los alumnos para la exploración grupal autónoma de soluciones.

La utilización del aula virtual habilita el desarrollo de trabajos grupales en la modalidad de trabajo colaborativo, en los cuales el diseño de interacción para alcanzar los objetivos surge desde el interior del grupo, siendo fundamental la responsabilidad de cada integrante por sus acciones y su aprendizaje; de este modo el grupo autogestiona la coordinación de actividades (Cardozo Cardona, 2010).

2.3 Evaluación

En los dos cursos de la primera autora la evaluación se desarrolla en un proceso continuo integrado a la enseñanza, a la que acompaña y apoya, y cumple una función formativa porque regula la acción pedagógica con el fin de mejorar el desarrollo de las actividades de profesores y estudiantes durante el curso (Davini, 2009). Se evalúa el aprendizaje como búsqueda de evidencias de los conocimientos y las habilidades adquiridos y como juicio de valor por comparación con el referente construido en base a criterios, es decir, establecidos con relación a objetivos a ser logrados e indicadores de un determinado dominio a ser alcanzado (Mazza, 2019):

1 Funcionamiento del Programa FDP. El alumno debe lograr un programa eficaz, es decir, un programa que informe los resultados esperados, que cumpla con los requerimientos del enunciado y que establezca una comunicación clara y precisa con el usuario en la solicitud de datos y en la exhibición de resultados.

2 Metodología de Desarrollo MDD. El alumno debe lograr un programa inteligible, es decir, un

programa que facilite a un lector-programador la comprensión de la manera en que se resolvió el problema (diseño algorítmico) con el programa mediante la apropiada descomposición del problema en subproblemas y la descripción sintética de los subproblemas que resuelven las instrucciones del programa.

3 Calidad de Diseño CDD. El alumno debe lograr un programa eficiente, es decir, un programa que permita resolver cada uno de los subproblemas mediante el descubrimiento de su naturaleza y la elección de la herramienta de programación apropiada para implementar la solución.

La evaluación permite al profesor medir la eficiencia de las acciones pedagógicas y regular la enseñanza, y a los estudiantes, poner a prueba sus conocimientos y habilidades disciplinares y regular su aprendizaje. El profesor utiliza los resultados de la evaluación para detectar las principales dificultades de aprendizaje, testear la calidad del aprendizaje en el momento de la evaluación en correspondencia con los objetivos de enseñanza y emprender acciones que posibiliten remediar y/o mejorar la enseñanza y el aprendizaje. Los estudiantes utilizan los resultados de la evaluación para detectar sus fortalezas y debilidades en la aprehensión de los contenidos, visibilizar cómo proceder para satisfacer las demandas de aprendizaje y reflexionar sobre cómo están aprendiendo e intervenir en la regulación de su aprendizaje mediante acciones remediales.

En la evaluación de las AGF y de los TGF se valora la Responsabilidad Grupal en el desarrollo de los programas que solucionan los problemas propuestos y la Responsabilidad Individual implicada en las intervenciones y aportes de cada integrante del taller en el despliegue del pensamiento computacional en las tablas. Las AGF se evalúan en los aspectos FDP y MDD mediante una devolución detallada escrita y se considera la entrega en tiempo y forma de la actividad grupal y la calidad de la participación individual en el descubrimiento del algoritmo; el aspecto CDD se corrige en un foro de discusión en aula real en el cual los participantes despliegan el pensamiento crítico para valorar las soluciones propias y las de los otros grupos. Los TGF se evalúan en los aspectos FDP, MDD y CDD en una entrevista grupal y devolución oral en aula real antes de la próxima instancia de evaluación individual. La calificación de las actividades y trabajos formativos grupales es conceptual y cualitativa.

Las evaluaciones parciales prácticas individuales escritas (clase 8, fin módulos 1 y 2) consisten en la solución de un problema original con la construcción en papel de un programa que implique un desafío cognitivo que interpele al estudiante con relación al logro de las competencias algorítmicas. La evaluación tiene una función formativa y sumativa, arroja un balance al final de cada módulo y determina el resultado total final del curso (Davini, 2009). La calificación de los parciales prácticos individuales es cuantitativa (requerimiento institucional).

El segundo y tercer autores establecen instancias evaluativas en hitos del desarrollo de sus cursos que consideran más oportunos en cuanto a temas y desarrollo de competencias a evaluar (evaluación continua). Así es que, para cada tema de relevancia en los contenidos se establecen evaluaciones individuales escritas o en máquina (en estas últimas se considera tanto el diseño algorítmico como la codificación y documentación). Se busca situar estas instancias - como ha señalado Litwin (Litwin, 1998), en las oportunidades más convenientes, haciendo foco en los problemas, las dificultades o los hallazgos para habilitar la extracción de inferencias sobre el proceso de aprendizaje.

La evaluación formativa es un proceso que acompaña todo el trayecto de aprendizaje. Se propende, a través de diversos recursos (entre los cuales está el diálogo didáctico), a determinar si se ha internalizado un concepto adecuadamente más que a realizar mediciones cuantitativas.

Los trabajos grupales colaborativos elaborados a través del aula virtual (plataforma Moodle) se evalúan considerando la cantidad y calidad de las participaciones de los miembros de los respectivos grupos, y la evolución de la producción (dado que la utilización de herramientas tales como foros grupales habilitan el mantenimiento de sucesivas versiones del producto desarrollado). Asimismo, se propone a los grupos que lleven a cabo una instancia de reflexión acerca de la experiencia grupal.

Las últimas clases en aula real están dedicadas a la realización de un trabajo grupal, cuya temática elige cada grupo y que debe tener objetivos previamente aceptados por los docentes. En este trabajo final se integran los saberes adquiridos durante el curso; los integrantes de cada grupo hacen una defensa de este en una instancia final de evaluación.

