



Universidad Nacional de la Patagonia Austral

Dirección de Postgrado

**“Metodología de seguimiento de los estudiantes de Ingeniería Eléctrica
de la Facultad de Ciencias Exactas y Naturales y Agrimensura de la UNNE
para evaluar el grado de avance en el desarrollo del pensamiento
computacional y la programación”**

LUCIA SALAZAR

Tesis para optar al título de Magister en Educación en Entornos Virtuales

Director de Tesis: Dr. Miguel Edgardo Prado Lima

Co-Directora: Magister María Eugenia Valesani

Río Gallegos, 2023

Índice

Universidad Nacional de la Patagonia Austral.....	1
Dirección de Postgrado.....	1
DEDICATORIA.....	5
AGRADECIMIENTOS.....	6
INDICE DE FIGURAS.....	7
INDICE DE TABLAS.....	8
INDICE DE GRAFICOS.....	8
Introducción.....	9
CAPITULO 1.....	12
EI CONTEXTO ACTUAL Y EL ENCUADRE DEL PROBLEMA DE INVESTIGACIÓN.....	12
Contexto Actual.....	12
PLANTEAMIENTO DEL PROBLEMA.....	16
PREGUNTAS DE INVESTIGACIÓN.....	16
OBJETIVO GENERAL.....	17
OBJETIVOS ESPECÍFICOS.....	17
JUSTIFICACIÓN.....	17
CAPÍTULO 2.....	20
ESTADO DE INVESTIGACION SOBRE EL TEMA.....	20
Qué significa Evaluar.....	27
Evaluación del aprendizaje de la programación.....	31
CAPÍTULO 3.....	37
PERSPECTIVA TEÓRICA RESPECTO A CONCEPTOS APLICADOS AL PROBLEMA DE INVESTIGACIÓN...	37
Implicancias de resolver problemas.....	37
Entornos virtuales de aprendizaje.....	40
CAPÍTULO 4.....	46
ASPECTO METODOLÓGICO.....	46
Introducción.....	46
Metodología de seguimiento.....	50
Propuesta para el seguimiento del avance del pensamiento computacional de estudiantes de Ingeniería Eléctrica.....	51
Primera etapa de diagnóstico:.....	52
Segunda etapa: Ejercitación.....	55
Tercera etapa: trabajo colaborativo.....	59
Capítulo 5.....	63
Resultados.....	63

CAPÍTULO 6	77
CONCLUSIONES.....	77
Resultados en relación a los objetivos.....	77
En relación a las preguntas de Investigación:.....	80
Mejoras a futuro	83
Bibliografía:.....	85
Anexo I	89
Anexo II	91
Anexo III	94

DEDICATORIA

Este trabajo de tesis está principalmente dedicado a mi familia (Leo y Alfredo), sin ellos no sería posible seguir mis sueños. A mis padres que siempre me estimularon para que logre mis objetivos sin bajar los brazos y a luchar siempre en circunstancias adversas, a todos los amigos y colegas que me dieron fuerzas cuando estaba desalentada y abatida, también a los docentes guías. Sin todos ellos no hubiera logrado llegar al final de este camino tan esperado de obtener este título de Postgrado.

AGRADECIMIENTOS

Este trabajo de tesis resulta del esfuerzo conjunto de maestrando, familia, director y co-directora de tesis, colegas y alumnos de la carrera de Ingeniería Eléctrica y de Ingeniería en Agrimensura de la FaCENA-UNNE.

En primer lugar, mi agradecimiento va a mi familia conformada por mi esposo y mi hijo que me acompañaron y toleraron con su paciencia y calma las horas de trabajo, dejando de lados muchas otras obligaciones para poder permitirme lograr este anhelado sueño. A mi amiga Yolanda con la que cursé esta Maestría y con la que mutuamente nos alentamos a seguir adelante cuando ya no nos quedaban fuerzas a raíz del trabajo, la familia y otras obligaciones educativas.

A mi director de tesis que me dirigió y fue de guía y ayuda para realizar las etapas del mismo promoviendo en cada paso el esfuerzo que debía realizar para concluir cada etapa.

A mi co-directora de tesis, quién sostuvo desde el comienzo la realización del mismo y que me orientó en los diferentes desarrollos que implicó el trabajo.

A mis estudiantes de la carrera de Ingeniería eléctrica que se prestaron a los diferentes desafíos que les fuimos planteando a través del aula virtual de la asignatura, en el año de pandemia y posterior al mismo para lograr que tuvieran éxito en sus aprendizajes.

¡A todos ellos gracias!!

INDICE DE FIGURAS

Figura 1: Pantalla principal del aula virtual de la asignatura informática.....	pág.15
Figura 2: Habilidades del pensamiento computacional.....	pág.21
Figura 3: Fases en la resolución de problemas - elaboración propia.....	pág.23
Figura 4: Aprendizaje centrado en el estudiante- elaboración propia.....	pág.26
Figura 5: Descripción de las tareas complejas de competencias resolución de problemas.....	pág.32
Figura 6: E-rubricas para evaluar la competencia y resolución de problemas.....	pág.32
Figura 7: Perfil de un usuario de Scratch.....	pág.35
Figura 8: Evaluación de un programa de Scratch de nivel medio realizado con Dr. Scratch.....	pág.35
Figura 9: Pantalla del ingreso a la versión Beta de Dr. Scratch.....	pág.36
Figura 10: recursos de la plataforma virtual Moodle de la asignatura informática.....	pág.43
Figura 11: recorte del Padlet utilizado dentro del aula de informática.....	pág.45-46
Figura 12: Pantalla de del aula virtual de la asignatura informática en Moodle.....	pág.52
Figura 13: Enunciado realizado por alumnos del grupo 1 – IE.....	pág.58
Figura 14: Enunciado realizado por alumnos del grupo 2 – IE.....	pág.58
Figura 15: Enunciado realizado por alumnos del grupo 3 – IE.....	pág.59
Figura 16: Enunciado realizado por alumnos del grupo 4 – IE.....	pág. 59
Figura 17: Alumnos de Ingeniería Eléctrica durante el trabajo en el aula.....	pág.60
Figura 21: recorte de respuestas de los alumnos de IE – a la Pregunta 3.....	pág.65-66
Figura 22: respuesta de los alumnos de IE – pregunta 6 ¿Qué entiende por algoritmo?.....	pág.68-69
Figura 23: recorte Respuestas de los alumnos de IE a la pregunta 8.....	pág.70
Figura 24: Formato de la encuesta inicial a estudiantes de la IE de la FaCENA UNNE.....	pág.90-91
Figura 25: pregunta 1 de Plataforma Moodle.....	pág.95
Figura 26: pregunta 2 y 3 de Plataforma Moodle.....	pág.95
Figura 27: pregunta 4 de Plataforma Moodle.....	pág.95
Figura 28: preguntas 1 y 2 referente a lenguaje C de Plataforma Moodle.....	pág.96
Figura 29: preguntas 3 y 4 referente a lenguaje C de Plataforma Moodle.....	pág.96
Figura 30: pregunta 5 referente a lenguaje C de Plataforma Moodle.....	pág.96
Figura 31: pregunta 2 referente a selectiva if en C de Plataforma Moodle.....	pág.97
Figura 32: pregunta 4 referente a selectiva if en C de Plataforma Moodle.....	pág.97
Figura 33: pregunta 54 referente a selectiva if en C de Plataforma Moodle.....	pág.98

INDICE DE TABLAS

Tabla 1: Pilares del pensamiento computacional - elaboración propia.....	pág.22
Tabla 3: Tipos de evaluación según García Aretio L. (2008)	pág.24
Tabla 4: Marcadores de evaluación del PC para ejercicios de programación.....	pág.29
Tabla 5: Marcadores para medir la evaluación de la 3ra. etapa.....	pág.56
Tabla 6: Respuestas a la pregunta 4 - ¿Qué es un problema?.....	pág.67
Tabla 7: Porcentajes de evaluación a alumnos de IE según rúbrica de la etapa-2 de la metodología.....	pág.72
Tabla 8: Porcentajes de la evaluación a alumnos de IE según rúbrica de la etapa-3 de la metodología.....	pág. 74

INDICE DE GRAFICOS

Gráfico 1: respuesta de los alumnos de IE – a la Pregunta 1.....	pág.64
Gráfico 2: respuesta de los alumnos de IE – a la Pregunta 2.....	pág. 65
Gráfico 3: respuesta de los alumnos de IE – a la pregunta 5.....	pág.68
Gráfico 4: respuesta de los alumnos de IE – a la pregunta 8.....	pág. 69

Introducción

Se presentan en esta tesis de Maestría el resultado de lo transitado al momento de elaborarla como así también se expresa lo experimentado durante todo su desarrollo tanto al escribirla como al transmitir lo vivenciado con los estudiantes durante las instancias de aplicación metodológica de la misma; asimismo destaco el entusiasmo que deposité y las ganas de aprender mucho más sobre aulas virtuales y cómo poder sacarle el máximo provecho a las mismas a la hora de evaluar el aprendizaje de mis estudiantes de ingeniería eléctrica, en un tema complejo y poco cuantificable como es el pensamiento computacional y la programación.

Por otra parte, les comparto que no fue fácil poder decidir cómo haría para comenzar a escribir la misma sin antes haber recorrido muchos trabajos en internet en los que se mencionaban diferentes formas de evaluar el pensamiento computacional. No fue fácil sentarme a armar todo lo que tenía pensado y toda la información recopilada tantas veces y que había que ordenar y priorizar para poder concluir en este trabajo. La información que se encontraba muchas veces no se ajustaba a las ideas que tenía respecto al tema, otras veces al leer los trabajos sembraba dudas respecto a lo que yo quería proponer como metodología. Reconozco haber estado sentada muchas horas frente a la computadora y comenzar varias veces esta introducción, pero al no ser lo que esperaba volvía prontamente a la hoja de Word en blanco.

Luego de recorrer y leer varios trabajos, lo que quería realizar comenzó a tomar forma y me pareció muy interesante poder volcar lo que investigaron otros autores a lo que yo realmente quería aplicar en mis estudiantes.

Asimismo, no quiero dejar de mencionar el gran aporte de mis estudiantes de la carrera de Ingeniería Eléctrica de la Facultad de Ciencias Exactas y Naturales y Agrimensura (FaCENA) de la Universidad Nacional del Nordeste (UNNE) de Corrientes, República Argentina. Sin ellos no sería posible el logro de estos resultados que voy a

exponer en este trabajo de tesis. Ellos fueron el motor del mismo y la confianza que depositaron durante la etapa del aprendizaje.

Finalmente, para esta tesis se investigó y desarrolló una metodología de seguimiento de los estudiantes de Ingeniería Eléctrica de la FaCENA para medir la adquisición del pensamiento computación durante la cursada de la materia en el segundo cuatrimestre de la carrera de Ingeniería Eléctrica, mediante la utilización de diferentes tipos de recursos, actividades, ejercicios prácticos, estrategias de trabajo en grupo, se seleccionaron herramientas de la plataforma virtual Moodle (cuestionarios, evaluaciones prácticas, presentación de tareas, participación en foros). Otros recursos como los muros colaborativos fueron incorporados como herramienta dentro del aula virtual. Se trabajó utilizando diferentes estrategias de enseñanza aprendizaje con los estudiantes aprovechando al máximo lo que nos ofrecía el aula virtual durante el año de pandemia y el siguiente. Asimismo, se utilizaron rubricas de evaluación para volcar los resultados obtenidos de las diferentes actividades implementadas durante el cursado.

Para ello se trabajó respetando las pautas y estrategias de trabajo, pero continuando el cursado y los diferentes tipos de actividades que se debían realizar para cumplimentar el programa de la materia. Las evaluaciones (cuestionarios teóricos y prácticos) realizados en la plataforma virtual (Moodle) nos acercaron los alcances, potencialidades y límites que ofrecía la misma a partir del trabajo propuesto a los estudiantes de ingeniería eléctrica para luego contrastar dichos resultados con lo realizado al grupo de alumnos de la Ingeniería en Agrimensura, que no estaban expuestos al experimento. Para lo cual se desarrolló y aplicó un mecanismo de seguimiento de cada estudiante implicado desde el comienzo de la cursada hasta el final de la misma.

Como adelanto de los temas que se expondrán en este trabajo de tesis se describen los siguientes capítulos:

Capítulo 1: EL CONTEXTO ACTUAL Y EL ENCUADRE DEL PROBLEMA DE INVESTIGACIÓN. En este capítulo se presenta la problemática actual que muestran los

estudiantes y se introduce y argumenta respecto a la elección del tema y la definición del problema objeto de estudio de esta tesis. El objetivo y los específicos ubican al abordaje teórico y metodológico y su justificación como caso de estudio.

Capítulo 2: ESTADO DE INVESTIGACION SOBRE EL TEMA. En este capítulo se muestra lo recabado referente a las diferentes problemáticas/temáticas de investigación relacionadas con el tema/problema de la tesis.

Capítulo 3. PERSPECTIVA TEÓRICA. Abarca el desarrollo de conceptos aplicados al tema/ problema objeto de la investigación. Conceptos que sirven de sustento teórico para exponer las concepciones desde las que se aborda el aprendizaje y adquisición del pensamiento computacional y sus implicancias a la hora de su evaluación.

Capítulo 4. ASPECTO METODOLÓGICO. Presenta la perspectiva teórica metodológica seleccionada para el abordaje del tema-problema objeto de investigación, especificando la metodología que se desarrolló a los fines de desarrollar los marcadores válidos, instrumentos y técnicas de recolección de información.

Capítulo 5. PRESENTACIÓN DE RESULTADOS Y DISCUSIÓN. Muestra las opciones y decisiones acerca de los procedimientos y maneras metodológicas aplicadas y análisis de la información obtenida a partir de las técnicas implementadas en el curso de la investigación. Observación y análisis que se fueron realizando durante la puesta en marcha de esta metodología con los alumnos de Ingeniería Eléctrica, de modo que puedan seleccionarse otros datos en caso de ser necesarios.

Capítulo 6. CONCLUSIONES. Se exponen los resultados generales de la investigación teniendo en cuenta los objetivos de la investigación y los resultados obtenidos, las respuestas a las preguntas de investigación y posibilidades a futuro.

CAPITULO 1

EL CONTEXTO ACTUAL Y EL ENCUADRE DEL PROBLEMA DE INVESTIGACIÓN

Contexto Actual

En los cursos introductorios de programación, los profesores inician a los estudiantes en la disciplina informática¹ y en el proceso de resolución de problemas algorítmicos². El pensamiento computacional y la programación ayudan a los estudiantes proporcionando oportunidades de aprendizaje para desarrollar el conocimiento y saber hacer, necesarios para comprender un problema, analizarlo críticamente y desarrollar una solución eficiente en este mundo altamente tecnológico.

Aprender a programar es una tarea compleja que requiere, primero, entender el problema y luego diseñar una solución algorítmica que represente los pasos de solución para implementarlo en un lenguaje de programación. La construcción de la solución en un lenguaje de programación es un proceso cíclico que consiste en codificar, compilar, probar y depurar programas. Para comprobar que un programa está correcto, éste debe pasar exitosamente varios casos de prueba. Cada caso de prueba consiste en ejecutar el programa usando un conjunto de datos de entrada y luego analizar los resultados.

La enseñanza del pensamiento lógico y la programación en el ámbito universitario, como así también la forma de evaluación de los aprendizajes que han adquirido nuestros

¹ Informática: Según Konrad Zuse, ingeniero alemán y uno de los principales pioneros de la computación, la informática "Es la disciplina que estudia el tratamiento automático de la información utilizando dispositivos electrónicos y sistemas computacionales."

² El proceso de resolución de problemas algorítmicos es un conjunto de pasos a seguir para la solución de un problema.

estudiantes se realiza de manera habitual, es decir el docente imparte el contenido teórico del tema ayudado por un proyector y una presentación visual.

Para la parte práctica los docentes desarrollan un ejemplo de la guía práctica en el pizarrón y los alumnos realizan en papel el resto de los ejercicios, si surgen dudas las mismas son evacuadas para toda la clase por el docente. Al momento de ser evaluados el alumno prepara el tema y se presenta al examen, el cual tiene una parte escrita en papel y otra parte que se realiza en máquina, con el uso del software de programación específico de la materia, en el cual el alumno debe resolver el algoritmo adecuado al problema planteado, codificar y ejecutar correctamente en la computadora.

Los estudiantes que llegan al primer año de la carrera de Ingeniería eléctrica se encuentran con el desafío que en un cuatrimestre deben incorporar el pensamiento computacional³ mediante la resolución de problemas algorítmicos. El desafío no es fácil debido que programar computadoras es una tarea compleja, como lo es el desarrollo de habilidades de comprensión y entendimiento del problema.

Cada año se puede apreciar que los alumnos, en esta primera etapa, se encuentran con serios problemas en su desempeño académico debido a que el pensamiento lógico-computacional y la abstracción son habilidades que no les fueron enseñadas y por lo tanto no la han incorporado durante su paso por los distintos niveles del sistema educativo. *“Las tecnologías digitales y las redes sociales han hecho visibles las prácticas comunicativas que imperan en la educación: transmisivas y reproductoras donde las aulas actúan como si se tratara de un medio de comunicación masivo, un emisor y decenas o cientos de receptores. En muchas ocasiones se utilizan tecnologías para la conectividad, pero, para repetir las viejas concepciones pedagógicas de la reproducción y el aislamiento”* (Aparici, 2011).

³ El Pensamiento Computacional, según Wing, “implica resolver problemas, diseñar sistemas y comprender el comportamiento humano, basándose en los conceptos fundamentales de la informática. El Pensamiento Computacional incluye una gama de herramientas mentales que reflejan la amplitud del campo de la informática”.

La profesora del Departamento de Computación de la universidad de Carnegie Mellon de Estados Unidos, Jeannette M. Wing fue una de las primeras especialistas en utilizar el término de pensamiento computacional.

En particular, la asignatura Informática la cursan alumnos provenientes de diferentes escuelas de nivel secundario y de diferentes localidades de la provincia y provincias vecinas: Sus edades oscilan entre los 18 a 25 años, son asiduos a la tecnología dado que utilizan frecuentemente celulares, tabletas y notebook para sus actividades personales y académicas. Los contenidos que se dictan en la asignatura informática se dividen en unidades acordes al programa de la misma. Son contenidos referentes al hardware y software de la computadora, sistema operativo y software de aplicación, como también se desarrollan unidades referentes a la resolución de problemas por medio de algoritmos, las técnicas que utilizan son de manera gráfica por medio de un diagrama de flujo o escrita por medio de pseudocódigo, luego se codifica la solución en lenguaje de programación (lenguaje C), se compila, se corrigen errores y se prueba con datos reales, es decir se ponen en prácticas las fases para la resolución de problemas en computadora que son: el análisis del problema, diseño del algoritmo, codificación, ejecución y prueba.

En la actualidad, la asignatura informática cuenta con un aula virtual en la plataforma (Moodle) proporcionada por la Universidad Nacional del Nordeste Virtual (UNNE Virtual), y es utilizada como complemento de las clases presenciales. En ella se suben y comparten con los estudiantes los contenidos teóricos y las guías de trabajos prácticos para desarrollar en aula y en laboratorio y todo otro material adicional que el docente estima pertinente y adecuado para ampliar los conocimientos del estudiante. A través del aula virtual los alumnos pueden comunicarse con el docente y entre ellos, vía mensajería interna de la misma. Por medio del aula deben presentar trabajos que son requisito indispensable para regularizar la materia. Se informan también las novedades respecto al cursado, parciales, resultados de parciales, cursos y cualquier otra noticia de interés para los alumnos. No se

realizan parciales vía aula virtual, y se utiliza muy poco la herramienta foro para interactuar sobre algún tema en particular dado en la materia.

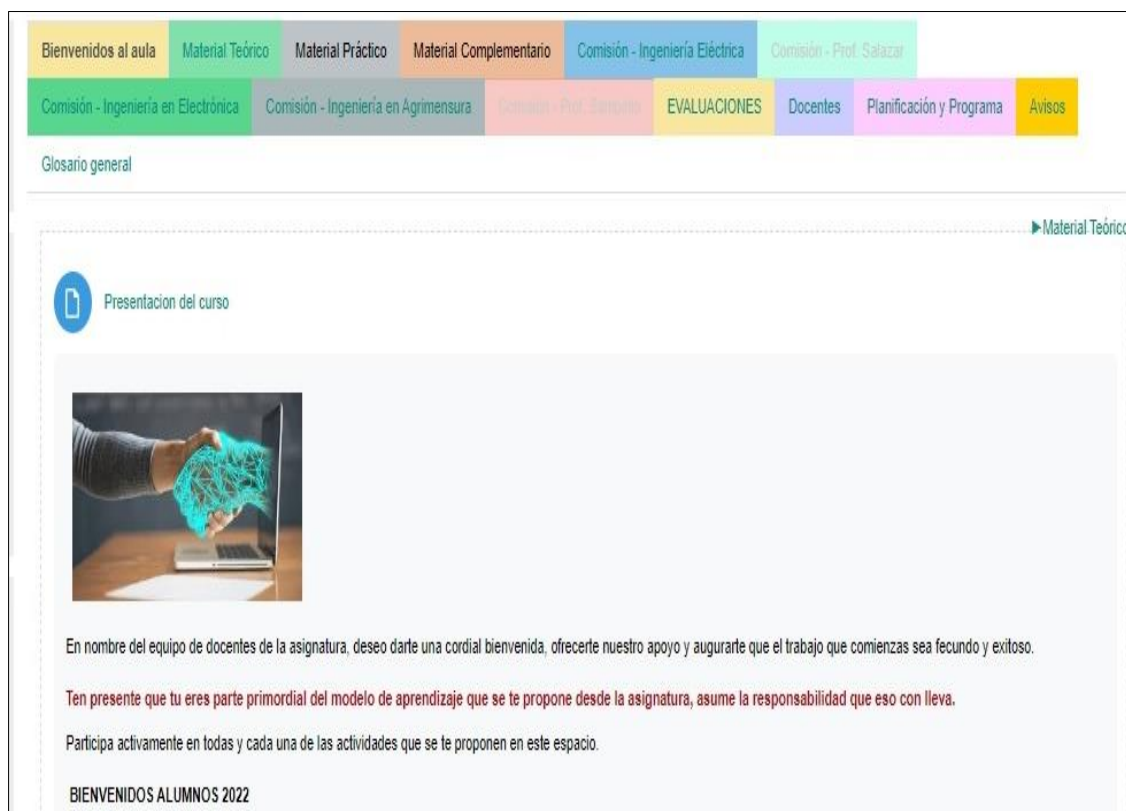


Figura 1: Pantalla principal del aula virtual de la asignatura informática

Fuente: Plataforma Moodle FaCENA_UNNE

En la actualidad es importante que la evaluación sea parte integral del proceso de enseñanza-aprendizaje; a la vez que sirva para aportar información útil para estudiantes, profesores e instituciones. Para ello debe aplicarse de forma permanente y propiciar una discusión efectiva sobre las dificultades que se evidencien en el aprendizaje de los alumnos de manera tal de llevar a cabo acciones correctivas de forma eficiente que favorezcan a mejoras en el aprendizaje. La evaluación debe verse como un proceso continuo, integral y participativo que permita identificar una problemática, analizarla y explicarla mediante información relevante.

PLANTEAMIENTO DEL PROBLEMA

Para la definición del problema de investigación se tuvo en cuenta lo planteado en este capítulo del cual surge:

¿El uso de recursos de la plataforma EVEA (Moodle), permite medir el grado de avance en el pensamiento computacional?

¿Los alumnos de la asignatura informática, pertenecientes al primer año del plan de estudios de la carrera de Ingeniería Eléctrica, tuvieron mejor rendimiento académico que los alumnos de Ingeniería en Agrimensura?

Expresada las mismas en forma afirmativa:

Uso de recursos de la plataforma EVEA (Moodle), para medir el grado de avance en el pensamiento computacional.

Los alumnos de la asignatura informática, pertenecientes al primer año del plan de estudios de la carrera de Ingeniería Eléctrica tuvieron mejor rendimiento académico que los estudiantes de Ingeniería en Agrimensura.

PREGUNTAS DE INVESTIGACIÓN

Del planteamiento del problema surgen las siguientes preguntas de investigación que serán respondidas durante el desarrollo de este trabajo:

1. *¿Qué recursos de la plataforma Moodle son pertinentes para medir el grado de avance en el pensamiento computacional de los alumnos de Ingeniería eléctrica?*
2. *¿Cuál es el nivel de conocimiento y competencias en pensamiento computacional y programación que poseen los estudiantes de Ingeniería Eléctrica al ingresar a la universidad?*
3. *¿Cuáles son las principales dificultades o desafíos que enfrentan los estudiantes de Ingeniería Eléctrica en el desarrollo del pensamiento computacional y la programación?*
4. *¿Qué métodos o enfoques pedagógicos se utilizan actualmente en la enseñanza del pensamiento computacional y la programación en Ingeniería Eléctrica y cuál es su efectividad?*

5. *¿Qué recomendaciones pueden formularse para mejorar la enseñanza y el desarrollo del pensamiento computacional y la programación en el plan de estudios de Ingeniería Eléctrica?*

OBJETIVO GENERAL

El objetivo general de esta tesis de investigación es desarrollar una metodología para evaluar el pensamiento computacional de los estudiantes de Ingeniería Eléctrica de la Facultad de Ciencias Exactas y Naturales y Agrimensura de la UNNE.

OBJETIVOS ESPECÍFICOS

1. Identificar las habilidades y competencias en pensamiento computacional y programación que se espera que los estudiantes adquieran durante su formación.
2. Diseñar una metodología de seguimiento que permita evaluar la adquisición o avance en el desarrollo del pensamiento computacional y la programación.
3. Implementar la metodología de seguimiento en un grupo piloto de estudiantes de Ingeniería Eléctrica y recopilar datos relevantes.
4. Analizar los resultados obtenidos a través de la metodología de evaluación.
5. Proponer recomendaciones y mejoras para fortalecer el desarrollo del pensamiento computacional y la programación en el plan de estudios de Ingeniería Eléctrica de la UNNE.
6. Elaborar un informe final que presente los resultados de la investigación y las recomendaciones para mejorar la formación en pensamiento computacional y programación en la carrera de Ingeniería Eléctrica.

JUSTIFICACIÓN

Los motivos de esta investigación sobre la evaluación del pensamiento computacional y la programación se centran en que, en la actualidad, éstos se han convertido en habilidades fundamentales en múltiples campos, incluida la ingeniería eléctrica. Se pretende desarrollar una metodología de evaluación sistemática que permita

identificar las fortalezas y debilidades de los estudiantes en estas áreas. Esto proporcionará información valiosa para el diseño de estrategias pedagógicas más efectivas y ajustadas a las necesidades de los estudiantes.

Desde las ciencias de la computación, el pensamiento computacional (PC) beneficia a los estudiantes principalmente en tres aspectos que son: *“permitir la resolución de problemas mediante las computadoras”*, *“enseñar a identificar los problemas que se pueden resolver”*, e *“impulsar a investigar modelos computacionales para situaciones que tradicionalmente no están relacionadas con las computadoras”*⁴. Como lo expresan Mohaghegh y McCauley (2016), *“el pensamiento computacional beneficia al estudiante en cualquier disciplina”*.

Necesidad de una evaluación sistemática: Para mejorar la enseñanza y el aprendizaje del pensamiento computacional y la programación, es fundamental contar con una evaluación sistemática que permita identificar las fortalezas y debilidades de los estudiantes en estas áreas. Esto proporcionará información valiosa para el diseño de estrategias pedagógicas más efectivas y ajustadas a las necesidades de los estudiantes.

Optimización de la formación académica: Mediante el seguimiento de los estudiantes de Ingeniería Eléctrica, se podrán identificar los momentos claves en los que se produce el desarrollo del pensamiento computacional y la programación, así como las dificultades más comunes que enfrentan los estudiantes. Esto permitirá ajustar el plan de estudios y los recursos educativos para mejorar la formación académica en estas áreas.

Contribución al cuerpo de conocimiento: La investigación propuesta pretende desarrollar una metodología de seguimiento específica para evaluar el grado de avance en el desarrollo del pensamiento computacional y la programación en estudiantes de Ingeniería Eléctrica. Esta metodología puede ser aplicada en otros contextos educativos similares,

⁴ Revista de Filosofía Vol. 38, N°99, (Sep-Dic) 2021-3, pp. 369 - 379 Universidad del Zulia. Maracaibo-Venezuela ISSN: 0798-1171 / e-ISSN: 2477-9598 Carencias en la evaluación del pensamiento computacional

aportando conocimientos y herramientas prácticas para la evaluación y mejora de las habilidades computacionales en otras instituciones.

En resumen, la justificación de esta tesis de investigación radica en la importancia de evaluar el grado de avance en el desarrollo del pensamiento computacional y la programación en estudiantes de Ingeniería Eléctrica de la FaCENA_UNNE, con el objetivo de optimizar la formación académica y contribuir al avance del conocimiento en el área de la educación en ingeniería.

CAPÍTULO 2

ESTADO DE INVESTIGACION SOBRE EL TEMA

Para la Royal Society (2012), *“El pensamiento computacional es el proceso de reconocer aspectos de computación en el mundo que nos rodea, y aplicar herramientas y técnicas desde la ciencia de la computación para comprender y razonar acerca de sistemas y procesos artificiales y naturales”*.

Selby (2015), incluye en la definición del PC las habilidades de abstracción, descomposición, diseño algorítmico, generalización y evaluación, en su trabajo de investigación.



Figura 2: habilidades del pensamiento computacional⁵
Fuente: <https://smartteamdigital.com/proyecto-educativo/pensamiento-computacional/>

En la Tabla 1, se puede ver el significado que tienen cada uno de estos conceptos:

⁵ El pensamiento computacional (Selby, 2015; Wing, 2006) es un proceso cognitivo que permite la generación de soluciones a problemas a través del uso de habilidades específicas, tales como abstracción, descomposición, generalización, evaluación y diseño algorítmico.

Abstracción	Se trata de la identificación de los pasos esenciales en un proceso y su relación con las instrucciones de la programación.
Reconocimiento de patrones	Permite optimizar procedimientos e identificar secuencias de instrucciones que luego pueden reutilizarse en otros contextos
Descomposición o modularización	Ayuda a resolver problemas complejos separándolos en pequeñas partes y reuniendo luego todas las soluciones o partes parciales en un producto final.
Pensamiento algorítmico	Implica ordenar secuencias de instrucciones y manejar estructuras de control en un programa.

Tabla 1: Pilares del pensamiento computacional - elaboración propia

Estas competencias no solo permiten entender el funcionamiento de las nuevas tecnologías, también ayudan al abordaje de problemas y situaciones complejas. Por eso *Steve Jobs*⁶, decía que “*todo el mundo debería aprender a programar una computadora, porque te enseña a pensar*”. Esta afirmación significa la comprensión de una serie de habilidades asociadas con este aprendizaje de la programación para el presente y futuro.

El estudio de este caso particular pretende dar cuenta la importancia de la adquisición del pensamiento computacional (PC) como así también la relevancia de la resolución de problemas, y de la aplicación de las fases para la resolución de problemas con una computadora que son:

1. Análisis del problema,
2. Diseño del algoritmo,
3. Codificación,
4. Ejecución y prueba.

⁶ Steve Jobs: Steven Paul Jobs fue un empresario, diseñador industrial, magnate empresarial, propietario de medios e inversor estadounidense. Fue cofundador y presidente ejecutivo de Apple y máximo accionista individual de The Walt Disney Company. (fuente: <https://www.biografiasyvidas.com/biografia/j/jobs.htm>)

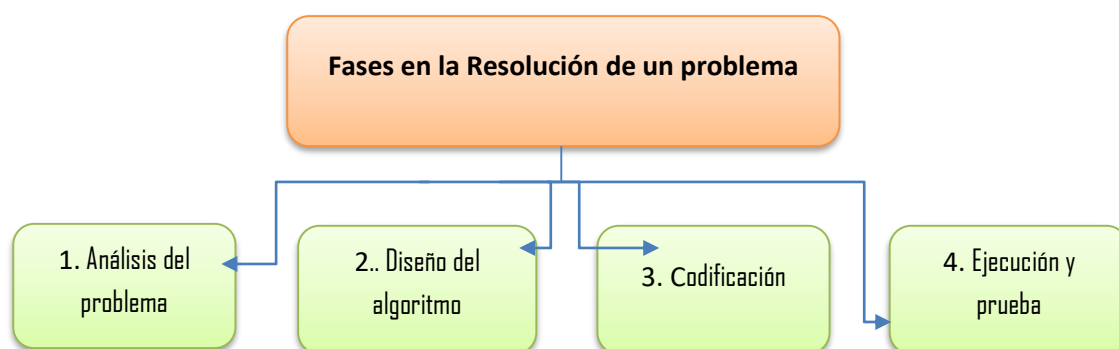


Figura 3: Fases en la resolución de problemas - Elaboración propia

Análisis del problema	Consiste en la definición del problema, donde se contemple exactamente lo que debe hacer el programa y el resultado o solución deseada. Dado que se busca una solución por computadora, se precisan especificaciones detalladas de entradas y salidas. Para resolver un problema con la computadora hay que disponer de los datos de entrada, estudiar el tratamiento que se ha de realizar a los mismos, la información que se desea obtener como resultado y de que manera deben presentarse.
Diseño del algoritmo	Teniendo en cuenta que un algoritmo es un método para resolver problemas, una vez analizado el mismo se precisa diseñar un algoritmo que indique claramente los pasos a seguir para resolverlo. Para realizar un determinado proceso, se le debe suministrar a la computadora una fórmula para la resolución de un problema (<i>algoritmo</i>), cuyo diseño debe ser independiente de la computadora que resuelve el problema. Dada la importancia del algoritmo en la ciencia de la computación, un aspecto muy importante será el <i>diseño del algoritmo</i>. En esta etapa se realizará una representación de la secuencia. Estas representaciones son las herramientas de: <i>diagramas de flujo</i>, <i>pseudocódigos</i> y/o <i>tablas de decisión</i>⁷.
Codificación	Una vez que el diagrama de flujo o el algoritmo de resolución del problema está definido se pasa a la fase de codificación del programa en cualquier lenguaje, cuyo resultado será el programa fuente, el cual sigue las reglas de sintaxis que el lenguaje elegido exija. Esta codificación se realiza en el editor de programas.

⁷ Una tabla de decisiones es una entrada de lógica de reglas planificadas, en formato de tabla, que se compone de condiciones, representadas en las cabeceras.

	Una vez dentro de la computadora el programa deber ser traducido al único lenguaje que éste entiende: Lenguaje de máquina. Dicha operación se realiza mediante el correspondiente programa traductor o compilador del lenguaje en el que está escrito el programa
Ejecución y Prueba	Antes de dar por finalizada cualquier labor de programación, es fundamental preparar un conjunto de datos lo más representativo posible del problema, que permitan probar el programa cuando se ejecute y así verificar los resultados. Cuanto más exhaustivas sean las pruebas de un programa, mayor seguridad se tendrá de que éste funcione correctamente y, por lo tanto, menor posibilidad de errores. El programa se considera terminado cuando se han realizado pruebas y ensayo de su fiabilidad con el conjunto de datos seleccionados y otros nuevos, hasta incluso con datos reales, y no se encuentren errores de ningún tipo.

*Tabla 2: Fases en la resolución de problemas en computadora
fuente: tema- 4 de la cátedra Informática*

Como antecedente de estudios sobre pensamiento computacional similares podemos citar a, Wing⁸ (2010) quien afirma que a mediados del siglo XXI el Pensamiento Computacional será una destreza fundamental usada por todo el mundo, así como lo son ahora otras destrezas como leer, escribir y la aritmética, de allí la importancia que se le da a la incorporación del mismo desde la materia informática, donde el alumno comienza los primeros pasos en la resolución de problemas con algoritmos, destaca además la importancia de medir estas destrezas de diferente formas. El pensamiento computacional es un proceso que abarca resolver problemas, diseñar sistemas y entender el comportamiento humano, haciendo uso de los conceptos fundamentales de la informática (Wing, 2006 pp 33).

Como bien sabemos el pensamiento computacional o (Computational Thinking), no posee un marco de referencia común que permita comprobar cuál es su definición y qué

⁸ Jannette Wing: vicepresidente corporativo de Microsoft Research y profesora de Computer Science Department Carnegie Mellon University, fue quien popularizó el término en su artículo “*Computational Thinking*”.

componentes la forman (Zapata_Ros, 2015) por consiguiente este hecho provoca un vacío en el desarrollo de herramientas y metodologías de evaluación (Román, 2015). Nace una necesidad creciente de llevar a cabo estudios que permitan evidenciar el aprendizaje de la programación en beneficio del pensamiento computacional (Zapata-Ros,2015).

Dentro de la evolución tan vertiginosa de las TIC, nos han hecho cambiar la mirada de la educación centrada en el docente hacia una educación centrada en el alumno, quien ya no es visto como un receptor de conocimientos sino como el generador (constructor) del mismo, el papel del docente es otro: el de guía, acompañante, facilitador; siendo el alumno el que tiene el rol activo en este nuevo proceso.

Este enfoque considera a los estudiantes como protagonistas activos de su propio proceso de aprendizaje y no como sujetos pasivos a los que se les imparten clases magistrales, independientemente de su aprovechamiento por los mismos. Es decir, es un cambio de paradigma de un modelo educativo centrado en la excelencia de la docencia impartida por el profesor a un modelo en donde lo que importa son los resultados del aprendizaje adquirido por los estudiantes (Gibbs, 1995). Representa, por tanto, un cambio cultural caracterizado por métodos docentes innovadores (Kember, 2008), promoviendo tanto los conocimientos cognitivos propios del tema como competencias transversales (O'Neill y McMahon, 2005) tales como ciudadanía activa, resolución de problemas, pensamiento crítico, trabajo en equipo, liderazgo, comunicación, compromiso social, etc

La educación centrada en el estudiante es un “nuevo paradigma” una corriente pedagógica actual que coloca al estudiante en el centro del proceso, donde los factores sociales y afectivos tienen la misma importancia que los cognitivos⁹.

⁹ Cognitivo: del conocimiento o relacionado con él. “la psicología cognitiva estudia procesos mentales como la percepción, la memoria o el lenguaje”

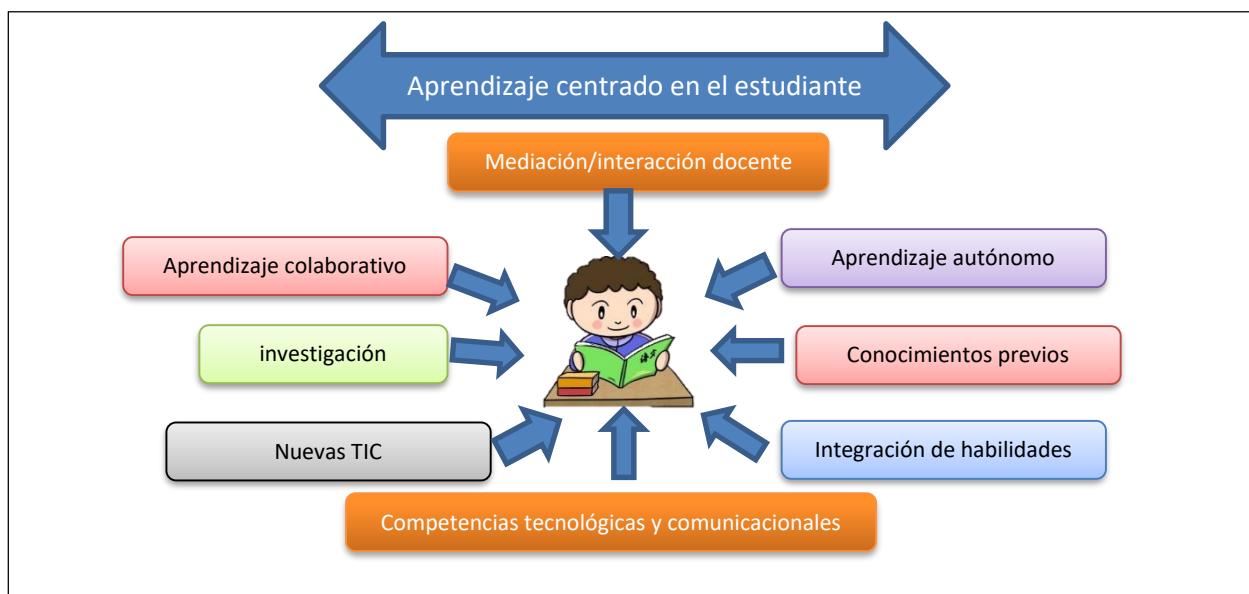


Figura 4: Aprendizaje centrado en el estudiante- elaboración propia

En el trabajo realizado por Salinas J., De Benito B. y Lizana A. (2014), dichos autores hablan de nuevas competencias que deben adquirir tanto docentes como alumnos para hacer frente a propuestas de aprendizaje de carácter tecnológico, donde dicen que, “...Las instituciones de educación, especialmente las de educación superior, necesitan adaptarse a nuevas modalidades de formación más acordes con la nueva situación: desde las aulas convencionales unidas a través de la red hasta grupos de trabajo colaborativo en contextos totalmente a distancia; desde clases de la educación formal a comunidades de práctica o al aprendizaje incidental..”. Lo cual implica hacer frente a un nuevo escenario educativo, aprovechar las herramientas que en la actualidad nos presentan las TIC e incorporar las mismas para facilitar la enseñanza-aprendizaje.

Otro punto tiene que ver con el “*pensamiento computacional*”, el cual se deduce cómo la abstracción y la descomposición de problemas que pueden aplicarse a cualquier área del conocimiento. Mediante el pensamiento computacional, puede utilizarse la computadora de diversas maneras como un nuevo instrumento científico (así como en un momento de la historia lo fueron el telescopio o el microscopio) para beneficiar a los estudiantes, las

instituciones en todos los niveles de educación y, finalmente, el desarrollo humano, científico y tecnológico global, que lleva a tomar decisiones (Zapotecatl, 2014, pp. 8, 10).

Los centros educativos innovadores forman parte de una nueva ecología del aprendizaje que tiene como particularidad romper los límites de espacio-temporales del proceso educativo gracias a las tecnologías digitales. Simultáneamente, se alteran los tradicionales roles educador-educando, puesto que, en el contexto de una enseñanza y aprendizaje en red, no se define de manera inamovible quién enseña y quién aprende. Desde este enfoque se entiende que los sistemas educativos no necesitan más contenidos sino, esencialmente, entornos flexibles que permitan desarrollar las capacidades de autoaprendizaje, creatividad, autonomía, iniciativa y expresión multilenguaje. En este contexto ha resurgido en los últimos años un movimiento educativo a nivel internacional relacionado con la introducción del pensamiento computacional, la programación informática y la robótica en las escuelas.

El pensamiento computacional no es sinónimo de capacidad para programar una computadora, dado que se requiere pensar en diferentes niveles de abstracción y es independiente de los dispositivos. Se puede desarrollar pensamiento computacional sin utilizar una computadora (basta con usar el papel y el lápiz), si bien los dispositivos digitales nos permiten abordar problemas que sin ellos no nos atreveríamos a enfrentar.

La programación fomenta la construcción de conocimientos mediante la “apropiación”, es decir, de manera que los resultados de las acciones cognitivas son consideradas como propias-personales. El pensamiento computacional proporciona una estructura indispensable para el estudio de la programación, que va más allá de la codificación en sí misma. Permite al alumno enfrentar problemas, descomponerlos en elementos y encontrar algoritmos que los resuelvan. En consecuencia, el pensamiento computacional implica: descomposición, reconocimiento de patrones, abstracción, generalización de patrones y diseño algorítmico (Kemp, 2014).

Qué significa Evaluar

Del texto de Camilloni A.(1998). “*La calidad de los programas de evaluación y de los instrumentos que los integran*”, se rescata lo que expresa la autora respecto a que “... No es posible hablar de la evaluación de los aprendizajes al margen de los procesos de enseñanza y aprendizaje que los han generado...”. Camilloni enfatiza que la evaluación debe ser entendida como parte integral y complementaria del proceso educativo. Implica que la manera en que se evalúan los aprendizajes debe estar estrechamente vinculada con cómo se enseñan y se adquieren esos conocimientos y habilidades. Implica que la evaluación no solo debe medir el nivel de conocimiento adquirido por los estudiantes, sino también su comprensión, aplicación y capacidad para transferir esos conocimientos a nuevas situaciones.

García Aretio L. (2008). *Evaluación en formatos no presenciales*. Editorial Del Bened. Enero 2000. Nos habla de *¿Cómo podemos evaluar?* y de la importancia que tienen las decisiones que se toman previas al desarrollo de una materia, es por ello que este autor destaca diferentes tipos de evaluación, como ser la evaluación *sumativa – formativa*. *La evaluación procesual*. *La evaluación cuantitativa y cualitativa*, las cuales se explican en detalle en la tabla-3.

Sumativa – formativa	hace referencia al momento final de un curso o actividad de aprendizaje.
Procesual	pretende que el alumno corrija defectos y confusiones supere dificultades y adquiera competencias, pretende recabar que es lo que alumno no sabe y debería saber en un momento dado.
Cuantitativa y Cualitativa	la cuantitativa se basa más en la medición, cuantificación y control en tanto que la cualitativa pretende recabar otras variables distintas a la objetividad como tener presente todas las variables que inciden en el estudiante por ejemplo al inicio de la cursada, expectativas, horas de la materia, importancia que le da al trabajo colaborativo, se tiene en cuenta al alumno de manera integral.

Tabla 3: Tipos de evaluación según Gracia Aretio L. (2008)

Del trabajo de Meriño Almaguer Y., Lorente Rodríguez A. E., Maribona M. G (2011). *“Propuesta de instrumentos de evaluación para entornos virtuales de aprendizaje: una experiencia en la universidad de las ciencias informáticas”*. Donde se rescata que el desarrollo de las Tecnologías de la Información y las Comunicaciones y su impacto en el sector educacional han puesto a disposición de docente nuevas formas de evaluar con las TIC, se han puesto en práctica varios instrumentos que contribuyen a mejorar este proceso, tal es el caso de los cuestionarios o pruebas objetivas, los blogs, foros, los portafolios electrónicos, así como los trabajos para entrega remota o las actividades y trabajos online. En este caso se presenta como herramientas de evaluación los mapas conceptuales, foros de discusión, cuestionarios interactivos y los e-portfolio para lo cual expone indicadores e integración de todas las evidencias de aprendizaje en el mismo.

Barberá E. (2006), en su artículo *“Aportaciones de la tecnología a la e-evaluación”*, de la Revista de Educación a Distancia, describe cuatro dimensiones que se consideran básicas para una práctica evaluativa completa: *evaluación del aprendizaje, evaluación para el aprendizaje, evaluación como aprendizaje y evaluación desde el aprendizaje*. Para lo cual explica un análisis de las aportaciones específicas que introduce la tecnología en este campo. Explica que la manera en que la tecnología modifica significativamente las prácticas presenciales de evaluación que conocemos como ser: *la evaluación automática, la evaluación enciclopédica y la evaluación colaborativa*. El artículo resalta el valor de los procesos comunicativos de los resultados a los alumnos y, en concreto, el valor de la retroalimentación y la necesidad y posibilidades que ofrece la tecnología para su aprovechamiento.

Díaz Barriga A., Frida & Barroso Bravo, R. (2014), hablan de una evaluación por competencias en su trabajo sobre *“Diseño y validación de una propuesta de evaluación auténtica de competencias”*, ahí nos explica que *“... La evaluación auténtica centrada en el desempeño resulta una opción conveniente para valorar las competencias...”*. La evaluación por competencias se traduce en un conjunto de prácticas encaminadas a recabar

información sobre el desempeño del estudiante a partir de situaciones significativas, contextualizadas y reales. La evaluación ocurre durante todo el proceso enseñanza–aprendizaje y articula sistemáticamente una amplia gama de instrumentos (e-portafolios, escalas, rúbricas, etc.) e implica a diferentes agentes (autoevaluación del estudiante, coevaluación de pares, heteroevaluación del docente). Proporciona un ejemplo de e-portafolio al que fue ajustándolo de acuerdo a la finalidad del curso dictado de manera tal que pueda reflejar mediante ciertas evidencias los aprendizajes adquiridos por los estudiantes.

En el trabajo de López Reguera J., Hernández Rivas C., Farran Y. (2011). *“Una plataforma de evaluación automática con una metodología efectiva para la enseñanza/aprendizaje en programación de computadores”*, presenta la aplicación de una plataforma de evaluación automática para apoyar el proceso de enseñanza/aprendizaje en un curso también introductorio de programación similar al curso de la asignatura informática. En la plataforma expuesta se utiliza un mecanismo que combina análisis estático-dinámico con la aplicación de evaluación en línea favoreciendo la retroalimentación personalizada de los alumnos. La metodología utilizada incluye análisis cuantitativo como cualitativo de los datos que ofrece la plataforma utilizada.

Cubero-Ibáñez, J., Ibarra-Sáiz, M.S. y Rodríguez-Gómez, G. (2018) en su *“Propuesta metodológica de evaluación para evaluar competencias a través de tareas complejas en entornos virtuales de aprendizaje”*, también hacen hincapié en la evaluación por competencias, el trabajo expone un tipo de evaluación de competencias a través de tareas complejas y en un contexto digital con e-rúbricas sacando provecho a lo que ofrecen en la actualidad las tecnologías de la información y la comunicación mediante nuevas tareas y formatos de evaluación más holísticos¹⁰ que permitan evaluar competencias transversales

¹⁰ La evaluación holística en la educación persigue, entre otras cosas, valorar el conocimiento, las habilidades y destrezas que han adquirido y desarrollado los discentes en el programa académico al que están adscritos. (fuente: Maritza Dolores Jaramillo Peláez- Narda del Rosario Cruz Erreyes- Leonela Elizabeth Hidalgo Montero)

difícilmente evaluables mediante los formatos tradicionales. Se comparten aquí ejemplos de las tareas y la evaluación con e-rúbricas.