3. CONCLUSIONES

Para evidenciar el rendimiento de los estudiantes participantes con relación a las estrategias de enseñanza implementadas, en la Tabla 1 se presenta los resultados obtenidos en los cursos de los autores en las principales instancias de evaluación propuestas.

Tabla 1 (Evaluaciones)

Aspecto	Cursos Primera Autora		Curso Segundo Autor	Curso Tercera Autora
	Curso 1	Curso 2		
Cupos	32	32	40	40
Inscriptos	33	32	43	49
Evaluados (parcial y actividades formativas)	28	25	28	47
Aprobados	18 (64,3%)	17 (68%)	25 (89,3%)	31 (66%)
Evaluados (clase 16)	20	21	22	47
Aprobados	20	20	22	46

El enfoque por competencias en el nivel universitario implica ensayar estrategias de enseñanza y de evaluación que permitan dar cuenta de las competencias adquiridas por los estudiantes en toda su complejidad. Para esto, el estudiante debe ser copartícipe en todo el proceso de enseñanza, aprendizaje y evaluación, teniendo no sólo responsabilidad por sí mismo sino también por sus pares.

La enseñanza centrada en el alumno exige la adopción de estrategias de enseñanza que rompen con el esquema tradicional de disociación de teoría y práctica, así como de sistemas de evaluación

formativa y continua. Los autores aprovechan el aula real para el desarrollo de actividades formativas que permitan verificar el desarrollo de competencias, e instrumentan sistemas de evaluación que fragmentan las instancias acotándolas a temas puntuales; esto permite una evaluación más profunda de cada tema, y la distribución en el tiempo reduce la concentración de evaluaciones en marcos temporales reducidos, y favorece la motivación y el compromiso constante por parte de los alumnos alentando la permanencia.

BIBLIOGRAFÍA

Arancibia, M., Oliva, I., Paiva, F. (2014). *Meaning Processes mediated through a Protagonists Collaborative Learning Platform*. Recuperado de <http://eprints.rclis.org/21080/7/c4207>

Cardozo Cardona, J. (2010). TIC y educación. Los aprendizajes colaborativos como estrategia para los procesos de construcción de conocimiento. *Revista Educación y Desarrollo Social*. ISSN: 2011-5318 | e-ISSN: 2462-865. Recuperado de <http://revistas.unimilitar.edu.co/index.php/reds/article/view/911/665>

Davini, M. C. (2009). *Métodos de enseñanza*. Buenos Aires: Santillana. ISBN 978-950-46-1910-9

Jiménez Rey, E. (2019). Computación en Ingeniería: La experiencia de pensar para solucionar problemas con algoritmos y programas en aula real y aula virtual. *48° JAIIO Jornadas Argentinas de Informática*. Anales de SAEI 2019 (pp. 1-16). Simposio Argentino de Educación en Informática. Universidad Nacional de Salta. Salta, Argentina. Recuperado de <http://48jaiio.sadio.org.ar/Anales/Saei/Contribuciones>

Jiménez Rey, E., Aveleyra, E., Barranquero, F. (2019). Competencias en Algoritmia y Programación como Formación Básica en Ingenierías: el Rol del Pensamiento Visible y la Mediación Tecnológica. *IV CIECIBA Congreso Internacional de Enseñanza de las Ciencias Básicas*. Actas de Resúmenes (pp. 14-15). Paysandú, Uruguay. Recuperado de: <http://cieciba.multisitio.interior.edu.uy/presentacion-de-trabajos/>

Jiménez Rey, E., Calvo, P., Servetto, A. (2018). Sistematización del análisis como estrategia didáctica para el diseño de algoritmos en la formación básica de los ingenieros. *III CIECIBA Congreso Internacional de Enseñanza de las Ciencias Básicas*. Universidad Tecnológica Nacional. Concordia, Entre Ríos, Argentina.

Litwin, E. (1998). La evaluación: campo de controversias y paradojas o un nuevo lugar para la buena enseñanza. En A.de Camilloni, S. Celman, E. Litwin, M. Palou de Maté (Eds.). *La evaluación de los aprendizajes en el debate didáctico contemporáneo* (pp. 11-33). Buenos Aires: Paidós.

Mazza, D. (2019). Taller sobre prácticas de evaluación en FIUBA. Subsecretaría de Planificación de Grado, Innovación Educativa y Formación Docente. Facultad de Ingeniería. Universidad de Buenos Aires.

Panadero, E., Alonso Tapia, J. (2014). ¿Cómo autorregulan nuestros alumnos? Revisión del modelo cíclico de Zimmerman sobre autorregulación del aprendizaje. *Anales de Psicología*, vol. 30, num. 2, (pp. 450-462). Recuperado de <http://www.redalyc.org/articulo.oa?id=16731188008>

Proyecto Cero de la Escuela de Educación de Harvard (s. f.): Recuperado de <http://www.pz.harvard.edu/>

Ritchhart, R., Church, M., Morrison, K. (2014). *Hacer visible el pensamiento*. Buenos Aires: Paidós.

Swartz, R., Costa, A., Beyer, B., Reagan, R., Kallick, B. (2008). La importancia del pensamiento eficaz. En *El aprendizaje basado en el pensamiento. Cómo desarrollar en los alumnos las competencias del siglo XXI* (pp. 15-47). Nueva York: Ediciones SM.

Zapata Ros, M. (2015). Pensamiento Computacional: Una nueva alfabetización digital. *Revista de Educación a Distancia*, 46(4). DOI: 10.6018/red/46/4. Recuperado de <https://www.um.es/ead/red/46/zapata.pdf>