Resolución de problemas		
	Evaluación inicial	Evaluación final
Situación/problema	En esta tarea partimos de su propia experiencia. Seleccione una situación-problema que haya vivenciado en el pasado, y a partir de la misma vaya respondiendo a la actividad de evaluación propuesta.	Tenemos cinco casas de cinco colores diferentes y en cada una de ellas vive una persona de una nacionalidad diferente. Cada uno de los dueños bebe una bebida diferente, fuma una marca de cigarrillos diferente y tiene una mascota diferente. Después de aportar diferentes claves, la pregunta es: ¿De quién es el pececito?
Recurso	Texto escrito (presentación qué hiciste)	Texto escrito (El acertijo de Einstein) Recuperado de https://goo.gl/f0lxVP
Actividad de evaluación	-Describa la situación-problema ¿Qué pasos siguió para solventarlo? -¿Ha aplicado alguna vez este mismo procedimiento en la resolución de otro problema? Explíquelo. -¿En qué otra situación cree que podría aplicar este mismo método?	-Intente resolver el problema y explique qué pasos ha dado para lograrlo: -¿Cuál es el problema? -¿Qué pasos ha seguido para solventarlo? (Explique detalladamente cada uno de ellos) -¿Ha aplicado alguna vez este mismo procedimiento en la resolución de otro problema? Explíquelo. -¿En qué otra situación cree que podría aplicar este mismo método?

Figura 5: Descripción de las tareas complejas de competencias- Resolución de problemas
Fuente: Cubero-Ibáñez, J., Ibarra-Sáiz, M.S. y Rodríguez-Gómez, G. (2018)

Rubrica para evaluar la competencia Resolución de problemas												
100%	Nivel competencial Resolución de problemas	cotidiano					reflexivo			comprensivo		
33%												
100%	Comprensión del problema	Reconoce los rasgos distintivos de un problema, aunque no conoce muy bien cómo ha llegado a identificarlo, intuitivamente sabe que en esa situación o explicación hay algo que no es coherente.					Analiza de forma sistemática el problema tomando en consideración qué pregunta hay que resolver, qué variables o dimensiones tiene el problema, cómo debe ser la respuesta, etc.			Reconoce las características del problema y busca problemas similares que le sirvan de referencia.		
		0	10	20	30	40	50	60	70	80	90	100
33%												
100%	Planificación y ejecución	Establece un plan de acción para resolver el problema apoyándose en las soluciones aportadas por otros que le resultan adecuadas. Hay un modelo experiencial, pero no teórico de resolución de problemas. Lleva a cabo el plan de acción diseñado					Define o elige una estrategia para enfrentarse al problema tomando como punto de partida un referente teórico que indica qué pasos deben darse para solucionar el problema. Lleva a cabo la estrategia escogida.			Examina diferentes posibilidades para construir o elegir un modo de resolución poniendo a prueba distintas estrategias y comparándolas entre sí para ver cuál es la más acertada. Resuelve el problema		
		0	10	20	30	40	50	60	70	80	90	100
33%												
100%	Revisión y transferencia	Analiza la solución dada al problema y se pregunta en qué medida puede aplicarlo a situaciones similares.					Cuestiona la adecuación de la respuesta que ha dado para resolver el problema preguntándose si hay una manera más sencilla de solucionarlo. Y se plantea la posibilidad de aplicar esta resolución en situaciones similares.			Analiza alternativas a la respuesta dada al problema (teniendo en cuenta qué tipos de conocimiento y qué consecuencias sociales están asociados a cada una de estas alternativas) y establece un procedimiento para determinar cuándo la alternativa adoptada para un contexto determinado, es transferible a otro contexto.		
		0	10	20	30	40	50	60	70	80	90	100

Figura 6: e-rubricas para evaluar la competencia y resolución de problemas
Fuente: Cubero-Ibáñez, J., Ibarra-Sáiz, M.S. y Rodríguez-Gómez, G. (2018)

Evaluación del aprendizaje de la programación

Para la evaluación se adopta el enfoque de “*evaluación para el aprendizaje*” y se recomienda usar técnicas como la autoevaluación, la heteroevaluación (evaluación entre pares), el establecimiento de objetivos como medio para fomentar el aprendizaje independiente, el uso de preguntas abiertas (ejemplo: *¿Por qué has elegido esta solución y no otra?*; “*¿Puedes explicar cómo funciona?*”), es decir, preguntar a los estudiantes lo que ya conocen, lo que quieren aprender y lo que han aprendido, como plataforma para la autoevaluación y el establecimiento de objetivos.

Como antecedentes de trabajos de este tipo podemos mencionar el de Reguera, Jorge & Rivas, Cecilia & Leiva, Yussef, (2011). Quienes exponen una plataforma de evaluación automática que apoya el proceso de enseñanza/aprendizaje en cursos

introdutorios de programación de computadores para estudiantes de ingeniería de la Universidad de Concepción. Esta plataforma posibilita la utilización de mecanismos que combinan el análisis estático/dinámico mediante la aplicación de evaluación de comprensión/análisis en línea, permitiendo una retroalimentación personalizada a los alumnos. Se presenta también los criterios usados para crear secuencias didácticas de problemas y una forma de aplicarlas efectivamente mediante la evaluación automática.

Tras la aplicación de la evaluación en una muestra después de seis años, se denota que la evaluación automática afecta positivamente la motivación, el desempeño y la autoeficacia de los alumnos. La metodología utilizada incluye análisis cualitativo y cuantitativo. Los aspectos cualitativos se extraen mediante la observación del comportamiento de los estudiantes durante el proceso de aprendizaje, mientras que el análisis cuantitativo está basado en los datos de los estudiantes registrados por la plataforma. Todo ello se realizó teniendo en cuenta el grupo de muestra y el grupo de control.

En el trabajo de López Reguera J., Hernández Rivas, C. & Farran Leiva, Y. (2011) se menciona el trabajo de T. Götschi, I. Sanders and V. Galpin donde estos autores identificaron modelos mentales para recursividad empleando una metodología basada en la inspección visual de exámenes manuscritos. La metodología de análisis de protocolos verbales (grabaciones de voz mientras se realizan programas) ha permitido observar que si la intención al analizar un programa fuente es leer-para-recordar (documentar un programa), el enfoque es el texto y su forma, en cambio, si la intención es leer-para-hacer (reutilizar un programa), el enfoque es la secuencia de pasos para lograr las metas del programa y el modelo mental usado es más profundo. Lo anterior tiene directa incidencia sobre el diseño de la estrategia de enseñanza y el diseño de tareas de programación. En tal sentido, las plataformas de evaluación automática son valiosas en la medida que generan un ambiente propicio para desarrollar metodologías de aprendizaje activo que incorporen de manera efectiva las recomendaciones de las teorías de carga cognitiva y de aprendizaje.

En el artículo publicado por Karen Brennan y Mitchel Resnick (2012), los autores estudian tres posibles enfoques para la evaluación de proyectos creados en Scratch, ofreciendo ventajas y limitaciones de cada uno de ellos. El primer enfoque abarca: *Análisis del Portfolio de Proyectos*, dado que cada usuario tiene una página de perfil en Scratch donde aparecen los proyectos creados.

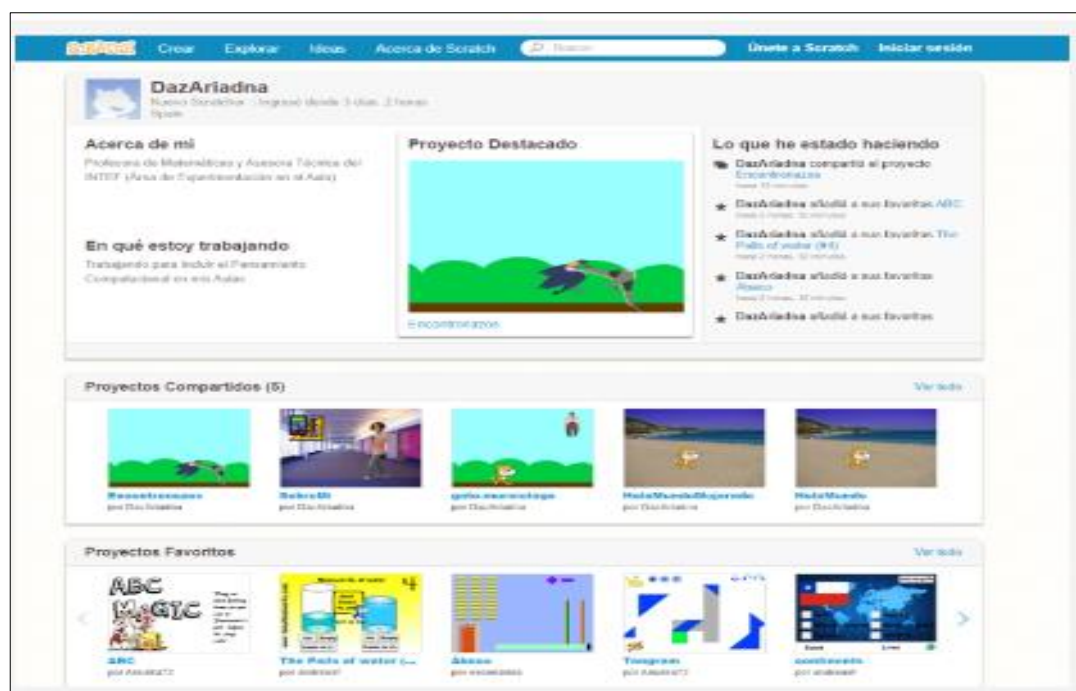


Figura 7: Perfil de un usuario de Scratch

Fuente: https://code.intef.es/prop_didacticas/estrategias-para-la-evaluacion-del-pensamiento-computacional/

En este *primer enfoque* se plantea el uso de un visualizador que analiza los bloques de programación. Una vez incluida la web para la evaluación del proyecto se obtiene un resultado con las distintas categorías a evaluar y los aspectos a mejorar.



Figura 8: Evaluación de un programa de Scratch de nivel medio realizado con Dr. Scratch
Fuente: https://code.intef.es/prop_didacticas/estrategias-para-la-evaluacion-del-pensamiento-computacional/

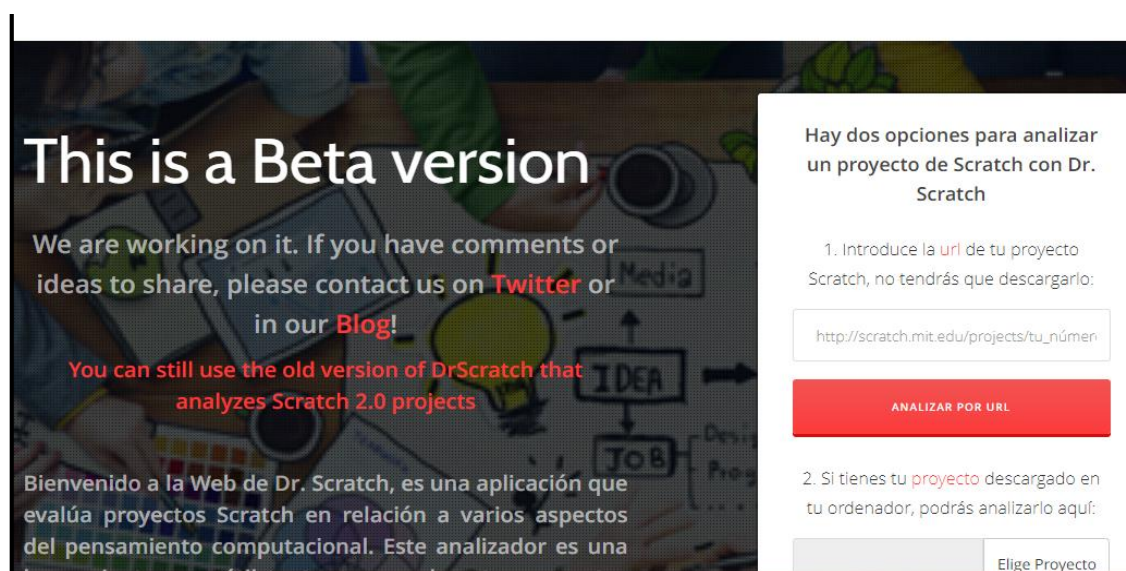


Figura 9: Pantalla del ingreso a la versión Beta de Dr. Scratch
Fuente: <http://www.drscratch.org/>

Esta metodología propone utilizar una aplicación de evaluación de proyectos denominado Dr. Scratch para el análisis de los proyectos realizados. A la cual se accede desde el sitio web: <http://www.drscratch.org/>. Al acceder a la web, solicita una URL o un archivo (url del proyecto) para su evaluación.

Esta herramienta es muy adecuada para que los estudiantes realicen una autoevaluación de sus proyectos. Asimismo, puede ser de utilidad al profesor dado que puede evaluar los proyectos de su grupo de alumnos. Al introducir la URL del proyecto se pueden comprobar los puntos frágiles y fuertes sin necesidad de revisar todos los bloques de código.

En un *segundo enfoque* estos autores proponen: *Entrevistas basadas en artefactos*, los autores realizan una investigación con entrevistas a diferentes Scratchers, que se organizan en 4 (cuatro) secciones: antecedentes del proyecto, creación del proyecto, comunidad en línea y mirando hacia delante. La sección de *creación del proyecto* es la que mayor importancia tuvo a la hora de evaluar los conceptos y prácticas del Pensamiento computacional. Al realizar estas entrevistas, detectaron fallos en el *primer enfoque*. Por ejemplo, se dieron cuenta de que los estudiantes habían reutilizado código sin llegar a entender exactamente cómo funcionaba. Las limitaciones del *segundo enfoque* fue el que conlleva a realizar las entrevistas y se hace necesario invertir una excesiva cantidad de tiempo para llevar a cabo este tipo de evaluación y sobre todo que sería recomendable realizarlo varias veces a lo largo del curso para poder comprobar avances de los alumnos.

El tercer enfoque *diseño de escenarios*, surge a raíz de las limitaciones de los dos anteriores, Este enfoque consistió en diseñar escenarios. Estos escenarios se realizaron en colaboración con investigadores del “Education Development Center” (EDC)¹¹. Este diseño de escenarios se aplicó exclusivamente en ambientes de aula. Desarrollaron tres conjuntos de proyectos en Scratch de complejidad incremental. Se presentaba cada conjunto al estudiante y se le pedía que eligiese uno de los dos proyectos. Después se realizaban

¹¹ Este centro se dedica al desarrollo del Pensamiento Computacional mediante la realización de actividades de programación en las que se usa Scratch.

preguntas sobre su funcionamiento, cómo introducir posibles mejoras y ampliaciones. Como en el enfoque anterior, este enfoque también tiene la limitación del tiempo. Además, en la evaluación sobre aspectos relacionados con el tiempo que se debe ofrecer al estudiante para resolver una de las preguntas o las ayudas que pueden llegar a ofrecérsele si se trava en algún punto.

El trabajo de García-Valcárcel-Muñoz-Repiso y Caballero-González (2019) cuyo objetivo es verificar la repercusión de las actividades de robótica educativa en estudiantes de educación infantil para la adquisición de habilidades de pensamiento computacional y programación.

Y también para desarrollar evaluaciones, pero centradas en las habilidades de PC como computación y no transversales o claves. Citamos como ejemplo el trabajo de González (2015) presenta el diseño de un Test de Pensamiento Computacional dirigido a estudiantes españoles de entre 12 y 13 años; y el de Korkmaz, Çakir y Özden (2017) que desarrolla una escala con el fin de determinar los niveles de habilidades de pensamiento computacional de los alumnos.

Lockwood y Mooney (2018) hicieron una revisión sistemática con el resumen de investigaciones del uso de Pensamiento Computacional en Secundaria dando información sobre temas como: temas utilizados como enseñanzas propias del PC o herramientas utilizadas para enseñar y evaluar PC, o beneficios y barreras de la incorporación de PC en la educación secundaria.

CAPÍTULO 3

PERSPECTIVA TEÓRICA RESPECTO A CONCEPTOS APLICADOS AL PROBLEMA DE INVESTIGACIÓN

Implicancias de resolver problemas

Como lo habíamos expuesto nuestro tema-problema es poder determinar si los recursos de la plataforma Moodle permiten medir o recabar el grado de avance en el desarrollo del pensamiento computacional de nuestros estudiantes de Ingeniería Eléctrica y saber si estos alumnos tuvieron mejor rendimiento académico que los alumnos de Agrimensura.

Para ello se debe poner en claro algunos conceptos básicos que hacen a nuestra asignatura.

La asignatura Informática¹² abarca el conocimiento de la información y como automatizar la misma. El concepto de problema es otro concepto que los alumnos deben ir incorporando a medida que transitan la materia.

Cuando hablamos de resolución de problemas mediante una computadora estamos considerando en elaborar una adecuada formulación de pasos precisos a seguir. Si pensamos en la manera o forma en que una persona indica a otra como resolver un problema, nos damos cuenta que generalmente se utiliza un lenguaje común, simple y cotidiano para realizar la explicación, a veces se mezcla con algunos vocablos o palabras técnicas.

Esto implica un gran riesgo, dado que es difícil transmitir el mensaje y con frecuencia se malinterpretan las instrucciones y por lo tanto se ejecuta incorrectamente la solución obteniéndose pasos incorrectos o erróneos. Cuando hablamos de pasos a ejecutar por una computadora las indicaciones deben ser claras no ambiguas¹³. Cada orden que se le da a

¹² Informática: Conjunto de conocimientos técnicos que se ocupan del tratamiento automático de la información por medio de computadoras.

¹³ Ambiguo: Que puede entenderse o interpretarse de diversas maneras.

la computadora debe tener una única interpretación de lo que hay que realizar. Una máquina no tiene la capacidad de decisión del ser humano para resolver alguna situación que no haya sido prevista de antemano. Si al dar una orden a la computadora se produce una situación no contemplada, será necesario cortar la tarea y volver a comenzar de nuevo el procedimiento. Además, para poder indicar claramente a la computadora mediante órdenes que es lo que debe realizar es necesario previamente entender exactamente lo que se quiere hacer. Lo fundamental o primordial es conocer con qué datos contamos y qué tipo de transformación se quiere hacer sobre los mismos.

Para poder resolver un problema utilizando una computadora como herramienta, se deben seguir algunas etapas que nos ayudarán a llegar a una solución, la misma es una tarea compleja y el proceso abarca todos los aspectos que van desde interpretar las necesidades del usuario hasta verificar que la respuesta brindada sea la correcta. Las etapas son las siguientes:

Análisis del problema: en esta etapa, se analiza el problema en su contexto del mundo real. Primeramente, deben obtenerse los requerimientos del usuario. El resultado de este análisis es un modelo claro y preciso del contexto del problema y del objetivo a resolver. Dos componentes importantes de este modelo son los *datos* a utilizar y las *transformaciones* de los mismos para obtener los resultados deseados/esperados.

Diseño de una solución: la resolución de un problema suele ser una tarea muy compleja para ser analizada como un todo. Una técnica de diseño en la resolución de problemas consiste en la identificación de cada una de las partes que componen el problema original, al que solemos llamar *sub-problemas*, y obtener la manera en que se relacionan. Cada uno de estos sub-problemas debe tener un objetivo específico, es decir, debe resolver una parte del problema original. La integración de las soluciones de los sub-problemas o partes permitirá obtener la solución buscada.

Realización de algoritmos: otro concepto fundamental que surge en esta etapa es la de algoritmos. La solución de cada sub-problema debe ser realizada a través de un algoritmo. En esta etapa se debe obtener la secuencia de pasos a seguir para resolver el problema. La elección del algoritmo adecuado al problema es fundamental para garantizar una solución eficiente.

Escritura del programa: un algoritmo¹⁴ es una especificación simbólica (diagrama de flujo o pseudocódigo) que debe convertirse en un programa en un lenguaje de programación concreto (programa fuente). A su vez, un programa escrito en un lenguaje de programación determinado (ej: python, c, c++, java, etc.) es traducido automáticamente al lenguaje de máquina de la computadora que lo va a ejecutar (programa objeto). Esta traducción que realiza la computadora se llama compilación, permite detectar y corregir los errores sintácticos que se cometan en la escritura del programa.

Verificación y depuración de errores: una vez que se tiene un programa escrito en un lenguaje de programación (programa ejecutable), se debe verificar que su ejecución produce el resultado deseado, utilizando datos representativos del problema real. Una vez ejecutado se debe comprobar que el programa cumple con los objetivos para los cuales fue creado. No obstante, es muy difícil realizar una verificación absoluta de todas las posibles condiciones de ejecución de un programa. Sin embargo, el poder verificar y depurar errores de funcionamiento del programa conducen a una mejor calidad del mismo.

Los estudiantes deben lograr apropiarse de estas etapas y ponerlas en práctica en la resolución de los problemas que se van dando a medida que avanzan con el cursado de la materia. Lo cual, una vez que lo implementan lo incorporan y lo realizan de manera automática.

¹⁴ La palabra algoritmo deriva del nombre de un matemático árabe del siglo IX, llamado Al-Khuwarizmi, quien estaba interesado en resolver ciertos problemas de aritmética y describió varios métodos para resolverlos. Estos métodos fueron presentados como una lista de instrucciones específicas (como una receta de cocina) y su nombre es utilizado para referirse a dichos métodos.

Entornos virtuales de aprendizaje

Las tecnologías de la información y la comunicación (TIC), se han innovado y presentan nuevos desafíos a las instituciones educativas de nivel superior. Aparecen nuevos ambientes de aprendizaje que favorecen la diversificación de ofertas educativas tradicionales. El uso de las TIC mediante las prácticas educativas no tradicionales ha tenido efectos significativos logrando un incremento de productividad y motivación en los estudiantes.

La educación virtual, se ha convertido en otra alternativa válida dado que posibilita vincular adecuadamente la pedagogía¹⁵ con las tecnologías, promoviendo que esta nueva didáctica se aplique satisfactoriamente y se generen contenidos educativos de calidad.

Los entornos virtuales de aprendizaje (EVEA) ofrecen múltiples beneficios como la flexibilidad en el horario de estudio, la accesibilidad desde cualquier lugar con conexión a internet, la posibilidad de acceder a recursos multimedia y herramientas interactivas, la personalización del aprendizaje según las necesidades de cada estudiante, la colaboración en línea con otros compañeros y profesores, entre otros. Facilidad de actualizar los materiales y recursos a medida que evoluciona la tecnología.

Permiten sustentar el modelo pedagógico centrado en el alumno, puesto que las herramientas tecnológicas que se utilizan, los recursos, juntamente con las estrategias de aprendizaje, requieren que el estudiante adopte un papel activo en su proceso de formación académica. Estas plataformas deben convertirse en un escenario de propuestas didácticas que resalten el protagonismo del alumno en la apropiación del conocimiento. (OCDE, 2011).

Un EVEA se define como un espacio con accesos restringidos (dado que se accede mediante un usuario y su contraseña). Está diseñado para promover la incorporación de saberes y habilidades mediante formas estructuradas de alcanzar los objetivos de

¹⁵ Práctica educativa o método de enseñanza en un terreno determinado.

La pedagogía es la ciencia que estudia la educación. El objeto principal de su estudio es la educación como un fenómeno socio-cultural, por lo que existen conocimientos de otras ciencias que ayudan a comprender el concepto de educación como, por ejemplo, la historia, la psicología, la sociología, la política.

Fuente: <https://concepto.de/pedagogia/#ixzz83tITN9Dj>

aprendizaje. A través de estos ambientes diseñados específicamente, los alumnos pueden acceder y desarrollar una serie de actividades propias del curso como si se tratara de un cursado presencial.

Las actividades dispuestas en la plataforma virtual están disponibles sin la necesidad de que exista interacción física entre los docentes y los alumnos, pudiendo establecer una comunicación a través de chats, emitir opiniones en foros, leer documentos, realizar preguntas al docente, trabajar en equipo con otros compañeros de clase, responder preguntas mediante cuestionarios, ver sus exámenes y calificaciones, pueden proporcionar retroalimentación inmediata a través de pruebas en línea, ejercicios prácticos y sistemas de evaluación automatizados entre otros. Los estudiantes pueden avanzar a su propio ritmo y revisar los materiales según sus necesidades individuales. Esto promueve el aprendizaje autodirigido y permite adaptar la experiencia de aprendizaje a diferentes estilos y ritmos de aprendizaje.

Particularmente la plataforma Moodle que utilizamos en nuestra asignatura posee los siguientes recursos disponibles, que se observan en la Figura 10.

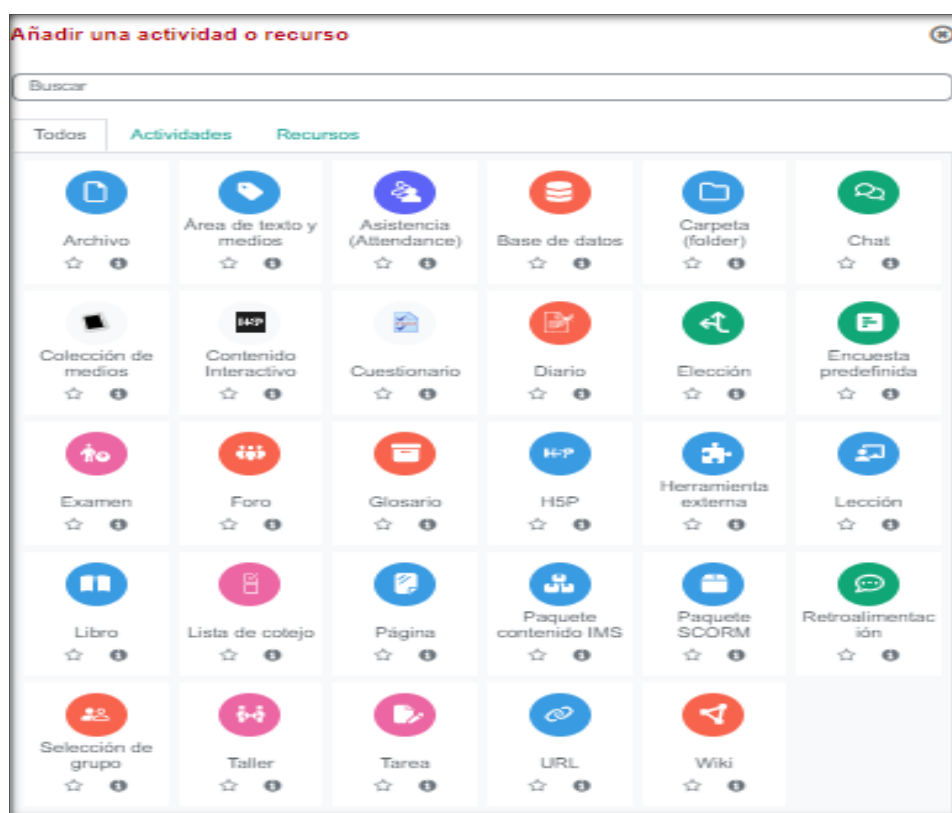


Figura 10: recursos de la plataforma virtual Moodle de la asignatura informática
Fuente: <https://virtual-moodle.unne.edu.ar/>

De los recursos de esta plataforma se utilizaron: Archivos, Área de texto y medios, Asistencia, Carpeta (para contenidos teóricos), Cuestionario. Grupos de trabajo, Tarea, URL, Foros. Herramienta externa (Padlet).

Descripción de las herramientas de Moodle¹⁶:

- Archivo: Los archivos no son sólo materiales como documentos procesados de texto o presentaciones de diapositivas. Se pueden cargar y acceder a todo tipo de archivos a través de Moodle, pero el estudiante debe tener el software correcto para poder abrirlos. Los diferentes tipos de archivos se muestran con diferentes íconos. Los archivos se pueden agregar de dos maneras: Se pueden arrastrar y soltar directamente en la página del curso o subiendo desde la computadora.
- Área de texto y medios: (anteriormente conocida como Etiqueta) un área de Texto y medios sirve como espaciador en la página de un curso de Moodle. Se puede usar para agregar texto, imágenes, multimedia o código entre otros recursos en las diferentes secciones. Es un recurso muy versátil. Allí se pueden agregar pancartas o descripciones para distinguir y resaltar diferentes áreas.
- Asistencia: Esta actividad está diseñada para que los profesores puedan tomar asistencia durante la clase y para que los estudiantes puedan ver su propio registro de asistencia.
- Carpeta: La carpeta permite al profesor mostrar varios archivos del curso juntos. Los archivos pueden ser de diferentes tipos y pueden cargarse de una sola vez, como una carpeta comprimida que luego se descomprime, o pueden agregarse uno por uno a una carpeta vacía en la página del curso.
- Cuestionario: El módulo cuestionario permite encuestar a los participantes de un curso. facilita a los profesores crear una amplia gama de preguntas para obtener retroalimentación de los estudiantes, por ejemplo, en un curso o actividades.

¹⁶ Fuente: <https://docs.moodle.org/all/es/Recursos>

- Tarea: La actividad de tarea proporciona un espacio en el que los estudiantes pueden enviar sus trabajos para que los profesores los califiquen y proporcionen retroalimentación.
- URL: Una URL (localizador uniforme o universal de recursos) es un enlace en Internet a un sitio web o archivo en línea. Los profesores pueden utilizar el recurso URL para proporcionar a sus estudiantes enlaces web para realizar investigaciones, ahorrándoles tiempo y esfuerzo al escribir manualmente la dirección.
- Foro: La actividad del foro permite a estudiantes y profesores intercambiar ideas mediante la publicación de comentarios como parte de un "hilo". Es posible que se incluyan archivos como imágenes y medios en las publicaciones del foro. El profesor puede optar por calificar y/o calificar las publicaciones del foro y también es posible dar permiso a los estudiantes para calificar las publicaciones de los demás.
- Herramienta externa Padlet: es un muro donde nuestros estudiantes escriben, suben sus trabajos, tareas y archivos en general, observan lo que comparten sus compañeros. Además, es una herramienta muy potente para trabajar de manera colaborativa. Una interfaz muy sencilla y muy cómoda visualmente, que nos permite en pocos minutos crear un espacio de aprendizaje¹⁷.

Al momento de evaluar a través de una plataforma virtual es necesario diseñar una evaluación que sea integral y permita la reflexión por parte de los alumnos. Se debe integrar los recursos disponibles en la plataforma con los objetivos formativos de la evaluación. Como habíamos establecido las concepciones desde las que se aborda el aprendizaje *“aprendizaje centrado en el alumno”*, el alumno como gestor y centro del mismo. Por consiguiente, la evaluación *“se centraliza en el aprendizaje y no en el rendimiento”*, se busca el aprendizaje de la metodología de resolución de problemas paso a paso por medio de algoritmos y su programación en la computadora.

Consecuentemente, se debe tener en claro la actividad evaluativa y elegir el recurso adecuado a esa necesidad. Evaluar la adquisición del pensamiento computacional implica

¹⁷ Fuente: https://intef.es/observatorio_tecno/padlet/

utilizar algunos recursos de esta plataforma virtual, pensar de manera más integral en cómo el alumno desarrolla su capacidad de pensar lógicamente al resolver un problema computacional. Si es claro su planteo, si logra determinar los datos con los que cuenta (entradas), cuál es la salida esperada, cómo plasma la solución algorítmica (pasos precisos), en un diagrama de flujo. Si aplica correctamente el lenguaje de programación para escribir el código del programa, compilarlo y ejecutarlo y llegar a la solución esperada.

Como herramienta evaluativa de la plataforma Moodle, en la asignatura, se utilizaron cuestionarios con preguntas teóricas, para recabar datos acerca de los conocimientos conceptuales alcanzados luego de desarrollar cada tema. Por otra parte, luego del desarrollo de la guía de trabajos prácticos se realizaban otros cuestionarios con ejercicios prácticos que eran evaluativos y los estudiantes tenían la posibilidad de recuperar el mismo en caso de no aprobarlos. Esto en general, se aplicaba a los alumnos de todas las comisiones de Ingeniería (Eléctrica, Electrónica y Agrimensura). Los cuestionarios nos permitieron dar cuenta de los alumnos que rendían adecuadamente los mismos y a los que les resultaba más difícil sacar una nota satisfactoria, y la cantidad de alumnos debían recuperar los mismos.

A la hora de evaluar el pensamiento computacional y su implicancia en la programación se tiene en cuenta que lo veníamos realizando en la asignatura progresivamente es decir se da por el tipo de ejercicio que el alumno debe resolver en cada etapa del aprendizaje, el cual comienza con ejercicios de tipo *secuencial* (un paso seguido de otro en secuencia), luego los ejercicios se van complejizando con la incorporación de otras estructuras como la de *selección*, donde el alumno aprende a tomar decisiones lógicas (es decir a seleccionar un camino u otro del programa) dependiendo de los resultados esperados según el problema planteado. A medida que avanza el cursado también debe aplicar las estructuras *iterativas* o *repetitivas* es decir ejecutar varias veces un conjunto de instrucciones dependiendo de una condición lógica. Finalmente, el alumno con todo el conocimiento adquirido debe resolver un problema integral, complejo pero cotidiano donde se aplica todo lo dado.

En la asignatura también se incorporó el recurso externo Padlet (muro) que se utilizó dentro del aula virtual de la asignatura para que los estudiantes pudieran compartir sus producciones (ejercicios de las guías) de trabajos prácticos y favorecer un intercambio de opiniones entre pares.

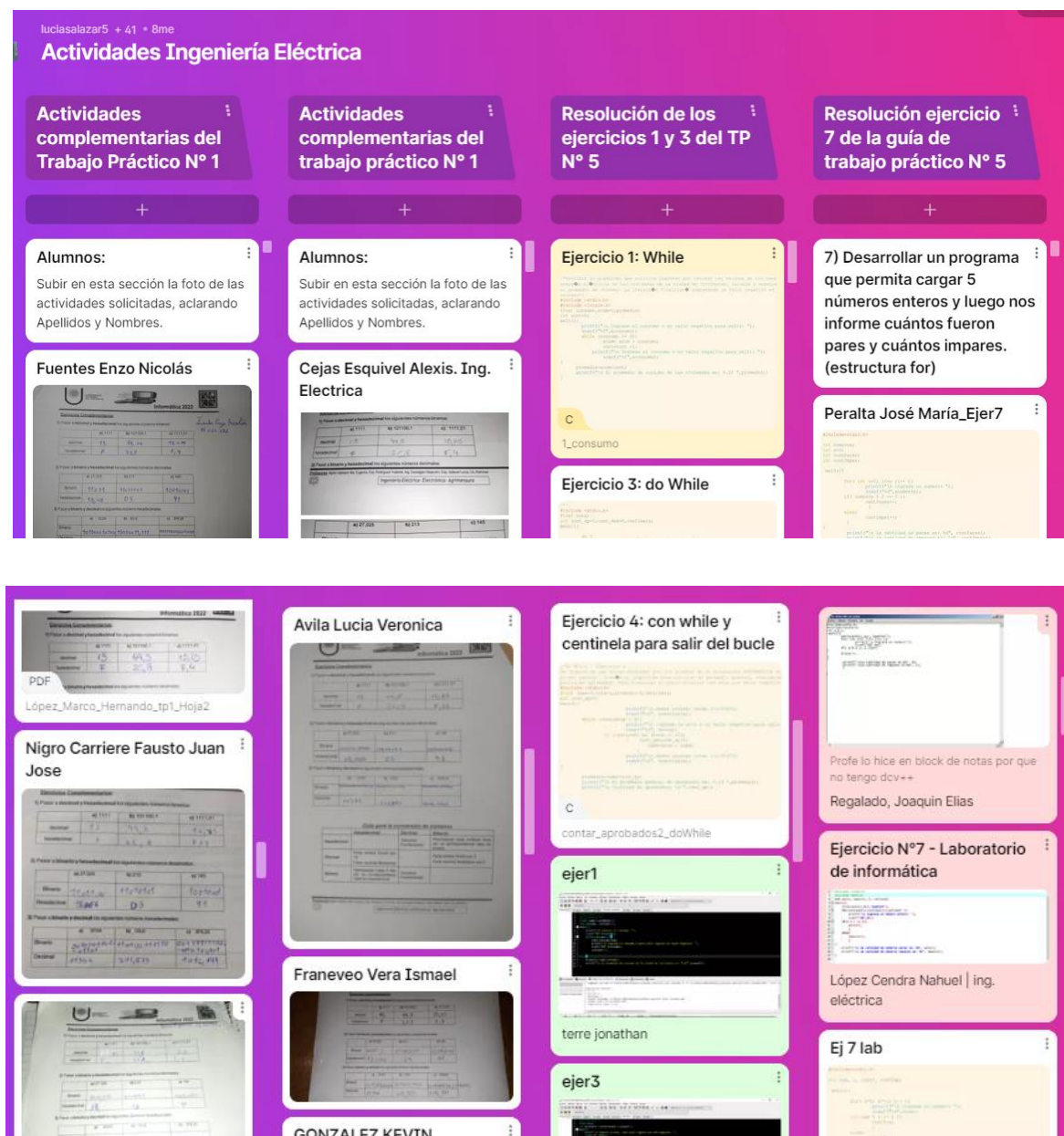


Figura 11: recorte del Padlet utilizado dentro del aula de la asignatura informática
Fuente: elaboración propia-<https://padlet.com/luciasalazar5/actividades-ingenier-a-el-ctrica->

CAPÍTULO 4

ASPECTO METODOLÓGICO

Introducción

En cuanto a la metodología para este trabajo se aplica la investigación experimental¹⁸ a través de la formación de grupos experimentales donde se tiene la posibilidad de manipular la realidad a través del control de todas las variables¹⁹ (independientes, dependientes, moderadas, extrañas, etcétera).

La investigación experimental es diferente a otros tipos de investigación dado que el objetivo de estudio y su método dependen del investigador que realiza la investigación, así como las decisiones que establezca el investigador para llevar a cabo el experimento. Durante el desarrollo del experimento o formación de grupo experimental se manipulan de manera voluntaria las variables y se observan los resultados en un ambiente controlado. Se realizan repeticiones de cada paso o experimento para así verificar determinadas hipótesis realizadas por el investigador. Esto se puede realizar en un laboratorio o en el campo propiamente dicho. (Arias F. (2012)).

Según Fidas G. Arias (2012)²⁰, *“la investigación experimental es un proceso que consiste en someter a un objeto o grupo de individuos en determinadas condiciones, estímulos o tratamiento (variable independiente), para observar los efectos o reacciones que se producen (variable dependiente)”*.

Otra característica de este tipo de investigación es que se establecen grupos de control. En particular deben disponerse de dos (2) grupos. Uno en el que no se modifiquen

¹⁸ El diseño de la investigación experimental se utiliza para establecer una relación entre la causa y el efecto de una situación. Es un diseño de investigación donde se observa el efecto causado por la variable independiente sobre la variable dependiente.

¹⁹ Una variable de investigación, es un término que se utiliza para referirse a cualquier tipo de relación de causa y efecto. En términos generales, una variable representa un atributo medible que cambia a lo largo de un experimento comprobando los resultados. (fuente: <https://www.questionpro.com/blog/es/diseño-de-investigación/>).

²⁰ Fidas G. Arias es autor del libro “El Proyecto de Investigación”. Introducción a la metodología científica. Editorial Episteme.

los factores ni las variables y otro en el cual se realiza la manipulación. De este modo es factible observar los resultados en ambos grupos y poder identificar las diferencias. Esto facilita cuantificar el cambio provocado por el proceso experimental y avala la posibilidad de verificar las variaciones en los grupos de las variables del experimento.

La investigación experimental se corresponde principalmente con un proceso de investigación cuantitativo. Esto se debe a que en este tipo de investigación se manipulan variables independientes de manera controlada para observar su efecto en una variable dependiente, y los datos recopilados suelen ser de naturaleza numérica, lo que facilita su análisis cuantitativo mediante métodos estadísticos.

Sin embargo, en algunas ocasiones la investigación experimental puede integrar elementos cualitativos, especialmente en lo que se refiere a la interpretación de los resultados o la comprensión de los procesos subyacentes. Por ejemplo, después de realizar un experimento, los investigadores pueden entrevistar a los participantes para obtener información cualitativa sobre sus experiencias y percepciones. También, en ciertos casos, los investigadores pueden combinar enfoques cualitativos y cuantitativos en un diseño de investigación mixto.

Caso de Estudio

Para nuestro caso el objeto de estudio o grupo experimental son los alumnos que cursan la asignatura informática pertenecientes al primer año de la Ingeniería eléctrica, siendo el grupo de control los alumnos del primer año de la carrera de ingeniería en agrimensura, que no estarán expuestos al experimento, pero que cursan la misma asignatura.

El experimento se realizó durante la cursada de la asignatura informática en el segundo cuatrimestre del año 2021 en la FaCENA-UNNE, con una muestra de treinta alumnos del primer año de la Ingeniería eléctrica. Durante el cursado de la misma se aprovecharon los recursos TIC existentes en la plataforma Moodle de la asignatura la cual cuenta con varias herramientas que fueron investigadas y puestas a prueba para

aprovecharlas convenientemente durante el cursado. Las herramientas de Moodle investigadas fueron los cuestionarios, tareas y foros los cuales tienen las siguientes características:

Cuestionarios permiten:

- Evaluar el conocimiento adquirido por los estudiantes.
- Realizar pruebas de autoevaluación.
- Recopilar información para retroalimentar el proceso de enseñanza-aprendizaje.
- Crear exámenes con preguntas de opción múltiple, verdadero/falso, respuestas cortas, ensayos, etc.
- Establecer fechas límite para completar el cuestionario.

Tareas facilitan:

- Recopilar trabajos individuales o grupales de los estudiantes.
- Evaluar habilidades específicas como la escritura, la investigación, la resolución de problemas, entre otras.
- Establecer fechas límite para la entrega de las tareas, ajustándose a la presentación de trabajos en tiempo y forma.
- Permitir la retroalimentación personalizada por parte del docente.

Foros ofrecen:

- Facilidad de discusión sobre temas e intercambio de ideas entre los estudiantes.
- Fomentar la participación activa de los estudiantes en temas específicos.
- Crear un espacio para que los estudiantes planteen dudas y resuelvan inquietudes.
- Realizar debates sobre temas relevantes de la asignatura.

Algunos de los **criterios** a considerar fueron:

Los *objetivos de aprendizaje*: dependiendo de si se buscará evaluar conocimientos, habilidades prácticas o fomentar la discusión y colaboración entre los estudiantes, se elegirá el tipo de actividad y herramienta más adecuado.

El *tipo de evaluación*: se tendrá en cuenta si la actividad requiere una evaluación cuantitativa como un examen, el cuestionario podría ser la mejor opción. Si lo que se busca evaluar son los ejercicios prácticos o trabajos más extensos, una tarea podrá ser más apropiada. Y el recurso foro facilitará fomentar la discusión e interacción entre estudiantes.

La *Interactividad*: permitirá establecer la interacción que se desea establecer entre los estudiantes y el docente, así como entre los propios estudiantes, se podría optar por una actividad donde se promueva dicha interacción.

El *formato de entrega y la retroalimentación*: se tendrá en cuenta la necesidad de que los estudiantes entreguen sus programas (algoritmos codificados) para su evaluación posterior, por lo tanto, el recurso tarea podría ser la mejor opción.

El *tiempo y recursos disponibles*: también es importante considerar el tiempo y los recursos disponibles para la creación y la evaluación de actividades. Algunas actividades, como los cuestionarios se programan para estar abiertos en un día y horario determinado luego del cual se cierran, el alumno podrá realizarlo y tener derecho a un recuperatorio en caso de ser necesario.

Parte de la investigación en este sentido, consistió en armar y probar con los estudiantes cada recurso para observar los resultados obtenidos si eran los esperados en base al desarrollo de los temas y ejercicios prácticos. Se utilizaron cuestionarios de preguntas y respuestas, mediante un banco de preguntas de tipo opción múltiple, respuesta corta, respuestas anidadas, verdadero-falso y la opción ensayos, como así también se procedió a la selección y secuenciación de las actividades que faciliten recabar esa información de nuestro caso de estudio.

El recurso cuestionario, permitió crear categorías de preguntas, según los temas que se desarrollan en la asignatura, para disponer de un banco de preguntas, el cual es una herramienta que permite a los docentes: crear, previsualizar, editar y organizar preguntas. Una vez cargadas todas las preguntas por temas se armaron los cuestionarios, por un lado, para los temas teóricos y luego para los ejercicios prácticos, para este último se utilizaron preguntas del tipo arrastrar y soltar para completar los algoritmos según los enunciados. La facilidad que tiene este tipo de cuestionarios es que después de cargar todas las preguntas, se pueden organizar las mismas según el orden en el que se desea que aparezcan, se puede arrastrar y soltar las preguntas para reorganizarlas según sea necesario. lo que favoreció llevar a cabo diferentes tipos de ejercicios de las guías de trabajos prácticos y observar los resultados que arrojaron luego de haberlos realizado los alumnos, y así analizar la interpretación que tuvieron de los ejercicios.

El recurso tarea de la plataforma Moodle se utilizó para que los alumnos subieron la resolución del ejercicio desarrollado durante la clase práctica, para que luego de su

revisión por parte del docente se le brinde una devolución. Es un recurso de fácil utilización donde se puede destinar un lapso de tiempo para que el alumno suba su trabajo y sea calificado. El recurso foro se utilizó para facilitar el intercambio de ideas y fomentar la participación de los estudiantes a medida que se avanzaba en los temas de la materia.

Metodología de seguimiento

En la actualidad no se encuentran en el ámbito universitario instrumentos estandarizados de medición de las habilidades adquiridas en la programación y por ende en la adquisición del pensamiento computacional. Lo que planteó la necesidad de elaborar una metodología propia que se ajustara a los objetivos de investigación del presente trabajo.

Las lecturas bibliográficas referentes a diferentes autores como: *García-Valcárcel-Muñoz-Repiso y Caballero-González (2019)*, *Cubero-Ibáñez, J., Ibarra-Sáiz, M.S. y Rodríguez-Gómez, G. (2018)*, *Karen Brennan y Mitchel Resnick (2012)*. Dan cuenta de metodologías para recabar información del grado de avance en la adquisición del pensamiento computacional, que si bien algunas de esas metodologías se utilizan para medir este tipo de pensamiento mediante la utilización de problemas a resolver que luego son evaluados mediante rúbricas (Cubero-Ibáñez, J., Ibarra-Sáiz, M.S. y Rodríguez-Gómez, G. (2018)), si bien algunos marcados sirvieron de base para nuestro trabajo porque en ellos menciona (comprensión del problema, planificación y ejecución, revisión y transferencia), estos son marcadores válidos dado que están comprendidos dentro de las fases de resolución de un problema en programación. Los demás autores vistos hablan de mediciones poco estandarizadas, y muy generales u orientadas al trabajo con ciertos grupos etarios y al uso de alguno software de programación muy específicos como (scratch). Las mismas, por lo tanto, no se ajustarían en su totalidad a lo que queremos obtener como marcadores válidos. Por lo tanto, no pudimos adaptar las mismas a nuestra población o muestra que son los estudiantes de Ingeniería eléctrica de la FaCENA-UNNE. Teniendo en cuenta esto, se elaboró la siguiente metodología teniendo en cuenta a nuestros estudiantes,

el programa que deberíamos desarrollar en la asignatura, los ejercicios de las guías de trabajos prácticos, los resultados esperados y los conocimientos previos de nuestros alumnos de ingeniería eléctrica.

La evaluación de habilidades en programación consiste en medir la capacidad que tiene el estudiante para resolver problemas de programación utilizando diferentes lenguajes y herramientas (software de programación instalables, plataformas de programación online, uso de computadoras y aplicaciones en celulares, tabletas). Este tipo de evaluación puede incluir ejercicios prácticos, planteamiento y resolución de problemas con diferentes niveles de dificultad, evaluaciones escritas y pruebas en línea. Se debe tener en cuenta la manera en la que se realiza la evaluación (si serán pruebas individuales o por grupos, concursos y otorgamiento de puntajes, observación directa de la realización de ejercicios en la computadora o celular). En el caso de los instrumentos se tienen en cuenta: el diseño del algoritmo, escribir el código del programa, ordenar diferentes líneas de código desordenadas, rellenar huecos o completar frases en programas incompletos; plantear algoritmos que resuelvan un problema en el menor tiempo posible entre otros.

Teniendo en cuenta las habilidades de (leer y entender el problema, diseñar el algoritmo que resuelva el problema que se plantea, escribir el código, ordenar las instrucciones del programa, ejecutar y probar la efectividad del mismo), que deben desarrollar los estudiantes en la adquisición del pensamiento computacional. Se desarrolló la siguiente metodología adaptada a nuestros alumnos de ingeniería eléctrica.

Propuesta para el seguimiento del avance del pensamiento computacional de estudiantes de Ingeniería Eléctrica

Se pensó en una metodología aplicable en tres etapas:

- 1) Una primera etapa de diagnóstico al inicio del cursado que serviría para sentar las bases de los conocimientos de nuestros estudiantes y nos permitirá mejorar el desarrollo de los temas de la materia y ver el punto de partida.

- 2) Una segunda etapa donde se realizarán los ejercicios prácticos y se generarán datos del proceso de enseñanza-aprendizaje de los estudiantes.
- 3) Una tercera etapa donde trabajarán colaborativamente para medir la capacidad del estudiante de trabajar en equipo y en colaboración con sus pares, durante la realización de los ejercicios que serán seleccionados de las guías de trabajos prácticos siguiendo el programa de la asignatura informática.

Cada etapa con su correspondiente evaluación mediante la elaboración de rúbricas con los marcadores propuestos a fin de obtener esas mediciones que den cuenta de cómo los alumnos avanzan a medida que adquieren y desarrollan el pensamiento computacional. A continuación, se detallan las etapas:

Primera etapa de diagnóstico:

- 1) Como **primera etapa**, se implementará una encuesta inicial al comenzar el cursado de la asignatura informática entre los alumnos ingresantes a la carrera de Ingeniería Eléctrica. La misma tiene como objetivo averiguar *¿qué conocimientos tienen los alumnos de ingeniería eléctrica en informática y programación?*
- 2) Se recopilarán datos del grado de conocimiento que los alumnos traen de su paso por la escuela secundaria en los **temas relacionados** a la informática, pensamiento lógico y la programación.
- 3) Con la encuesta se logrará tener en claro cómo se desarrollarán los conceptos fundamentales de la materia teniendo en cuenta los conocimientos previos de los estudiantes. Es decir, qué conceptos se explicarán primero en las clases de teoría y práctica, que denote la encuesta, falte reforzar, y que conceptos se pueden dejar para el final, dado que la mayor parte de los estudiantes ya posee conocimiento.
- 4) Los temas que formarán parte de la encuesta serán: definición de computadora: funciones. Dato e Información. Problema: definición. Algoritmos: resolución. Programa. Ejemplos prácticos.

La encuesta se desarrollará de manera virtual mediante un enlace del Goolge forms que estará disponible desde el aula virtual de Moodle. Antes del inicio de la primera clase práctica. En la asignatura se dispone de la plataforma virtual Moodle donde nuestra materia posee un espacio al que denominamos aula virtual. Figura 11.

- 5) En el **anexo I** se detalla el formato de la encuesta que estará disponible en el siguiente enlace del formulario de Google:

https://docs.google.com/forms/d/e/1FAIpQLSeyTFhEuPR3s7tSruqWtZyETL8lqP4Exoql74Doa94A_koEjw/viewform

Con la encuesta se podrá

- 6) Como se expuso al comienzo de este trabajo, la asignatura se dictará de manera presencial la parte teórica y el desarrollo de las guías de trabajos prácticos se realizará una parte en aula y otra en los laboratorios de programación donde el alumno dispondrá de una computadora o podrá llevar su notebook para realizar en máquina las soluciones a las que debió llegar previamente en el aula de práctica.



Figura 11: Pantalla de del aula virtual de la asignatura informática en Moodle

- 7) Otra herramienta que se utilizará en esta primera etapa será el recurso **foro** dentro del aula virtual de la asignatura, a efectos de promover la participación efectiva de los alumnos de manera que logren afianzarse en su manejo, fomentará la comunicación escrita, conocerán las opiniones de sus pares y opinarán según sus conocimientos, pondrán en práctica de escritura de textos cortos (concisos y concretos). Los temas presentados en el foro estarán destinados a recabar actitudes frente a la programación en temas de debate tales como, el “*planteo de situaciones problemáticas de la vida cotidiana*”, donde realizarán una tarea paso a paso para obtener un resultado esperado o pensar lógicamente teniendo que elegir entre una acción u otra.
- 8) Con esta actividad se pretende evaluar la resolución de problemas simples de la vida cotidiana. Fomentar el intercambio de conocimientos entre estudiantes con diferentes niveles de conocimiento en el tema.

Actividad: Dada la siguiente situación:

Suponga que se planea una salida con amigos. La salida depende del clima: si llueve vos y tus amigos irán al cine a ver la película elegida, por el contrario, si no llueve irán de pesca. Luego de realizar el paseo se juntarán a comentar la experiencia vivida en la casa de uno de tus amigos. Escriba el algoritmo que resuelva esta situación.

Tener en cuenta que deberás expresar las acciones o pasos de manera ordenada y con frases cortas y concretas. Se deberá vislumbrar la diferencia entre lo que se debe hacer según cada circunstancia.

Compartirás tu solución en el foro y compartirás algún planteo similar de la vida cotidiana.

En la sección **Resultado** se comparten algunas de las respuestas por parte de los alumnos a este planteo.

Por otra parte, al grupo de control que son los alumnos de la Ingeniería en Agrimensura, y que no estuvieron expuestos al experimento. No se les realizó la encuesta inicial y tampoco se utilizó el foro del aula virtual. Sin embargo, realizaron esta situación problemática en el aula de clases donde cada uno mostró al docente la respuesta a dicha solución. Fue un trabajo individual que no se compartió al resto del grupo de compañeros.

Segunda etapa: Ejercitación

- 1) La **segunda etapa** consistirá en la realización de diferentes ejercicios prácticos que serán seleccionados de las guías de trabajos prácticos, estos ejercicios abarcarán los tipos de estructuras de control (secuencial-selectivas-repetitivas). Los mismos se realizarán durante tres (3) clases en aula de manera presencial.
- 2) Se buscará promover la ejercitación mediante la realización de diferentes tipos de ejercicios en hoja, donde cada alumno deberá seguir los pasos de la programación estructurada: *leer y entender el problema, análisis del problema, desarrollo de la solución, escritura del algoritmo, programar la solución, depuración y prueba.*
- 3) Representarán la solución del algoritmo en alguna de las herramientas vistas (diagrama de flujo o pseudocódigo). El mismo lo realizará en papel.
- 4) Luego ~~que~~ de plantear dicha solución se les pedirá argumentar (de forma escrita), como pensó en esa solución es decir (deberá determinar cuáles fueron las entradas del problema, que se debería realizar en el proceso y qué resultados se deberían mostrar como salida del algoritmo).
- 5) Para comprobar la veracidad de la solución planteada, el alumno pondrá en práctica la programación. Deberá programar dicho algoritmo en la computadora o en el celular mediante su codificación en el lenguaje C (software dado en la asignatura), la solución que cada uno consiguió realizar.
- 6) Por último, deberá compilar y ejecutar el código (ejercicio escrito en el editor de códigos del lenguaje C), con la finalidad de comprobar que el resultado obtenido se corresponde con la solución esperada. En caso de no ser así se procederá a la depuración de errores de lógica u errores de sintaxis si los hubiere, para lograr el resultado correcto del problema.
- 7) Antes de cada asistencia a clase práctica en aula, el alumno deberá responder un cuestionario teórico mediante preguntas estilo múltiple choice o combinación de las

mismas (en el aula de Moodle), referentes al tema a tratar en la guía de trabajo práctico, que propiciará la lectura del material teórico disponible en el aula virtual.

8) En el **anexo III** se comparten algunos de los cuestionarios realizados en la plataforma Moodle del aula virtual de la asignatura.

Los ejercicios seleccionados de la guía de trabajo práctico para esta etapa serán los siguientes:

Ejercicio 1:

- La FaCENA recibió una donación especial, la cual sería repartida entre las tres carreras de las ingenierías que son: Ingeniería Eléctrica, Ingeniería en Electrónica e Ingeniería en Agrimensura y también se destinó un último porcentaje de dicha donación al Profesorado de Matemática. Siendo los porcentajes distribuidos de la siguiente manera:
 - Ingeniería Eléctrica: 30%
 - Ingeniería Electrónica: 15%
 - Ingeniería en Agrimensura: 20%
 - Profesorado de Matemática: lo que resta de la donación
- Diseñar una solución aplicando los conocimientos vistos sobre algoritmo, para determinar cuánto recibirá cada carrera según esa donación que recibió la FACENA y que se ingresa previamente por el teclado.

Ejercicio 2:

- En el negocio conocido como “Materiales eléctricos Ctes. S.A.”. Se presentó Juan a fin de comprar varios metros de cable de 2,5 mm para la conexión eléctrica de un velador de su dormitorio. Teniendo en cuenta que el costo del metro de cable está \$300. Se pide diseñar la solución en un algoritmo que calcule cuánto deberá pagar Juan por su compra. Teniendo en cuenta que el dueño del negocio le realizará un descuento del 10% si la compra supera los \$1000. La solución a este problema deberá poder mostrar en pantalla el costo total que debe pagar Juan según la cantidad de cable adquirido con el descuento o sin descuento en caso de cumplirse la condición establecida por el dueño del local.

Ejercicio 3:

- En una empresa de transporte de cereales trabajan 15 empleados, cuyos sueldos oscilan entre \$20000 y \$80000, se desea realizar un programa que lea los sueldos que cobra cada empleado e informe cuántos empleados cobran entre \$20000 y \$50000 y cuántos cobran

más de \$45000. Además, el programa deberá informar el importe total que gasta la empresa en sueldos a todo el personal de la misma.

9) Teniendo en cuenta estos ejercicios se diseñó una rúbrica donde se seleccionaron marcados válidos, los cuales permitirán comprobar en esta etapa como los estudiantes alcanzaron la solución a los ejercicios y poder de esta manera comprobar el alcance logrado en el desarrollo del pensamiento computacional. Tal lo expresado por Arends (2004, p.248), *“...Para que la evaluación auténtica sea realmente efectiva se requiere que estos principios se adapten a los alumnos, al igual que sucede con el tipo y nivel de trabajo que estos desarrollen. Del mismo modo, los alumnos deben conocer previamente –y en detalle– cuáles serán los criterios de evaluación que apliquemos”*.

Estos criterios se dispondrán previamente en el aula virtual para que los estudiantes se familiaricen con la forma en que serán evaluados en estos ejercicios.

Los marcadores válidos que se mencionan en la Tabla- 4, surgieron de la necesidad de evaluar diferentes aspectos claves en el proceso de resolución de problemas mediante la programación, ellos son:

Pensamiento algorítmico: este marcador evaluará la capacidad del estudiante para diseñar algoritmos eficientes y lógicos que resuelvan el problema planteado. Se enfocará en la habilidad del estudiante para descomponer el problema en pasos claros y precisos, mostrando un pensamiento estructurado y ordenado.

Pensamiento crítico: el marcador se centrará en la capacidad del estudiante para analizar, evaluar y mejorar su propio proceso de resolución de problemas. Evaluará la habilidad del estudiante para cuestionar suposiciones, identificar posibles mejoras y tomar decisiones fundamentadas durante el proceso de programación.

Elaboración del algoritmo: con este marcador se evaluará la habilidad del estudiante para detallar y planificar de manera precisa el algoritmo antes de comenzar a programar. Se buscará verificar si el estudiante puede describir claramente los pasos necesarios para resolver el problema, mostrando un entendimiento profundo de la lógica subyacente.

Programación-ejecución: este marcador se enfocará en la capacidad del estudiante para traducir el algoritmo diseñado en un código funcional y ejecutable. Evaluará la destreza del estudiante en la implementación práctica del algoritmo utilizando un lenguaje de

programación específico, asegurando que el código sea correcto y cumpla con los requisitos del problema.

Prueba: evaluará la capacidad del estudiante para probar y depurar el programa desarrollado, identificando y corrigiendo posibles errores o fallos. Se buscará verificar si el estudiante puede validar la funcionalidad del programa, asegurando que este produzca los resultados esperados en diferentes escenarios.

Creatividad e innovación: este marcador en el contexto de la programación se enfocará en evaluar la capacidad del estudiante para pensar de manera original, proponer soluciones novedosas y aplicar enfoques innovadores en la resolución de problemas.

Marcadores	Muy bueno	Bueno	Regular	Malo
Pensamiento algorítmico ²¹	Logra realizar cada etapa solicitada paso a paso y argumenta claramente la solución encontrada.	Logra realizar las etapas solicitadas pero no argumenta con claridad la solución encontrada.	logra realizar algunas etapas y argumenta en forma regular la solución.	No logra realizar las etapas solicitadas y argumenta con muy poca claridad o no argumenta.
Pensamiento Crítico ²²	Realiza deducciones lógicas y optimiza la solución correctamente.	Realiza algunas deducciones lógicas y optimiza la solución correctamente.	Realiza algunas deducciones lógicas, y la solución es medianamente óptima.	No realiza deducciones lógicas y no encuentra una solución óptima.
Elaboración del algoritmo	Elabora el algoritmo correctamente (el mismo tiene inicio y final, los pasos son correctos no son ambiguos)	Elabora el algoritmo correctamente (el mismo tiene inicio y final pero le faltan algunos pasos aunque no son ambiguos)	Elabora el algoritmo con algunos errores (el mismo tiene inicio y final pero le faltan algunos pasos aunque no son ambiguos)	No elabora el algoritmo correctamente (el mismo tiene inicio y final pero los pasos no son correctos, no llega a la solución esperada)
Programación, ejecución y prueba	Logra programar correctamente las instrucciones que refleja el algoritmo. Compila, ejecuta y prueba correctamente. Sabe reconocer, corregir errores de programación y lógica	Logra programar la mayoría de las instrucciones que refleja el algoritmo. Logra compilar y ejecutar pero arroja algún error de compilación, el alumno logra corregirlo y llega finalmente a la solución.	Logra programar las instrucciones, pero posee algunos errores de compilación y lógica. No sabe reconocer y corregir errores de programación y lógica.	No Logra programar correctamente las instrucciones que refleja el algoritmo. Al no conocer la sintaxis del programa no compila correctamente, no logra corregir errores y no puede ejecutarlo.

²¹ El pensamiento algorítmico es la capacidad para realizar el proceso de abstracción, modelización de un problema, deducciones lógicas y síntesis de la solución que conduzca a escribir el algoritmo correcto. (fuente: Cátedra de Conceptos de Algoritmos y Programa. UNLP, 2016)

²² pensamiento crítico: aquel que trata de acercarse a la verdad desde un punto de vista propio, gracias a él podemos detectar y resolver problemas.

Creatividad e Innovación ²³	Se vislumbra en todo momento la creatividad en la solución, claridad e innovación porque el algoritmo es claro, preciso y eficiente	Se vislumbra claridad en la solución, no es innovadora pero es adecuada al problema.	La solución es algo creativa, poco innovadora, poco clara y medianamente eficiente.	No refleja creatividad al momento de armar una solución al problema. No es claro, no es preciso ni eficiente
--	---	--	---	--

Tabla 4: marcadores de evaluación del PC para ejercicios de programación
Fuente: elaboración propia

El grupo de control en esta etapa, realizó los mismos ejercicios que se mencionan para los estudiantes de ingeniería eléctrica, los realizaron en el aula de clases prácticas en papel. Luego en el laboratorio de programación, programó la solución en la computadora. Su trabajo fue observado por el docente de manera procesual durante su desarrollo.

No realizaron previamente el cuestionario del Moodle propuesto para los alumnos de ingeniería eléctrica.

Tercera etapa: trabajo colaborativo

1) En esta **tercera etapa** se procederá a hacerlos trabajar en grupos de dos o más alumnos, con el fin de que sean ellos los que elaborarán los enunciados de un problema de la vida cotidiana y también la solución. Deberán codificar el algoritmo en el lenguaje de programación, compilar el mismo para comprobar y verificar errores, corregir en caso de ser necesario y finalmente los presentarán a sus compañeros en el salón de clases. Se pensó en que fueran los alumnos gestores de su propio aprendizaje. Tomando como base la evaluación auténtica la cual *“Permite confrontar el aprendizaje –usando criterios del mundo real– con cuestiones como el manejo y solución de problemas intelectuales y sociales, los roles desempeñados, las actitudes y valores mostrados, las formas de interacción y cooperación entre participantes, así como las habilidades profesionales o académicas adquiridas o perfeccionada”*²⁴.

²³ Permite generar la capacidad para crear soluciones innovadoras, propias para diferentes problemas en programación utilizando distintos lenguajes o herramientas de programación

²⁴ REVISTA IBEROAMERICANA DE EDUCACIÓN. N.º 64 (2014), pp. 11-25 (ISSN: 1022-6508)

- 2) Cada grupo de 2 (dos) o más alumnos deberán desarrollar la solución por medio de algoritmos en diagrama de flujo o pseudocódigo. Programar el algoritmo en la computadora o celular, compilar y ejecutar para probar con un lote de prueba de datos reales, si el mismo logra llegar a la solución del problema planteado.
- 3) Finalmente, cada grupo deberá compartir la solución en un muro de Padlet que se creó e incrustó en el aula virtual para que los grupos de alumnos pudieran visualizar el trabajo en conjunto de todos los grupos. Permitiéndose comentar el trabajo de los demás y aprender de los pares.
- 4) La evaluación de este último paso también se realizó utilizando rúbricas.

Los marcadores válidos que se mencionan en la Tabla- 5, surgieron de la necesidad de evaluar diferentes aspectos claves en el proceso de resolución de problemas mediante la programación y el trabajo colaborativo, ellos son:

Planteamiento mediante algoritmos: referirá a la capacidad de identificar problemas cotidianos que puedan ser resueltos mediante algoritmos. Es fundamental en el proceso de evaluación porque demuestra la habilidad del estudiante para analizar situaciones de la vida real y traducirlas en pasos lógicos y estructurados que puedan ser programados.

Pensamiento Crítico y programación: Este marcador destaca la importancia de analizar, evaluar y mejorar los algoritmos propuestos. En el contexto de evaluación, se busca que el estudiante sea capaz de cuestionar sus propias soluciones, identificar posibles mejoras y optimizar el rendimiento de los algoritmos.

Trabajo en Equipo: La capacidad de trabajar en equipo es esencial en la programación, ya que muchos proyectos requerirán la colaboración de diferentes personas con habilidades complementarias. En este contexto, se evaluará la capacidad del estudiante para comunicarse, colaborar y coordinar esfuerzos con otros para lograr objetivos comunes.

Comunicación: los programadores deben ser capaces de explicar sus ideas, soluciones y resultados de manera clara y concisa. Es por ello que se buscará que el estudiante pueda

expresar sus pensamientos de forma coherente y comprensible tanto en la presentación de los algoritmos como en la explicación de su funcionamiento.

Presentación: se valorará la capacidad del estudiante para organizar y estructurar la información de forma visualmente atractiva y comprensible para diferentes audiencias.

Marcadores	Alto	Medio	Bajo
Enunciar problemas cotidianos que se resuelven mediante algoritmos.	Elabora adecuadamente el problema con claridad y se encuadra dentro de una estructura de control vista	Elabora el problema, pero el mismo es poco claro, aunque se encuadra en alguna estructura de control vista	No puedo elaborar claramente un problema de la vida cotidiana que contenga una estructura de control vista
Planteamiento mediante algoritmos	Los estudiantes son capaces de seguir instrucciones básicas paso a paso mediante un algoritmo	Los estudiantes demuestran dificultades para plantear un problema paso a paso o más complejos mediante algoritmos.	Los estudiantes no son capaces de identificar errores en su código por lo tanto no puede resolver el problema planteado paso a paso.
Pensamiento Crítico y programación	El estudiante tiene pensamiento crítico siendo capaz de programar con claridad su algoritmo con lógica clara. Reconoce y corrige errores. Logra obtener la solución óptima.	El estudiante es tiene algún pensamiento crítico dado que es capaz de programar, aunque es poco claro en su lógica, pero logra reconocer y corregir sus errores y obtiene la solución esperada.	El estudiante no logra aplicar el pensamiento crítico para programar con claridad. Su enunciado es poco claro en su lógica. No logra reconocer y corregir sus errores y no obtiene la solución esperada.
Trabajo en Equipo	El estudiante es capaz de colaborar de manera efectiva con otros en problemas de programación, y de liderar y guiar el trabajo en equipo.	El estudiante tiene algunas dificultades para trabajar en equipo pero algunas tareas logra realizarlas con su par.	El estudiante no trabaja en equipo, no lidera ni guía a sus compañeros. Tampoco entiende que significa colaborar con otros.
Comunicación y Presentación	Los estudiantes son capaces del comunicar sus ideas y soluciones de programación de manera clara y efectiva, tanto oralmente como por escrito, y de manera creativa e innovadora. Utiliza algún recurso TIC en su presentación	Los estudiantes son capaces del comunicar sus ideas y soluciones de programación de forma poco clara no maneja el vocabulario técnico, aunque son creativos y utilizan las en su presentación	Los estudiantes no logran expresar con claridad sus ideas y soluciones de programación y utilizan muy poco las TIC

Tabla 5: Marcadores para medir la evaluación de la 3ra etapa
Fuente: elaboración propia

Los alumnos del grupo de control que no fueron expuestos al experimento trabajaron de manera habitual, es decir, durante las clases prácticas procedieron a resolver los ejercicios de las guías de trabajos prácticos con la orientación y guía del docente. Realizaron los algoritmos en papel y luego codificaban su solución en la computadora en el lenguaje C, compilaban y ejecutaban y algunos utilizaban sus celulares mediante la aplicación (coding C). Si bien no se les exigió el trabajo grupal, algunos lo hacían al estar en el aula de manera presencial y de esta manera lograron llegar a la solución de los ejercicios.

Los ejercicios en esta etapa para el grupo de control fueron:

Ejercicio 1: mediciones de teodolitos y promedio

Se ingresan por teclado las mediciones de 3 teodolitos en diferentes partes de la ciudad de Corrientes, y se necesita conocer el promedio de dichas mediciones. Realizar el algoritmo que muestre en pantalla los valores ingresados y el promedio obtenido de los teodolitos.

Ejercicio 2: Cálculo de porcentajes.

La FaCENA recibió una donación especial que será repartida entre las carreras de: Ingeniería Eléctrica, Ingeniería en Electrónica e Ingeniería en Agrimensura y Contabilidad: -Ingeniería Eléctrica: 30% - Ingeniería Electrónica: 15% -Ingeniería en Agrimensura: 20% -Contabilidad: lo que resta de la donación - Diseñar un algoritmo que determine cuánto recibirá cada carrera y mostrar en pantalla todo lo hallado.

Ejercicio 3: Cálculo de porcentajes

Dado como dato el sueldo de un trabajador, que se ingresa por teclado, aplicar un aumento del 15% si su sueldo es inferior a \$10.000. Mostrar en pantalla el sueldo a cobrar e indicar si le corresponde aumento.

Ejercicio 4: Escribir un programa que solicite el ingreso de una letra por teclado. Mostrar por pantalla el valor decimal que le corresponde según la tabla ASCII. Evaluando dicho número, mostrar un mensaje en pantalla cuando la letra ingresada sea minúscula.

Capítulo 5

PRESENTACIÓN DE RESULTADOS Y DISCUSIÓN

Resultados

Se presenta en el este capítulo la información obtenida luego de haberse aplicado la metodología de seguimiento descrita en el capítulo anterior a través de diferentes técnicas de recolección de información que forman parte del esquema metodológico definido anteriormente.

En el capítulo precedente se explicitó sobre los pasos que se realizaron en cada etapa a fin de medir el grado de avance en el desarrollo del pensamiento computacional y la programación de los estudiantes de Ingeniería Eléctrica.

Para poder responder a las preguntas de investigación primeramente se tomaron los datos que arrojó la encuesta realizada a 30 estudiantes. En la obtención de los datos se observó lo siguiente:

Pregunta 1: ¿Cuál es la opción correcta y más completa del término computadora? A la misma los estudiantes respondieron de la siguiente manera:

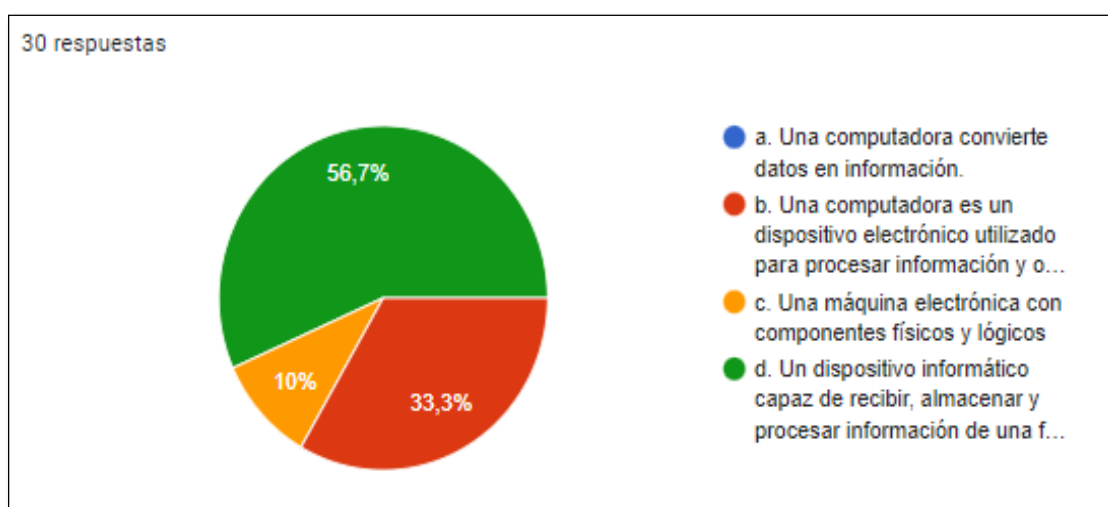


Gráfico 1: Respuesta de los alumnos de IE – a la Pregunta 1
Fuente: Elaboración propia

Según se pudo observar en el Gráfico 1, de un total de 30 (treinta) respuestas: el 56,7% respondió acertadamente, el 33,3% respondió a la segunda opción del término, aunque no es la más completa y solo el 10% no respondió acertadamente.

Pregunta 2: ¿Cuáles son las funciones de la computadora?

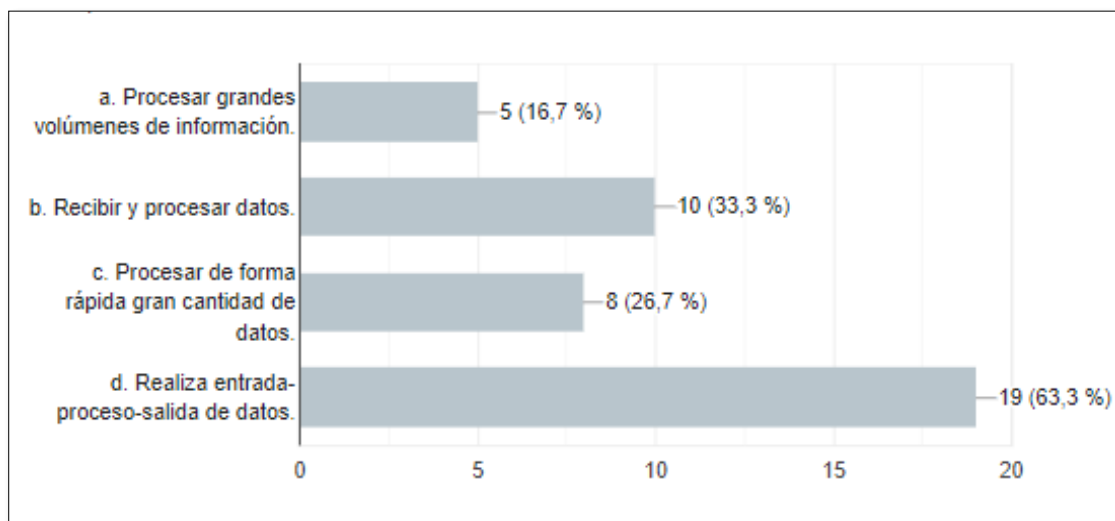


Gráfico 2: Respuesta de los alumnos de IE – a la Pregunta 2

Fuente: Elaboración propia

Podemos observar en el gráfico de barras que de las 30 (treinta) respuestas: el 63.3 % de los alumnos encuestados respondió correctamente a esta pregunta y solo el 16,7% respondió con la respuesta menos acertada.

Otra de las preguntas que se destacó como importante para saber si los alumnos traen conocimientos previos relativos a ¿qué es un problema? y conocer acerca de resolver problemas por medio de pasos ordenados (algoritmos). Vemos las respuestas y preguntas correspondientes:

Pregunta 3: ¿Es lo mismo dato e información? ¿por qué? Las respuestas a esta pregunta fueron:

No, porque los Datos son símbolos y la información conjunto de datos organizados

No por qué los datos es la representación de un número, letra sin importancia y la información es un conjunto de datos procesados para transmitir un mensaje

No porque los datos procesados

30 respuestas
No, no son lo mismo. Un dato es un algoritmo aislado, en cambio la información es un conjunto de datos contextualizados
No es lo mismo porque en la información entran todo lo que sea datos.
Información es un conjunto de datos Datos son las palabras, números, letras
No, no es lo mismo. El dato es un número, una letra, y la información es una cantidad
Si es lo mismo porque la informacion esta conformado por un conjunto de datos
No es lo mismo por q la informacio son datos procesados adecuadamente
No,un dato es un fragmento de información y no tiene valor por si solo.Informacion es un conjunto de datos relacionados entre si.
Datos es es un recurso donde almacenamiento de, información y dónde adquirimos lo que buscamos de menera aleatorio también podríamos decir ambos son convenientes a la ahora de guardarlo y informarse...
No es lo mismo. Dato son cosas dispersas ej color, esas etc y la información es un conjunto de datos.
No por qué no es lo mismo datos e INFORMACIÓN por qué la información se da atra vez de los datos tiene que a ver un dato para que haiga una información
Datos es un término Es la entrada sin procesar de la cual se produce la información que se refiere a los datos que han sido procesados y comunicados de tal manera que pueden ser entendidos e interpretados por el receptor.
No, ya que los datos son inútiles por si solos (ejemplo un número, color, etc) en cambio la información son la suma de esos datos procesados.
No. Un Dato se podría decir que es un hecho aislado que por si solo carece de "sentido" significado... mientras que la informacion es un conjunto de datos, que poseen un sentido, es decir, un significado real y preciso. Y que tiene como utilidad la toma de decisiones!

*Figura 21: recorte de respuestas de los alumnos de IE – a la Pregunta 3
Fuente: Elaboración propia*

En relación a esta pregunta, la mayoría pudo describir acertadamente la diferencia entre datos e información. Destacando la diferencia sobre todo en el aspecto de que la información es un conjunto de datos procesados. Los datos son representaciones de objetos del mundo real (ejemplo: color, número de teléfono, un DNI, entre otros).

Pregunta 4: ¿qué es un problema? Pudimos obtener las siguientes respuestas más afines a la mayoría, tabuladas en la siguiente tabla:

Un problema es un hecho que irrumpe en una acción. Por ejemplo, quedarse sin gas mientras se cocina	Problema es una situación que necesita ser resuelta. Ej: Se cortó el internet, tengo que buscar una solución que podría ser llamar a la compañía	Es una situación que hay que pensar en cómo resolver.	Una situación que requiere una resolución. No tengo batería y debí de avisar cuando estoy llegando a casa para que me abran la puerta
Una situación de conflicto ej una calle cortada que nos lleva la lugar donde queremos ir y ver por dónde podemos llegar.	Un obstáculo en el desarrollo del sistema	Es una situación que requiere, de cierto conocimiento para poder resolverlo. Que de otra forma no se podría!	Problema es el conflicto que se nos presenta para tomar alternativas y elegir en las opciones que tenemos la más relevante para brindar una solución.

Tabla 6: Respuestas a la pregunta 4 - ¿Qué es un problema?

Fuente: Elaboración propia

En las respuestas recabadas en este punto de la encuesta, se observó que los estudiantes respondieron adecuadamente a la idea de problema según Kerlinger y Lee (2002), *un problema es un enunciado u oración interrogativa que pregunta: ¿Qué relación existe entre dos o más variables? La respuesta constituye aquello que se busca en la investigación. Es una contradicción entre lo que sucede, lo que es y lo que debería ser.*

La situación no resuelta o indeterminada podría llamarse situación “problemática”; se hace problemática en el momento mismo de ser sometida a investigación. El resultado primero de la intervención de la investigación es que se estima que la situación es problemática (Dewey).

Para la Real Academia Española en el Diccionario de la lengua, en su vigésima primera Edición (R.A.E., 1992, p. 1184):

Problema: (Del Lat. Problema) Cuestión que se trata de aclarar. // **2.** Proposición o dificultad de solución dudosa. // **3.** Conjunto de hechos o circunstancias que dificultan la consecución de algún fin. // **4.** Disgusto. Preocupación. // **5.** *Mat.* Proposición dirigida a averiguar el modo de obtener un resultado cuando ciertos datos son conocidos. **Determinado.** *Mat.* Aquel que no puede tener sino una solución, más de una en un número fijo. // *Indeterminado.* *Mat.* Aquél que puede tener indefinido número de soluciones.

Por lo tanto, la idea de problema fue bastante bien definida y declarada por los estudiantes encuestados. Esto siembra una base importante para el desarrollo de nuestra metodología dado que partir de la base del conocimiento de un problema, poder enunciarlo

y luego resolverlo es fundamental en la asignatura, para poder avanzar luego con la representación de su solución. Los estudiantes lograron tener claridad acerca de la situación o problema y de llevar adelante diferentes alternativas de solución.

Pregunta 5: ¿Qué haría usted para resolverlo?

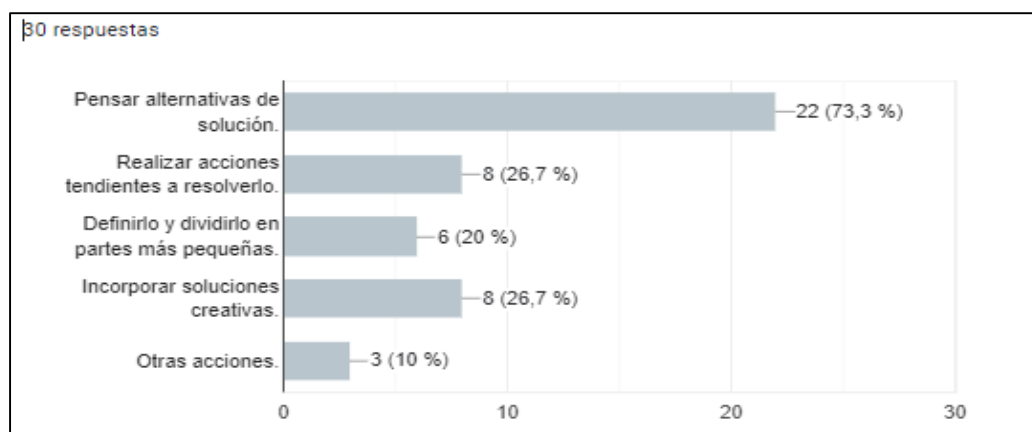


Gráfico 3: respuesta de los alumnos de IE – a la pregunta 5
Fuente: Elaboración Propia

La mayoría respondió como refleja el gráfico de los 30 (treinta) encuestados: el 73.3% contestó que se deben pensar alternativas de solución, el 26,7% dijo que se debe realizar acciones tendientes a resolverlo e incorporar soluciones creativas, un 20% aplicó que es conveniente dividir al problema en partes más pequeñas, Y un 10% manifestó otras acciones.

Pregunta 6: ¿Qué entiende por algoritmo? Las respuestas fueron:

Conjunto ordenado de operaciones sistemáticas que permite hacer un cálculo y hallar la solución de un tipo de problemas.
Conjunto de acciones ordenadas para resolver una incógnita o problema.
Es una forma de resolver problemas, es decir, un medio que mediante pasos lógicos y ordenados, finitos, precisos tienen a resolver un problema en cuestión.
Es una serie de pasos que nos sirve de solución para resolver un problema
Es un conjunto ordenado de operaciones sistemáticas que permiten hacer cálculo y hallar soluciones a los problemas.
Conjunto ordenado de operaciones que permiten hallar una solución a algún problema.
Procesos o pasos a seguir para conseguir algo o un resultado específico
Se entiende como un conjunto de pasos a realizar de manera eficaz y eficiente de acuerdo al problema en

Un algoritmo es un dato aislado que sirve para programar
Problemas y soluciones
Es un conjunto de pasos o procedimientos que tiene un fin.
Es un conjunto de pasos para resolver un problema
Son pasos ordenados para resolver un problema
Operaciones ordenados
Una serie o conjunto de operaciones organizadas de forma lógica y/o ordenada con la finalidad de resolver un problema o llegar a un objetivo en concreto
Un sistemas de organización y métodos de solución entre otras cosas ...
Una forma de resolver problemas. secuencias-instrucciones. pasos.

*Figura 22: respuesta de los alumnos de IE – a la pregunta 6 ¿Qué entiende por algoritmo?
Fuente: Elaboración Propia*

Las respuestas a la pregunta 6, fueron en su mayoría muy adecuadas y da la pauta que los alumnos ingresantes de ingeniería eléctrica ya tenían algún conocimiento previo de concepto de algoritmos o por lo menos la interpretación que realizan del término es correcta.

Pregunta 7: ¿Qué significa la programación de computadoras?

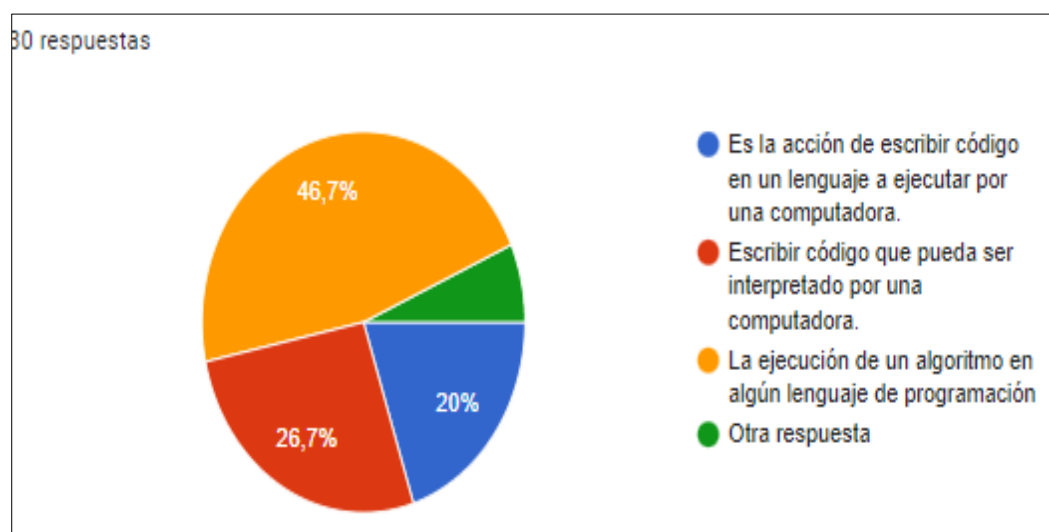


Gráfico 4: Respuestas de los alumnos de IE a la pregunta 7
Fuente: Elaboración propia

El 46,7 % de los encuestados respondió acertadamente a la pregunta que significa programar con una computadora, lo que nos favoreció al momento de enseñarles un lenguaje de programación para que luego lo pongan en práctica durante el cursado.

Pregunta 8: Si se le presenta el caso de encender un automóvil ¿Usted qué haría?

Describe los pasos:

Poner las llaves poner en marcha, hacer el primer cambio, ver mi dirección y pisar el acelerador
1 tomar las llaves 2 dirigirme al automóvil 3 quita la alarma / seguro con el receptor 4 abrir la puerta
Tomr las llaves colocarlas aclarar y arrancar de esa manera luego a proceder de realizar los cambios necesarios 1.Tomar la llaves de automovil en particular. 2.Dirigirse a al automovil. 3.Abrir la puerta. 4.Entrar al vehiculo. 5.Cerrar la puerta. 6.Visualizar al costado del volante la entrada de la llave. 7.Agarrar la llave el auto y conectarlo a la entrada que se expreso en el anterior paso 6. 7.Encender el auto.
En caso de que se presente la situación de encender un automóvil, haría lo siguiente: 1) Abrir la puerta del conductor y sentarse en el asiento. 2) Verificar que el auto esté en punto neutro (0) 3) Ubicar la llave al costado derecho del volante (bombín de arranque) 4) Girar la llave levemente hacia la derecha hasta que el motor encienda.

Las respuestas que se observaron a esta pregunta, indicó que los alumnos comprendieron el problema “encender el automóvil”. Por consiguiente, podemos deducir que el realizar una tarea paso a paso es un concepto que lo fueron incorporando en su paso por el sistema educativo anterior. Si bien algunos alumnos no lo expresaron como una instrucción seguida de otra, lograron escribir, en general, cómo se haría para llegar al objetivo (encender un automóvil).

Resultados de la aplicación de las rúbricas en cada etapa

En la segunda etapa de la metodología aplicada, se seleccionaron ejercicios conteniendo estructuras de control para la evaluación de los alcances del pensamiento computacional y la programación. De la rúbrica de elaboración propia que se aplicó al grupo piloto de los 30 alumnos, ~~de la misma~~ surgieron los siguientes resultados, expresados en porcentajes y teniendo en cuenta los marcadores aplicados con sus correspondientes alcances.

Marcadores	Muy bueno	Bueno	Regular	Malo
1. Pensamiento Algorítmico	50%	34%	10%	6%
2. Pensamiento crítico	40%	34%	15%	11%
3. Elaboración del Algoritmo	50%	33%	9%	8%
4. Programación, ejecución y prueba	43%	29%	18%	10%
5. Creatividad e innovación	46%	26%	17%	11%

Tabla 7: Porcentajes de evaluación a alumnos de IE según rúbrica de la etapa 2 de la metodología
Fuente: Elaboración propia

Podemos deducir del primer marcador (**pensamiento algorítmico**), que el 50% de los alumnos involucrados en la investigación logró realizar cada etapa solicitada paso a paso y argumentó claramente la solución encontrada; el 34% logró realizar todas las etapas, pero no argumentó con claridad dicha solución, el 10% logró realizar solo algunas etapas, pero

argumentó con claridad la solución encontrada y el 6% restante no logró realizar todas las etapas y su argumentación respecto a la solución encontrada fue poco clara.

Del segundo marcador (**pensamiento crítico**) podemos deducir que el 40% de los alumnos logró realizar deducciones lógicas y optimizar la solución correctamente; el 34% realizó algunas deducciones lógicas y optimizó la solución correctamente; el 15% de ellos realizó solo algunas deducciones lógicas y la solución es medianamente óptima mientras que el 11% de los estudiantes no logró realizar deducciones lógicas y no encontró una solución óptima al problema planteado.

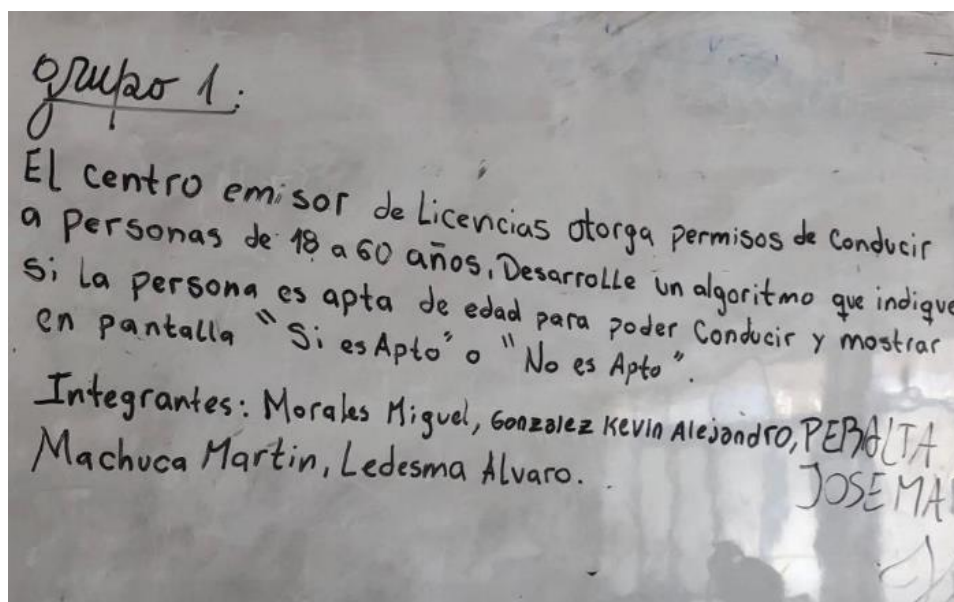
Con relación al marcador 3 (**elaboración del Algoritmo**), se observó que el 50% de los estudiantes elaboró el algoritmo correctamente (el mismo tiene inicio y final, los pasos son correctos no son ambiguos); el 33% de ellos elaboró el algoritmo correctamente (el mismo tiene inicio y final, pero le faltan algunos pasos, aunque no son ambiguos). El 9% elaboró el algoritmo con algunos errores (el mismo tiene inicio y final, pero le faltan algunos pasos, aunque no son ambiguos); mientras que solo el 8% de los alumnos no elaboró el algoritmo correctamente (el mismo tiene inicio y final pero los pasos no son correctos, no llega a la solución esperada).

Del marcador número 4 (**programación, ejecución y prueba**), se comprobó que el 43% de los alumnos expuestos al experimento logró programar correctamente las instrucciones que refleja el algoritmo; pudo compilar y ejecutar bien y sabe reconocer errores de programación y lógica. El 29% logró programar la mayoría de las instrucciones que reflejan el algoritmo, también compiló y ejecutó, pero el algoritmo arrojó algún error de compilación que luego el alumno logró corregir y llegó finalmente a la solución esperada. El 18% de los alumnos, programó las instrucciones, pero las mismas poseen algunos errores de compilación y de lógica. No logró reconocer adecuadamente, aunque alcanzó corregir algunos. Y solamente un 10% de ellos, no logró programar correctamente las instrucciones, no corrigió adecuadamente los errores de programación y lógica por lo tanto no pudo ejecutar el programa.

El último marcador de esta etapa (**creatividad e innovación**) reflejó lo siguiente: el 46% de los alumnos del experimento logró ser creativo en todo momento en la solución que realizó, la misma reflejó claridad e innovación porque el algoritmo fue preciso y eficiente. En el 26% de los alumnos se vislumbró claridad en la solución, aunque fue poco innovadora, fue adecuada al problema planteado. El 17% de los alumnos logró una solución algo creativa, aunque poco clara, medianamente eficiente y algo innovadora. Mientras que el 11% restante no reflejó creatividad al momento de armar una solución al problema, no consiguió innovar en la solución y la misma no fue clara y ni eficiente.

En la etapa 3 del experimento se evaluó el trabajo colaborativo en grupo de dos o más estudiantes que trabajaron en la elaboración de ejercicios donde fue factible aplicar las estructuras de control vistas. Los cuales una vez realizados y programada su solución, debían poder explicar al resto de sus compañeros, cómo lograron llegar al resultado final.

Algunos de los enunciados que elaboraron en los diferentes grupos de alumnos y que fueron compartidos durante el cursado fueron los siguientes:



*Figura 13: Enunciado realizado por alumnos del grupo 1 - IE
Fuente: elaboración grupal- alumnos de Ingeniería eléctrica*

Grupo 2:
ESCRIBA UN ALGORITMO QUE CALCULE
EL TOTAL QUE DEBE PAGAR UNA PERSONA
QUE VA A COMPRAR TELA (VEGETAL). TENIENDO
EN CUENTA QUE EL METRO CUESTA \$45,80 Y
QUE ADEMÁS SI SE SUPERA LOS 5 METROS
SE LE REALIZARÁ UN DESCUENTO DEL 10%
INFORMAR POR PANTALLA EL VALOR FINAL
CON Y SIN DESCUENTO A ABONAR.
BLANCO, ROMINA; ZARZA ANALIA Y CEJAS, ALEXIS.

Figura 14: Enunciado realizado por alumnos del grupo 2 - IE
Fuente: elaboración grupal- alumnos de Ingeniería eléctrica

Grupo 3:
El supermercado en la sección
de verdulería da la opción a sus
clientes de saber el precio de
su producto antes de llegar a casa
para evitar demora y/o inconvenientes
Integrantes: Avila Juan, Galarza Ivan
Ramirez Mosquera Jose, Ramirez Andres

Figura 15: Enunciado realizado por alumnos del grupo 3 - IE
Fuente: elaboración grupal- alumnos de Ingeniería eléctrica

Grupo 4:
Realice un programa que le permita al operador de un
semáforo poder cambiar su color utilizando una variable que
puede tomar los valores 1, 2 o 3.
Debe mostrar por pantalla: 1- Rojo
2- Amarillo
3- Verde
Integrantes:
Carrio, Ariadna
Velazquez, Johana
Delfino, Juliana.

Figura 16: Enunciado realizado por alumnos del grupo 4 - IE
Fuente: elaboración grupal- alumnos de Ingeniería eléctrica

La rúbrica aplicada al grupo piloto de 30 alumnos de la Ingeniería Eléctrica en la etapa 3 del experimento, donde se seleccionaron marcadores para esa etapa, arrojó los resultados que se presentan en la tabla siguiente:

Marcadores	Alto	Medio	Bajo
1. Enunciar problemas cotidianos que se resuelven mediante algoritmos	56%	25%	19%
2. Planteamiento algorítmico	54%	26%	20%
3. Pensamiento crítico y programación	49%	34%	17%
4. Trabajo en equipo	65%	28%	7%
5. Comunicación y presentación	64%	31%	5%

Tabla 8: Porcentajes de la evaluación a alumnos de IE según rúbrica de la etapa 3 de la metodología
Fuente: Elaboración propia

Del primer marcador **“enunciar problemas cotidianos que se resuelven mediante algoritmos”** se desprende que el 56% de los alumnos del total de la muestra, poseía un alto nivel para elaborar adecuadamente problemas con claridad y se encuadró dentro de una estructura de control vista. Había un 25% que se encontraba en un nivel medio es decir, elabora el problema, pero el mismo fue poco claro, aunque se encuadra en alguna de las estructuras de control vista y el 19% restante poseía un nivel bajo es decir no puedo elaborar claramente un problema de la vida cotidiana que contenga una estructura de control aprendidas.

Del segundo marcador **planteamiento mediante algoritmos**, se pudo establecer que el 54% de los alumnos del experimento fueron capaces de seguir instrucciones básicas paso a paso mediante un algoritmo (nivel alto). El 26% de ellos mostró dificultades para plantear un problema paso a paso o más complejos mediante algoritmos (nivel medio). Y el 20% de los alumnos no fueron capaces de identificar errores en su código por lo tanto no pudieron resolver el problema planteado paso a paso (nivel bajo).

Respecto al marcador que nos indicaba **el pensamiento crítico y la programación**, el 49% de los alumnos sometidos al experimento demostraron tener pensamiento crítico siendo capaces de programar con claridad su algoritmo con lógica clara y reconocer y corregir sus errores y por lo tanto obtener la solución óptima (nivel alto). Mientras que el 34% de los alumnos tenía algún pensamiento crítico dado que fueron capaces de programar, aunque fueron poco claros en su lógica, pero lograron reconocer y corregir sus errores y obtener la solución esperada. El 17% no logró aplicar el pensamiento crítico para programar con claridad. Su enunciado fue poco claro en su lógica. No pudo reconocer y corregir sus errores y no llegó a la solución esperada.

En cuanto al marcador **trabajo en equipo**. Se pudo rescatar que en un nivel alto del mismo estarían un 65% de los alumnos del experimento, ellos fueron capaces de colaborar de manera efectiva con otros en problemas de programación, y de liderar y guiar el trabajo en equipo. El 28% de ellos poseía un nivel medio dado que tuvieron algunas dificultades para trabajar en equipo, pero algunas tareas lograron realizarlas con sus pares. Solo el 7% tenía un nivel bajo es decir no trabajó en equipo, no lideró ni orientó a sus compañeros. Tampoco entendió que significó colaborar con otros.

El último marcador **comunicación y presentación**, arrojó que en un nivel alto se encentraron el 64% de los alumnos a los que se les aplicó el experimento, dado que fueron capaces de comunicar sus ideas y soluciones de programación de manera clara y efectiva, tanto oralmente como por escrito, y de manera creativa e innovadora utilizando el recurso TIC (proyector, computadora, software adecuado) en su presentación.



Figura 17: Alumnos de Ingeniería Eléctrica durante el trabajo en el aula

El 31% de ellos estuvo en un nivel medio dado que fueron capaces de comunicar sus ideas y soluciones de programación de forma poco clara y no manejaban el vocabulario técnico, aunque fueron creativos y utilizaron las TIC favorablemente. Solo el 5% de los alumnos del experimento no lograron expresar con claridad sus ideas y soluciones de programación, el uso de las TIC no fue muy adecuado en su presentación.

CAPÍTULO 6

CONCLUSIONES

En esta instancia, se presentan las conclusiones finales como producto de la reflexión a la que se arribó como resultado de aplicar la metodología propuesta a los alumnos de la Ingeniería Eléctrica de la FaCENA.

Se presenta los siguiente: por un lado, las respuestas que tienen que ver con los objetivos y por otro lado las respuestas a las preguntas de investigación.

Resultados en relación a los objetivos

De los objetivos específicos que se plantearon al comienzo de la investigación se obtuvieron los siguientes resultados:

Del objetivo 1) **Identificar las habilidades y competencias en pensamiento computacional y programación que se espera que los estudiantes adquieran durante su formación**, podemos destacar que se logró identificar las siguientes:

- Capacidad de resolver problemas de la vida cotidiana
- Aplicar algoritmos paso a paso
- Capacidad para aplicar estructuras de control
- Destreza y creatividad para elaborar situaciones problemáticas.
- Trabajo en equipo
- Automatización de los resultados
- Habilidades de comunicación.

En cuanto al objetivo 2) **diseñar una metodología de seguimiento que permita evaluar el grado de avance de los estudiantes en el desarrollo del pensamiento computacional y la programación**. Afirmamos que se logró diseñar una metodología de seguimiento acorde a los estudiantes de la carrera de Ingeniería Eléctrica, que fue aplicada al grupo piloto conformado por 30 (treinta) alumnos de la misma, la cual permitió evaluar el grado de avance en el desarrollo del pensamiento computacional y la programación.

Del objetivo 3) **Implementar la metodología de seguimiento en un grupo piloto de estudiantes de Ingeniería Eléctrica y recopilar datos relevantes.**

Durante este trabajo se logró aplicar la metodología de seguimiento en el grupo piloto de estudiantes de Ingeniería Eléctrica durante el cursado de la asignatura informática en el segundo cuatrimestre del 2022, se recopilaron datos relevantes referentes a cómo aprenden a resolver problemas informáticos y aplican la programación

Los alumnos aprenden mejor si ellos se involucran como partícipes y creadores de contenido.

- a. Los datos de las rúbricas aplicadas revelan que los alumnos desarrollan más creatividad si tienen que pensar ellos situaciones problemáticas de la vida real.
- b. También otro dato importante fue el trabajo colaborativo en grupos de tres o más alumnos, el cual favoreció positivamente al pensamiento computacional y resolución de problemas dado que se sumó al pensamiento individual el pensamiento del “par”, el cual aportó su idea, conocimiento, creatividad y estrategia. Este tipo de trabajo contribuyó al respeto por las opiniones de los pares, el dividir tareas para avanzar en la resolución de problemas, establecer un líder que los guíe en las tareas a realizar de manera tal de ordenar el trabajo.
- c. Se vislumbró mejoras en la comunicación entre pares y con el docente, en el aula y fuera de ella a través de medios virtuales (foros, mensajería interna, e-mail).

Del objetivo 4) **Analizar los datos obtenidos a través de la metodología de evaluación.**

Analizando los datos obtenidos a través de la metodología de seguimiento que se aplicó se pudo determinar el grado de avance en el pensamiento computacional de manera progresiva, ~~Dado que a medida que transcurrían las clases y se avanzaba en los conceptos de la programación y el pensamiento computacional se vio un avance de los alumnos que se reflejan en las diferentes etapas mediante la evaluación de diferentes aspectos como ser~~

el nivel de complejidad de los problemas que se les plantean, en principio fueron desafíos simples y a medida que transcurrían las clases los desafíos fueron más complejos que requerían el uso de estructuras de control más avanzadas.

Evaluar la eficiencia de los algoritmos propuestos por los estudiantes fue fundamental para medir su progreso en el pensamiento computacional. A medida que avanzaron en el aprendizaje y la ejercitación, los estudiantes lograron ser capaces de optimizar sus soluciones, reduciendo el tiempo de codificación, ejecución y prueba.

Con el transcurrir de las clases pudieron aplicar conceptos aprendidos a situaciones nuevas desarrollando su creatividad. Se observó cómo se comunican, comparten ideas y conocimientos, y trabajan juntos para lograr objetivos comunes, poniendo en práctica el trabajo colaborativo. El avance en el aprendizaje de la programación también pudo observarse al estructurar la información, utilizando programas de visualización adecuadas y explicando sus procesos de pensamiento de manera coherente.

La primera etapa de diagnóstico arrojó que los conocimientos previos de los estudiantes influyen positivamente en la adquisición del pensamiento computacional dado que se parte de la base de dichos conocimientos al inicio (conceptos de algoritmos, resolver problemas paso a paso, identificar los datos) y se trabaja en afianzar o fortalecer dicho pensamiento, mediante la búsqueda de soluciones a diferentes problemas que se les plantean y en la forma de resolverlos metodológicamente.

La mayoría de los grupos manifestó que *“les resultó una tarea creativa, divertida y que les permitió trabajar colaborativamente dado que aprovecharon fehacientemente los conocimientos que los pares aportaban”* para resolver el ejercicio que cada grupo planteo. El hecho de poder crear un problema los puso en un rol activo dado que no se trataba de resolver un problema lúdico y orientado a una cierta estructura sino que era necesario pensar en las palabras adecuadas para dar claridad al enunciado (*tarea fundamental al momento de escribir o enunciar un problema*) y en buscar las formas de escribir correctamente un

enunciado de un problema computacional, es decir que el mismo pudiera ser resuelto y luego programado en la computadora y que el mismo arroje resultados válidos.

En relación a las preguntas de Investigación:

1. *¿Qué recursos de la plataforma Moodle son pertinentes para medir el grado de avance en el pensamiento computacional de los alumnos de Ingeniería eléctrica?*

Los recursos de la plataforma Moodle que resultaron más adecuados para para medir el grado de avance en el pensamiento computacional fueron:

Las **tareas**, permitió evaluar los ejercicios resueltos en algoritmos, observando la eficiencia de las soluciones y la capacidad de los estudiantes para aplicar conceptos de programación en el contexto de la vida cotidiana y la ingeniería eléctrica. Evaluar sus habilidades, y proporcionar retroalimentación personalizada.

Cuestionarios, se utilizaron para evaluar el conocimiento teórico de los estudiantes en temas relacionados con el pensamiento computacional, algoritmos, estructuras de datos y programación. También se pudieron incluir preguntas prácticas que requerían la resolución de problemas computacionales.

Base de datos de preguntas: La creación de una base de datos de preguntas nos permitió crear una gran cantidad de preguntas relacionadas con los contenidos desarrollados en el programa de la asignatura ordenados en categorías según los temas dados. Las preguntas se utilizaron tanto en cuestionarios teóricos como prácticos.

Foros de discusión: que se utilizaron entre los estudiantes para discutir soluciones a problemas computacionales. Logró participación de los alumnos y el desarrollo de su capacidad para argumentar y justificar sus soluciones, y la habilidad para trabajar en equipo.

2. *¿Cuál es el nivel de conocimiento y competencias en pensamiento computacional y programación que poseen los estudiantes de Ingeniería Eléctrica al ingresar a la universidad?*

Los estudiantes de Ingeniería eléctrica poseen algunas competencias en pensamiento computacional y programación, dado que la encuesta de diagnóstico arrojó que pueden enunciar problemas cotidianos con claridad, la mayoría logra crear el algoritmo (pensamiento

algorítmico), distingue datos e información, reconoce la diferencia entre dato e información, tiene conocimientos en lenguaje de programación, sabe lo que es una computadora y cómo funciona. Si bien en la encuesta no se incluyó la pregunta para saber en qué escuela secundaria finalizó sus estudios, se hicieron consultas verbales durante las clases y muchos de ellos vienen de escuelas técnicas de la provincia que ya desarrollaron entre sus contenidos esos temas.

3. *¿Cuáles son las principales dificultades o desafíos que enfrentan los estudiantes de Ingeniería Eléctrica en el desarrollo del pensamiento computacional y la programación?*

Las principales dificultades encontradas refieren a la escasa metodología o método de resolver problemas informáticos que los alumnos traen. Si bien las encuestas arrojaron que los mismos tienen cierto conocimiento sobre algoritmos o noción de problema. Les dificulta el pensar en pasos ordenados para resolver los mismos. Se complejiza un poco más cuando tienen que aprender las estructuras de control dado que las mismas son abstractas y al momento de aplicarlas a situaciones problemáticas para resolverlos en computadora con el lenguaje de programación dado, los alumnos sienten que no pueden llevar a cabo esos casos de manera correcta, porque no internalizan rápidamente la estructura y se les dificulta un poco más la aplicación de la misma en un lenguaje de programación que conlleva conocer su sintaxis. También se les dificulta el concepto de abstracción es decir enfocarse en los datos importantes y relevantes del problema y dejar de lado los demás datos que no aportan a la solución del mismo.

4. *¿Qué métodos o enfoques pedagógicos se utilizan actualmente en la enseñanza del pensamiento computacional y la programación en Ingeniería Eléctrica y cuál es su efectividad?*

Los métodos o enfoques didácticos que caracterizan la enseñanza-aprendizaje de la programación y adquisición del pensamiento computacional tiene que ver con un enfoque didáctico caracterizado por la resolución de problemas y desarrollo de algoritmos y programación. En todos los casos se realiza la introducción de conceptos a través de la

enseñanza de la programación usando un determinado lenguaje y el uso de dispositivos (computadora, notebook, notebook, celular), avanzando con el desarrollo del pensamiento algorítmico a través de la programación. Otras técnicas y estrategias que aplicamos tienen que ver con organizar los datos de entrada, el proceso y la salida de información; dividir problemas en partes para obtener la solución adecuada al problema que se plantea. Algunos ejemplos de ejercicios son instrucciones simples para resolver alguna ecuación, aplicar la fórmula del triángulo para hallar la superficie del mismo. Tomar mediciones y calcular promedios entre otros. Este método es efectivo dado que durante el cursado los estudiantes avanzan en el conocimiento y aplicación de estructuras de control en las diferentes situaciones problemáticas.

5. *¿Qué recomendaciones pueden formularse para mejorar la enseñanza y el desarrollo del pensamiento computacional y la programación en el plan de estudios de Ingeniería Eléctrica?*

Luego de haber transcurrido y cumplimentado los pasos de esta investigación se pueden formular algunas mejoras a fin de que la enseñanza y el desarrollo del pensamiento computacional y la programación para los alumnos de Ingeniería Eléctrica sea altamente satisfactorio. En primer lugar, la sugerencia es que la asignatura tenga una continuidad en algún otro año de la carrera, sugerimos en el último año de la misma dado que los alumnos deben realizar su trabajo final y precisan en muchos casos, aplicar la programación y poner en práctica el pensamiento computacional. Dado el ritmo vertiginoso con que avanzan las tecnologías si bien las bases en estructuras de control la tienen de nuestra materia, los lenguajes cambian y también los paradigmas de programación, no solo existe la programación estructurada, también pueden aprender la programación orientada a objetos, programación funcional, programación Web u otras, estas tecnologías ayudarían en el desarrollo de la tesina.

Mejoras a futuro

Entre las mejoras que se desean establecer para la asignatura informática está lo que se planteó al comienzo de este trabajo que es poder contribuir al cuerpo de conocimiento de la asignatura, la metodología desarrollada podrá aplicarse no solo a un grupo de alumnos sino también a los demás estudiantes de las carreras de Ingeniería en Electrónica y en Agrimensura. También puede aplicarse en otros contextos como ser los alumnos que cursan la carrera de Licenciatura en Sistemas que en los primeros años desarrollan un programa similar al nuestro, pero con más profundidad en los conceptos fundamentales de la programación. La misma aporta conocimientos y herramientas prácticas para la evaluación (rúbricas) y permite mejorar las habilidades computacionales de los estudiantes tanto de la Universidad como también puede aplicarse a los Institutos Superiores donde se den materias de programación.

Propongo el uso de más recursos dentro del aula virtual similares a Padlet, dado que su aplicación como herramienta de apoyo para la presentación de trabajos dentro del aula virtual fue altamente satisfactorio; lo que les permitió publicar trabajos y ver el de los demás para evacuar dudas, realizar comentarios y mejorar sus producciones. También incluir espacios activos de Debate dentro del aula virtual (foros de opinión, debates, consultas), buscando involucrar más y mejor a los alumnos en los diferentes temas que desarrollamos durante la cursada.

Propongo aplicar más trabajo grupal y colaborativo durante más etapas del desarrollo de las guías de trabajos prácticos. El trabajo colaborativo y en grupos es una herramienta fundamental para el desarrollo de la competencia que tiene que ver con fomentar en los estudiantes la *autorregulación del aprendizaje*, permitiéndoles reflexionar sobre sus fortalezas y debilidades o deficiencias, así como facilitarles la fijación de metas y áreas en las que deben apoyarse en los compañeros para lograr éxito en los desafíos.

Respecto a los alumnos de la Ingeniería Eléctrica que fueron parte del experimento y los alumnos de la Ingeniería en Agrimensura que no estuvieron expuestos al mismo. Se

deduce que al no aplicarse una encuesta de diagnóstico al comienzo no pudimos saber de antemano qué conocimientos traían de su paso por el sistema educativo anterior para seleccionar más adecuadamente los contenidos y ejercicios a desarrollar en las clases. Cuando se los sometió a ejercicios de las guías de trabajos prácticos que no fueron seleccionados exclusivamente para ser evaluados con los marcados de la tabla-4 y tabla-5, no pudimos recabar datos de cómo aprenden e incorporan el pensamiento computacional a medida que fuimos avanzando en el desarrollo de las clases, solamente pudimos observarlos trabajar durante el desarrollo de las clases y luego esos trabajos fueron subidos a la opción tarea de Moodle para su posterior devolución y calificación.

A estos estudiantes de Ingeniería en Agrimensura se les evaluó mediante la opción Tarea en la plataforma Moodle y cuestionarios teóricos y prácticos y se vio que los que poseían mejor comprensión en los temas dados aprobaron los mismo y lograron avanzar en el cursado de la asignatura, pero muchos de ellos no pudieron avanzar teniendo que darles más posibilidades de recuperación de ejercicios que al resto.

Al no trabajar colaborativamente con este grupo de alumnos, no pudimos recabar como se desenvolvían para el trabajo en equipo o de a pares ni cómo desarrollaban su creatividad para lograr satisfactoriamente alcanzar algún objetivo de aprendizaje de los que se plantearon.

Algunas **limitaciones** que podríamos considerar en la metodología de seguimiento para el grupo de estudiantes de ingeniería eléctrica serían:

Conocimientos previos: Los estudiantes pueden tener diferentes niveles de conocimientos previos en programación, lo que puede dificultar la homogeneidad en el aprendizaje.

Disponibilidad de recursos: Limitaciones en el acceso a computadoras, software específico o conexión a internet pueden afectar la continuidad y calidad del seguimiento.

Tiempo: Los estudiantes de ingeniería eléctrica suelen tener una carga académica pesada, lo que puede limitar el tiempo disponible para dedicar al aprendizaje de programación.

Diversidad de estilos de aprendizaje: Los estudiantes pueden tener diferentes estilos de aprendizaje, por lo que la metodología de seguimiento debe ser flexible para adaptarse a estas diferencias.

Falta de retroalimentación: La falta de retroalimentación o seguimiento individualizado puede dificultar la identificación de dificultades específicas en el aprendizaje de la programación.

Falta de motivación: Algunos estudiantes pueden no estar motivados para aprender programación, lo que puede afectar su compromiso y progreso en el seguimiento.

Limitaciones tecnológicas: Problemas técnicos, como fallas en el hardware o software, pueden interrumpir el proceso de seguimiento y aprendizaje de programación.

Bibliografía:

- Aparici, R. (2011). Principios pedagógicos y comunicacionales de la educación 2.0. *La educación*, 145. Recuperado desde: http://www.educoas.org/portal/La_Educacion_Digital/laeducacion_145/articles/Roberto_Aparici.pdf (consultado 10/10/19)
- Arranz, H. y Pérez, A (2017). Evaluación del pensamiento computacional en RIITE. *Revista Interuniversitaria de Investigación en Tecnología Educativa*, 3, 25-39. Doi: <http://dx.doi.org/10.6018/riite/2017/267411> (consultado 10/10/19)
- Barberá, E. (2006, julio). *Aportaciones de la tecnología a la e-Evaluación*. RED. Revista de Educación a Distancia, número M6. Consultado (24/11/2017), en <http://www.um.es/ead/red/M6> (consultado 10/10/19).
- Brennan, K., & Resnick, M. (2012). Using artifact-based interviews to study the development of computational thinking in interactive media design. Paper presented at annual American Educational Research Association meeting, Vancouver, BC, Canada
- Camilloni, A. R. W. (1998). La calidad de los programas de evaluación y de los instrumentos que lo integran En: *La evaluación de los aprendizajes en el debate didáctico contemporáneo*. Camilloni, A.R.W.; S. Celman; E. Litwin & M. del C. Palou de Maté. Editorial Paidós, Bs. As., Barcelona, México. pp: 67-92.
- Celman, S. (2001). “¿Es posible mejorar la evaluación y transformarla en herramienta de conocimiento?”, en: Camilloni, A., Celman, S., Litwin, E., Palou de Maté, M. *La evaluación de los aprendizajes en el debate didáctico contemporáneo*. Argentina, Paidós.
- Compañ P., Satorre R., Llorens, F., & Molina, R. (2015). *Enseñando a programar: un camino directo para desarrollar el pensamiento computacional*. RED. Revista de Educación

- a Distancia, (46), 1-15, e-e-ISSN: 1578-7680. Recuperado de: <https://revistas.um.es/red/article/view/240191>
- Cubero-Ibáñez, J., Ibarra-Sáiz, M.S. y Rodríguez-Gómez, G. (2018). Propuesta metodológica de evaluación para evaluar competencias a través de tareas complejas en entornos virtuales de aprendizaje. *Revista de Investigación Educativa*, 36(1), 159-184. DOI: <http://dx.doi.org/10.6018/rie.36.1.278301>
 - Díaz Barriga A., Frida & Barroso Bravo, Ramsés (2014), "*Diseño y validación de una propuesta de evaluación auténtica de competencias en un programa de formación de docentes de educación básica en México*". *Perspectiva Educacional, Formación de Profesores* [en línea] 2014, 53 (Enero) : [Fecha de consulta: 24 de noviembre de 2018] Disponible en: <http://www.redalyc.org/articulo.oa?id=333329700004> ISSN 0716-0488.(Consultado 10/10/19).
 - Flavell, J. (1976). *Metacognitive aspects of problem solving. The nature of intelligence*, 231-235, ISBN: 04700-13842. Hillsdale, New Jersey: Lawrence Erlbaum Associates.
 - Garcia Aretio L. (2008). *Evaluación en formatos no presenciales*. Editorial Del Bened. Enero 2000. Disponible en: <http://e-spacio.uned.es/fez/eserv/bibliuned:20091/evalform.pdf> (consultado el 20/11/2018).
 - Ibarra-Sáiz, M.S., & Rodríguez-Gómez, G. (2014b). Formación del profesorado Universitario en evaluación: análisis y prospectiva del Programa Formativo EVAPES-DevalSimWeb "Evaluación para el aprendizaje en la Educación Superior". En Congreso Iberoamericano de Ciencia, Tecnología, Innovación y Educación. Buenos Aires. Organización de Estados Iberoamericanos para la Educación, la Ciencia y la Cultura. Recuperado de <https://goo.gl/MFmbEa>
 - López Reguera J., Hernández Rivas C., Farran Leiva Y. (2011). Una plataforma de evaluación automática con una metodología efectiva para la enseñanza/aprendizaje en programación de computadores. Marzo 2010. *Revista chilena de ingeniería*, vol. 19 N° 2, 2011, pp. 265-277. Disponible en: <https://scielo.conicyt.cl/pdf/ingeniare/v19n2/art11.pdf> (consultado 13/10/19).
 - Meriño Almaguer Y., Lorente Rodríguez A. E., Maribona M. G. (2011). Propuesta de instrumentos de evaluación para entornos virtuales de aprendizaje: una experiencia en la universidad de las ciencias informáticas. *Revista de Informática Educativa y Medios Audiovisuales* Vol. 8(15) , págs. 1-8. 2011 1
ISSN 1667-8338 © LIE-FI-UBA. liema@fi.uba.ar .
Recuperado de: http://bibliografia.eovirtual.com/Meri%C3%B1oY_2011_Propuesta.pdf (Consultado 10/10/19).
 - Reguera, Jorge & Rivas, Cecilia & Leiva, Yussef. (2011). Una plataforma de evaluación automática con una metodología efectiva para la enseñanza/aprendizaje en programación de computadores. *Ingeniare. Revista chilena de ingeniería*. 19. 265-277. 10.4067/S0718-33052011000200011. Consultado: 30/10/2019. Disponible en: https://scielo.conicyt.cl/scielo.php?script=sci_abstract&pid=S0718-33052011000200011&lng=n&nrm=iso
 - Salinas j., De Benito b. Y Iizana a. (2014). "Competencias docentes para los nuevos escenarios de aprendizaje ". ISSN 0213-8646 • *Revista Interuniversitaria de Formación del Profesorado*, 79 (28.1) (2014), 145-163. Disponible en:

<http://gte.uib.es/pape/gte/sites/gte.uib.es/pape/gte/files/salinas-debenito-Lizana.pdf>[Última consulta: 15/07/19].

- T. Götschi, I. Sanders and V. Galpin (2003). "Mental Models of Recursion". ACM SIGCSE Bulletin. Vol. 35, Issue 1. January, 2003.
- Trilling, B., & Fadel, C. (2009). 21st Century Skills: Learning for Life in Our Times. Primera edición, ISBN: 978-0-470-47538-6. San Francisco, C.A.: John Wiley & Sons.
- Valcárcel n-m., Cabellos A., Aviles R. H. (2012) "La evaluación del aprendizaje en entornos virtuales de enseñanza aprendizaje: notas para una reflexión". Revista Iberoamericana de Educación / Revista Ibero-americana de Educação. ISSN: 1681-5653. n.º 58/2 – 15/02/12. Organización de Estados Iberoamericanos para la Educación, la Ciencia y la Cultura (OEI-CAEU). Disponible en: <https://dialnet.unirioja.es/servlet/articulo?codigo=5894699>
- Valverde-Berrocoso, J., Fernández-Sánchez, M.R., Garrido-Arroyo, M.C. (2015). El pensamiento computacional y las nuevas ecologías del aprendizaje. *RED, Revista de Educación a Distancia*. Número 46. Número monográfico sobre «Pensamiento Computacional». Septiembre de 2015. Consultado el (30/10/19) en <http://www.um.es/ead/red/46>
- Wing, J. (noviembre, 2010). Computational Thinking: What and Why? Disponible el 19 de agosto de 2011.
En: <http://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf> (Consultado 30/10/19).
- Wing, J. (marzo, 2006). Computational Thinking. View Point. Communication of ACM. Vol. 49, Número 3, marzo de 2006. Páginas 33-35. Disponible el 19 de agosto de 2011 en <http://www.cs.cmu.edu/afs/cs/usr/wing/www/publications/Wing06.pdf>. (Consultado: 13/10/19).
- Zapata Ros, M. (2015). Pensamiento Computacional: Una nueva alfabetización digital. *Revista de educación a distancia RED*, 46(4)
doi: 10.6018/red/46/. Disponible en <https://revistas.um.es/riite/article/view/267411/223391> (Consultado: 30/10/19).
- Zapotecatl, J.L. (2014). Pensamiento computacional [web log post]. Recuperado de <https://www.redalyc.org/jatsRepo/4418/441849209007/html/index.html>, 8, 9-13,15. (Consultado 30/10/19).
- LUIS MARÍA DELGADO MARTÍNEZ. Aprendizaje centrado en el estudiante, hacia un nuevo arquetipo docente. *Enseñanza & Teaching*, 37, 1-2019, 139-154.
- OCDE (2011), La medición del aprendizaje de los alumnos: Mejores prácticas para evaluar el valor agregado de las escuelas, OECD Publishing. México. Consultado el 18/02/2014. Disponible en: <http://dx.doi.org/10.1787/9789264090170-es>
- Korkmaz, Özgen & Cakir, Recep & Ozden, Muhammet. (2017). A validity and reliability study of the Computational Thinking Scales (CTS). *Computers in Human Behavior*. 72. 10.1016/j.chb.2017.01.005.
- *Revista de Filosofía* Vol. 38, N°99, (Sep-Dic) (2021-3), pp. 369 - 379 Universidad del Zulia. Maracaibo-Venezuela ISSN: 0798-1171 / e-ISSN: 2477-9598 Carencias en la evaluación del pensamiento computacional. Consultado: 14/3/24. Disponible en:

<https://www.researchgate.net/publication/355977436> Carencias en la evaluacion del p
[ensamiento computacional](#)

- Revista iberoamericana de educación. N.º 64 (2014), pp. 11-25 (issn: 1022-6508). Disponible en: <https://dialnet.unirioja.es/servlet/articulo?codigo=4774107>
- Lockwood, J., & Mooney, A. (2018). Computational Thinking in Secondary Education: Where Does It Fit? A Systematic Literary Review. International Journal of Computer Science Education in Schools, 2, 41-60. Consultado: 14/3/24. Disponible: <https://doi.org/10.21585/ijcses.v2i1.2>.
- García-Valcárcel-Muñoz-Repiso y Caballero-González (2019). Robótica para desarrollar el pensamiento computacional en la educación infantil. Consultado el 14/4/24. Disponible en: <https://dialnet.unirioja.es/servlet/articulo?codigo=6868305>

Anexo I

1.- Indique cual es la opción correcta y más completa del término computadora *

- ☐ a. Una computadora convierte datos en información.
- ☐ b. Una computadora es un dispositivo electrónico utilizado para procesar información y obtener resultados.
- ☐ c. Una máquina electrónica con componentes físicos y lógicos
- ☐ d. Un dispositivo informático capaz de recibir, almacenar y procesar información de una forma útil

2.- ¿Cuáles son las funciones de una computadora? *

- ☐ a. Procesar grandes volúmenes de información.
- ☐ b. Recibir y procesar datos.
- ☐ c. Procesar de forma rápida gran cantidad de datos.
- ☐ d. Realiza entrada-proceso-salida de datos.

3.- ¿Es lo mismo dato e Información? ¿Por qué? *

Tu respuesta _____

4.- ¿Qué entiende por problema?. De algún ejemplo de la vida cotidiana. *

Tu respuesta _____

5.- ¿Qué realiza usted para resolverlo? *

- ☐ Pensar alternativas de solución.
- ☐ Realizar acciones tendientes a resolverlo.
- ☐ Definirlo y dividirlo en partes más pequeñas.
- ☐ Incorporar soluciones creativas.
- ☐ Otras acciones.

6.- ¿Qué entiende por algoritmo? *

Tu respuesta

7.- ¿Qué es la programación de computadoras? *

☐ Es la acción de escribir código en un lenguaje a ejecutar por una computadora.

☐ Escribir código que pueda ser interpretado por una computadora.

☐ La ejecución de un algoritmo en algún lenguaje de programación

☐ Otra respuesta

8.- Si se le presenta el caso de encender un automóvil ¿Usted qué haría? Describa *

los pasos.

Tu respuesta

9.- ¿Qué otra actividad realiza paso a paso? *

Tu respuesta

10.- Conoce acerca de *

☐ Pilas-bloques

☐ Scratch

☐ App Inventor

☐ Lenguajes como: Java, C, Python, PHP

☐ LightBot

☐ Otros

Enviar

Borrar formulario

Figura 24: Formato de la encuesta inicial a estudiantes de la IE de la FaCENA UNNE
Fuente: elaboración propia

Anexo II

Respuesta 1:

Inicio:

Salimos con amigos.

Si llueve --- → proponemos ir al cine a ver una película

Si no llueve- → vamos a pescar

Luego del paseo - → comentar experiencias

Final

Respuesta 2:

Iniciar algoritmo

Salir con amigos

Si llueve vamos al cine

Si no llueve vamos a pescar

Juntarnos en la casa de alguno

Comentar lo realizado

Finalizar algoritmo

Respuesta 3:

Inicio

1.- planear la salida

2.- ver el estado del tiempo

3.- si el día es soleado entonces

Ir a pescar

4.- si el día es lluvioso entonces

Ir al cine

5.- comentar lo vivido.

Final

Respuesta 4:

Inicio algoritmo

1.- planear la salida con amigos

2.- si llueve (vamos al cine)

3.- si no llueve (vamos a pescar)

4.- Al finalizar el paseo

5.- Nos reunimos en casa de uno

6.- hablamos de lo vivido

final

Respuesta 5:

Comenzar

1.- juntarnos con amigos

2.- planear salida

2.- si llueve - → ir al cine

3.- si no llueve → ir a pescar

4.- volver a casa de un amigo

5.- contar lo vivido

finalizar

Respuesta 6:

Inicio

1.- planear la salida

2.- si llueve (ir al cine)

3.- si no llueve (ir a pescar)

*4.- Si llegó el fin - entonces
reunirnos en casa de alguno*

6.- relatar lo vivido

Fin

Como tareas similares los alumnos escribieron los siguiente:

Preparar un buen mate:

- 1.- *tomar el mate*
- 2.- *colocar la yerba en el mate*
- 3.- *colocar el agua en la pava*
- 4.- *calentar el agua*
 - *si está caliente*
 - 6.- *colocar el agua en el mate.*
 - 7.- *tomar la bombilla*
 - 8.- *colocar la bombilla*
 - 9.- *tomar el mate*
- 7.- *si no está caliente*
 - *seguir calentando el agua.*
 - *repetir pasos 6-7-8-9*

Finalizar

Lavarse correctamente las manos:

- 1.- *abrir la canilla de agua*
- 2.- *mojar las manos*
- 3.- *tomar el jabón*
- 4.- *enjabonar las manos*
- 5.- *dejar el jabón*
- 6.- *seguir esparciendo el jabón por ambas manos.*
- 7.- *enjuagar las manos*
- 8.- *secar ambas manos*

Preparar un sándwich

- 1.- *tomar un pan*
- 2.- *tomar un cuchillo*
- 3.- *partir el pan en 2 mitades*
- 4.- *tomar una feta de fiambre de la heladera*
- 5.- *colocar la feta en la mitad del pan*
- 5.- *tomar una feta de queso de la heladera*
- 6.- *colocar la feta en la mitad del pan*
- 7.- *cerrar ambas mitades del pan*

Tender la cama

- 1.- *tomar la sabana de abajo*
- 2.- *colocar la sabana al colchón*
- 3.- *tomar una funda*
- 4.- *colocar la funda a la almohada*
- 5.- *colocar la almohada arriba del colchón*
- 6.- *tomar la sabana de arriba*
- 7.- *colocar la sabana de arriba pasando arriba de la almohada*
- 8.- *hacer el dobléz de la almohada*

Realizar una suma de 3 numeros en el celular

- 1.- abrir la aplicación calculadora
- 2.- introducir un número y apretar el símbolo +
- 3.- introducir el segundo número y apretar el símbolo +
- 4.- introducir el tercer número y apretar el símbolo =
- 5.- Se visualiza el resultado

Realizar actividad física

- 1.- buscar ropa adecuada en el placar
- 2.- ponernos la ropa
- 3.- ponernos las zapatillas
- 4.- atar las zapatillas
- 5.- ir al gimnasio
- 6.- hacer ejercicios por una hora
7. al finalizar regresar.

Anexo III

Pregunta 1
Sin contestar
Puntaje de 1

Señalar con bandera la pregunta

Editar pregunta

Indique que pasos incluye la fase Diseño del algoritmo

Seleccione una:

- ☐ a. herramientas de programación
- ☐ b. diseño descendente
- ☐ c. refinamiento paso a paso
- ☐ d. Todas son correctas

Figura 25: pregunta 1 de Plataforma Moodle

Pregunta 2
Sin contestar
Puntaje de 1

Señalar con bandera la pregunta

Editar pregunta

Para confeccionar un diagrama de flujo, es aconsejable respetar reglas. Indique la/las correctas.

Seleccione una:

- ☐ a. Debe tener un comienzo y un fin
- ☐ b. El orden en que deben escribirse los símbolos es de arriba abajo y de izquierda a derecha
- ☐ c. Todas son correctas
- ☐ d. El diagrama de flujo deberá ser lo mas claro posible de forma tal que cualquier otro programador pueda seguirlo o usarlo con total facilidad de entendimiento.

Su respuesta es incorrecta.

La respuesta correcta es: Todas son correctas

Pregunta 3
Sin contestar
Puntaje de 1

Señalar con bandera la pregunta

Editar pregunta

Para confeccionar un diagrama de flujo, es aconsejable respetar reglas. Indique la/las correctas.

Seleccione una o más de una:

- ☐ a. El orden en que deben escribirse los símbolos es de arriba abajo y de izquierda a derecha
- ☐ b. No hace falta emplear un símbolo para cada acción
- ☐ c. De todos los símbolos pueden salir varias flechas
- ☐ d. Dentro de los símbolos se puede especificar instrucciones propias de algún lenguaje de programación
- ☐ e. Cuando trabajamos con operaciones lógicas recurrir preferentemente a la lógica positiva
- ☐ f. Dentro de los símbolos no especificar instrucciones propias de algún lenguaje de programación.

Figura 26: pregunta 2 y 3 de Plataforma Moodle

Anulaciones

Preguntas

Vista previa

Roles asignados localmente

Comprobar los permisos

Filtros

Desglose de competencia

Ritmo

Pregunta 4
Sin contestar
Puntaje de 1

Señalar con bandera la pregunta

Editar pregunta

Indique porque un algoritmo es finito

Seleccione una:

- ☐ a. Cada paso del algoritmo debe indicar la acción a realizar sin criterios de interpretación.
- ☐ b. Tiene un número específico y numerable de pasos
- ☐ c. Debe siempre devolver un resultado
- ☐ d. El tiempo y esfuerzo por cada paso realizado debe ser preciso

Figura 27: pregunta 4 de Plataforma Moodle

1 2 3 4 5

Terminar intento ...

Nueva vista previa

Recursos UNNE-Virtual

Tutoriales

USO DE LA PLATAFORMA MOODLE UV

Guía

TRABAJAR UNA ASIGNATURA PARA EAD

Recomendaciones

OPTIMIZACIÓN DE RECURSOS DE LA PLATAFORMA VIRTUAL

Biblioteca eLibro

Pregunta 1

Sin responder aún

Puntaje de 1.00

✓ Señalar con bandera la pregunta

⚙ Editar pregunta

Cuando se compila un programa en C, el programa que genera tiene extensión .o (objeto)

Elija una:

☐ Verdadero

☐ Falso

Pregunta 2

Sin responder aún

Puntaje de 1.00

✓ Señalar con bandera la pregunta

⚙ Editar pregunta

Indique la característica del Lenguaje C

Seleccione una:

☐ a. Lenguaje estructurado

☐ b. Lenguaje portable

☐ c. Lenguaje pequeño

☐ d. Todas son correctas

Tiempo restante 0:05:55

Figura 28: preguntas 1 y 2 referente a lenguaje C de Plataforma Moodle

Nueva vista previa

Recursos UNNE-Virtual

Tutoriales

USO DE LA PLATAFORMA MOODLE UV

Guía

TRABAJAR UNA ASIGNATURA PARA EAD

Recomendaciones

OPTIMIZACIÓN DE RECURSOS DE LA PLATAFORMA VIRTUAL

Biblioteca eLibro

Pregunta 1

Sin responder aún

Puntaje de 1.00

✓ Señalar con bandera la pregunta

⚙ Editar pregunta

Se dispone de un programa que sensa la temperatura de la camara frigorifica, se ingresa la temperatura de la camara por pantalla y el estado de la puerta de la camara (1 es ABIERTA o 0 es CERRADA).

Se desea que cuando la puerta este abierta y la temperatura sea inferior a 5°C imprima el valor de la temperatura, en cambio si supera los 5°C muestre un alerta de "PUERTAABIERTA".

Si la temperatura es superior a los 10°C y la puerta esta abierta, que muestre un mensaje de "TEMPERATURAALTA"

Teniendo el siguiente programa, indique cual es el bloque de instrucción correcto:

```
#include <stdio.h>
#include <stdlib.h>

int main() {

    int puerta = 0;
    int temperatura = 0;

    printf("Ingrese la temperatura: ");
    scanf("%d", temperatura);

    printf("Ingrese el estado de la puerta: ");
    scanf("%d", puerta);

    _____ //Seleccione el bloque de codigo correcto

}
```

Seleccione una:

☐ a. if(puerta == 1 && temperatura < 5){

Figura 29: preguntas 3 y 4 referente a lenguaje C de Plataforma Moodle

1 2 3 4 5

Terminar intento ...

Nueva vista previa

Recursos UNNE-Virtual

Tutoriales

USO DE LA PLATAFORMA MOODLE UV

Guía

TRABAJAR UNA ASIGNATURA PARA EAD

Biblioteca eLibro

Pregunta 5

Sin responder aún

Puntaje de 1.00

✓ Señalar con bandera la pregunta

⚙ Editar pregunta

La función principal main ¿para que se utiliza?

Seleccione una:

☐ a. Donde se declaran las funciones a utilizar

☐ b. Ninguna es correcta

☐ c. Para ejecutar el preprocesador

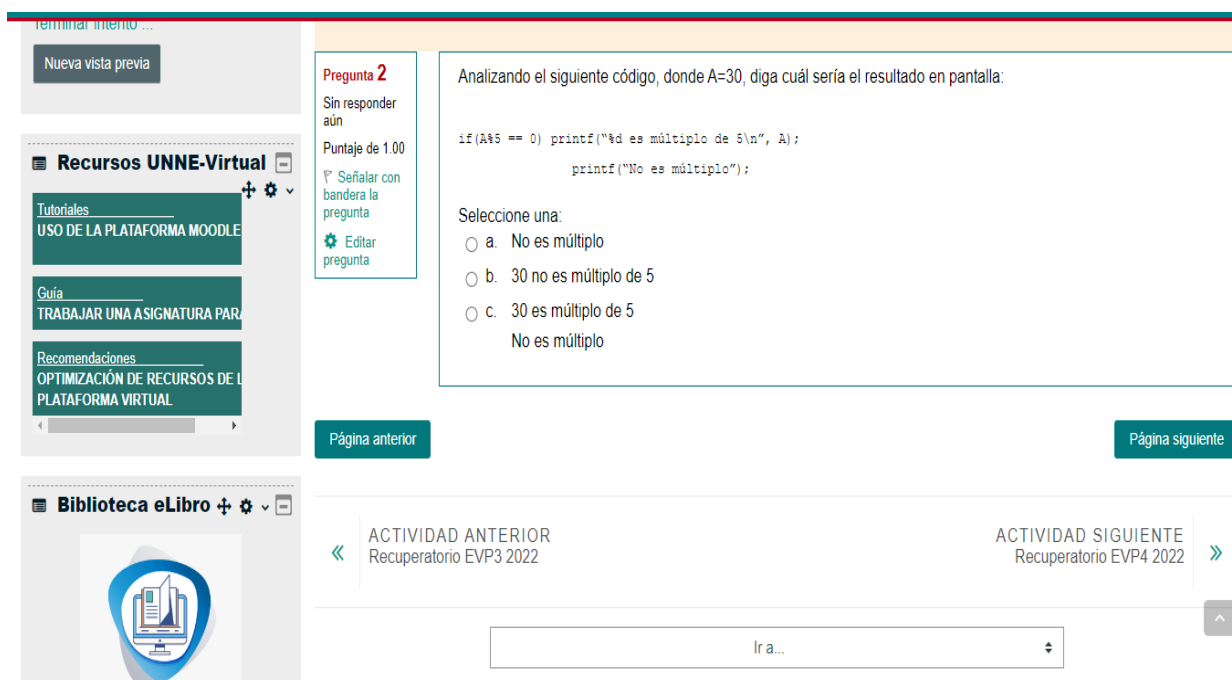
☐ d. Donde se inician todas las instrucciones programadas

Página anterior

Terminar intento ...

Tiempo restante 0:05:09

Figura 30: pregunta 5 referente a lenguaje C de Plataforma Moodle



The screenshot shows the Moodle interface for a question. On the left, there is a sidebar with 'Recursos UNNE-Virtual' and 'Biblioteca eLibro'. The main content area displays 'Pregunta 2' with a score of 1.00. The question text asks for the output of a C code snippet where A=30. The code is:

```
if(A%5 == 0) printf("%d es múltiplo de 5\n", A);  
printf("No es múltiplo");
```

 The options are: a. No es múltiplo, b. 30 no es múltiplo de 5, and c. 30 es múltiplo de 5. The interface includes navigation buttons like 'Página anterior', 'Página siguiente', and 'ACTIVIDAD ANTERIOR'.

Pregunta 2
Sin responder aún
Puntaje de 1.00
Señalar con bandera la pregunta
Editar pregunta

Analizando el siguiente código, donde A=30, diga cuál sería el resultado en pantalla:

```
if(A%5 == 0) printf("%d es múltiplo de 5\n", A);  
printf("No es múltiplo");
```

Seleccione una:

- ☐ a. No es múltiplo
- ☐ b. 30 no es múltiplo de 5
- ☐ c. 30 es múltiplo de 5

No es múltiplo

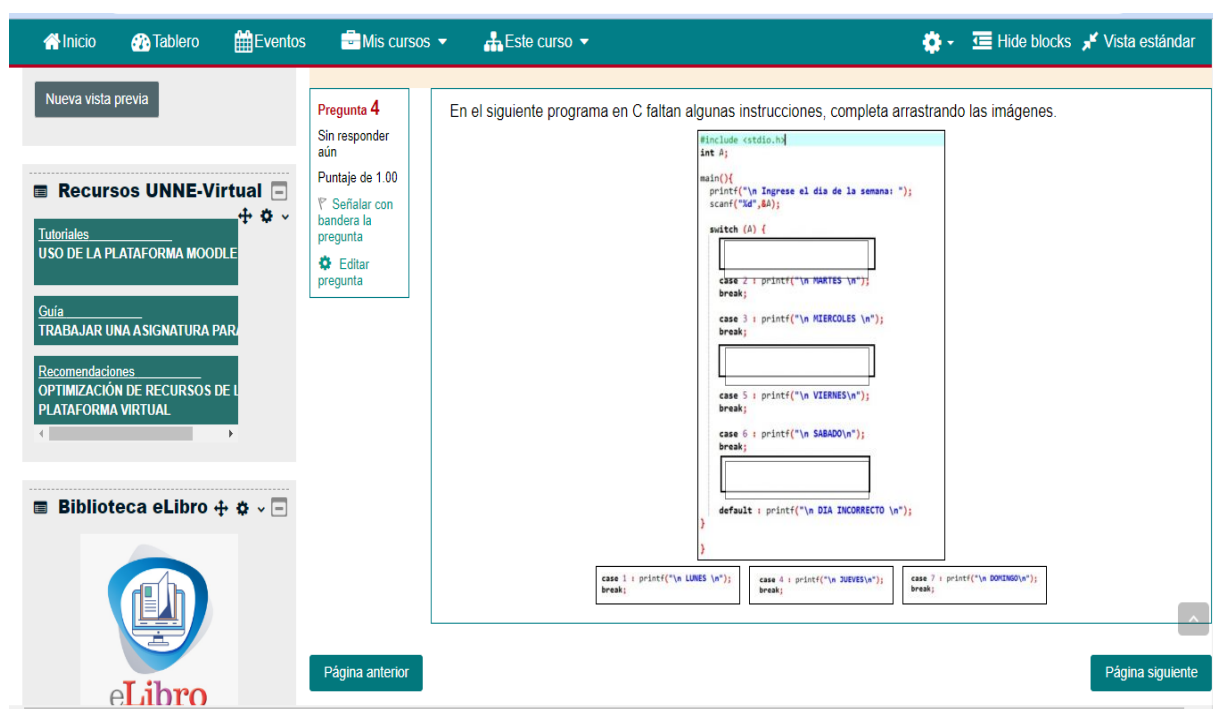
Página anterior Página siguiente

ACTIVIDAD ANTERIOR
Recuperatorio EVP3 2022

ACTIVIDAD SIGUIENTE
Recuperatorio EVP4 2022

Ir a...

Figura 31: pregunta 2 referente a selectiva if en C de Plataforma Moodle



The screenshot shows the Moodle interface for a question. On the left, there is a sidebar with 'Recursos UNNE-Virtual' and 'Biblioteca eLibro'. The main content area displays 'Pregunta 4' with a score of 1.00. The question text asks to complete a C code snippet by dragging images. The code is:

```
#include <stdio.h>  
int A;  
main(){  
printf("\n Ingrese el día de la semana: ");  
scanf("%d",&A);  
switch (A) {  
case 2 : printf("\n MARTES \n");  
break;  
case 3 : printf("\n MIERCOLES \n");  
break;  
case 5 : printf("\n VIERNES \n");  
break;  
case 6 : printf("\n SABADO \n");  
break;  
default : printf("\n DIA INCORRECTO \n");  
}
```

 The interface includes navigation buttons like 'Página anterior', 'Página siguiente', and 'ACTIVIDAD ANTERIOR'.

Pregunta 4
Sin responder aún
Puntaje de 1.00
Señalar con bandera la pregunta
Editar pregunta

En el siguiente programa en C faltan algunas instrucciones, completa arrastrando las imágenes.

```
#include <stdio.h>  
int A;  
main(){  
printf("\n Ingrese el día de la semana: ");  
scanf("%d",&A);  
switch (A) {  
case 2 : printf("\n MARTES \n");  
break;  
case 3 : printf("\n MIERCOLES \n");  
break;  
case 5 : printf("\n VIERNES \n");  
break;  
case 6 : printf("\n SABADO \n");  
break;  
default : printf("\n DIA INCORRECTO \n");  
}
```

case 1 : printf("\n LUNES \n"); break;
case 4 : printf("\n JUEVES \n"); break;
case 7 : printf("\n DOMINGO \n"); break;

Página anterior Página siguiente

Figura 32: pregunta 4 referente a selectiva if en C de Plataforma Moodle

Inicio Tablero Eventos Mis cursos Este curso

Terminar intento ...

Nueva vista previa

Recursos UNNE-Virtual

Tutoriales
USO DE LA PLATAFORMA MOODLE

Guía
TRABAJAR UNA ASIGNATURA PARA

Recomendaciones
OPTIMIZACIÓN DE RECURSOS DE LA PLATAFORMA VIRTUAL

Biblioteca eLibro

Pregunta 5
Sin responder aún
Puntaje de 1.00

Señalar con bandera la pregunta
Editar pregunta

Arrastrar y soltar la acción que corresponde para la evaluación de la condición en esta porción de código

```
printf("Introduzca un valor entero\n");
scanf("%d ", &a1);
if ( a1 >= 0 )
    if ( a1 <= 10 )
    else
    printf("el numero no esta entre 0 y 10\n");
    printf("el numero esta entre 0 y 10\n");
```

Página anterior Terminar intento ...

ACTIVIDAD ANTERIOR
Recuperatorio EVP3 2022

ACTIVIDAD SIGUIENTE
Recuperatorio EVP4 2022

Ir a...

Figura 33: pregunta 54 referente a selectiva if en C de Plataforma Moodle