

UNIVERSIDAD NACIONAL DANIEL ALCIDES CARRIÓN

FACULTAD DE CIENCIAS DE LA EDUCACIÓN

ESCUELA DE FORMACIÓN PROFESIONAL DE EDUCACIÓN

SECUNDARIA



T E S I S

**Aplicación de algoritmos para el aprendizaje de la programación
informática en estudiantes universitarios de la Filial Yanahuanca,**

2023

Para optar el título profesional de:

Licenciado en Educación

Con Mención: Tecnología Informática y Telecomunicaciones

Autores:

Bach. Eliseo David BARRETO REQUIZ

Bach. Kevin OLIVAS ARRIETA

Asesor:

Dr. José Rovino ALVAREZ LOPEZ

Yanahuanca – Perú - 2023

UNIVERSIDAD NACIONAL DANIEL ALCIDES CARRIÓN
FACULTAD DE CIENCIAS DE LA EDUCACIÓN
ESCUELA DE FORMACIÓN PROFESIONAL DE EDUCACIÓN
SECUNDARIA



T E S I S

**Aplicación de algoritmos para el aprendizaje de la programación
informática en estudiantes universitarios de la Filial Yanahuanca,
2023**

Sustentada y aprobada ante los miembros del jurado:

Mg. Juan Antonio CARBAJAL MAYHUA
PRESIDENTE

Mg. Jorge BERROSPI FELICIANO
MIEMBRO

Mg. Shuffer GAMARRA ROJAS
MIEMBRO



Universidad Nacional Daniel Alcides Carrión
Facultad de Ciencias de la Educación
Unidad de Investigación

INFORME DE ORIGINALIDAD N° 68-2023

La Unidad de Investigación de la Facultad de Ciencias de la Educación de la Universidad Nacional Daniel Alcides Carrión ha realizado el análisis con exclusiones en el Software Turnitin Similarity, que a continuación se detalla:

Presentado por:

BARRETO REQUIZ, Eliseo David y OLIVAS ARRIETA, Kevin

Escuela de Formación Profesional

Educación Secundaria

Tipo de trabajo: **Tesis**

Título del trabajo

**Aplicación de algoritmos para el aprendizaje de la programación
Informática en estudiantes universitarios de la Filial Yanahuanca, 2023**

Asesor:

ALVAREZ LOPEZ, José Rovino

Índice de Similitud: **17%**

Calificativo

APROBADO

Se adjunta al presente el informe y el reporte de evaluación del software Turnitin similarity.

Cerro de Pasco, 16 de agosto del 2023


Dr. Jacinto Alejandro Alejos Lopez
Director (e) Unidad de Investigación
Facultad de Ciencias de la
Educación

DEDICATORIA

A Dios y mis padres, por su amor, paciencia y sacrificio, lo cual han sido la base fundamental para poder perseguir mis sueños.

A mi amada pareja y mi hijo, por ser mis mejores regalos que Dios me ha dado, sus sonrisas iluminan mi mundo, alegran mi mañana y emocionan mi noche, ellos son mi fuente de motivación e inspiración para superarme cada día más y más.

(Kevin OLIVAS ARRIETA)

A Dios por ayudarme en todos los momentos de mi vida, a mi familia por apoyarme en todas las dificultades que tuve, especialmente a mi madre quien fue una de mis mayores inspiraciones para poder estudiar y seguir adelante.

A mi maestro Dr. José Rovino ALVAREZ LOPEZ, por haberme brindando sus enseñanzas, que me fueron muy útiles para poder elaborar mi tesis.

(Eliseo David)

AGRADECIMIENTO

A ti, Dios, por tu amor infinito y por ser nuestro refugio en momentos de dificultad. Gracias por tus constantes cuidados, por escuchar nuestras oraciones y por sostenernos cuando sentíamos que flaqueábamos. Tu presencia en nuestras vidas nos ha dado la confianza y la fuerza necesaria para superar los desafíos y seguir adelante.

Agradecemos a la Universidad Daniel Alcides Carrión por brindarnos la oportunidad de adquirir conocimientos, explorar nuevas ideas y crecer intelectualmente. El ambiente académico y los recursos ofrecidos nos han impulsado a alcanzar metas más altas y a superar nuestros propios límites.

A los docentes, quienes han compartido sus conocimientos, experiencias con nosotros, les agradecemos por su dedicación y desafiarnos a crecer intelectualmente. Asimismo, sus enseñanzas han sido fundamentales para nuestro desarrollo académico y personal. A través de su apoyo y aliento, hemos adquirido habilidades, perspectivas que nos serán valiosas a lo largo de nuestras vidas.

Queremos expresar nuestro más profundo agradecimiento a nuestro asesor, José Rovino Álvarez López. Su guía, paciencia y experiencia han sido invaluable en el proceso de investigación y redacción de esta tesis. Por su constante apoyo, por compartir sus conocimientos y por ayudarnos a pulir nuestras ideas para hacer de este trabajo un reflejo de excelencia académica.

A nuestra familia, por ser nuestra roca y apoyo incondicional, su amor, comprensión y aliento han sido nuestro motor durante esta desafiante trayectoria.

RESUMEN

La investigación se desarrolló con el propósito de demostrar la influencia de los algoritmos en el aprendizaje de la programación informática con sus respectivas dimensiones, indicadores e ítems propuestos en los estudiantes del III ciclo del Programa de Estudios Tecnología Informática y Telecomunicaciones filial Yanahuanca - 2023. La metodología utilizada corresponde a un estudio de enfoque cuantitativo, nivel explicativo donde se ha manipulado la variable independiente desarrollando sesiones de aprendizaje enfatizando la variable dependiente para comprobar el experimento respectivo, su diseño fue pre experimental con un solo grupo, tomando como muestra a 16 estudiantes del ciclo respectivo, quienes participaron directamente en las sesiones de aprendizaje desarrolladas en el curso de Metodología de la Programación. El resultado obtenido ha demostrado que existe influencia significativa por la diferencia de medias equivalente a 08.06 en la pre prueba y 12.50 en la posprueba, además el valor de significancia en la correlacional y prueba t es $0.000 < 0.05$, asimismo el valor de t de student fue -4.886 frente a -1.753 de la tabla, con un valor de significancia de $0.000 < 0.05$ por lo que se acepta la hipótesis de investigación y se rechaza la nula, demostrando de manera pertinente que las herramientas, recursos y fases de los algoritmos han permitido desarrollar y fortalecer las capacidades de programación con el uso intenso de Pseint como herramienta de comprobación de pseudocódigos.

Palabras Clave:

Algoritmo / Programación / Definición del problema / Elaboración de pseudocódigos / Codificación y prueba /

ABSTRACT

The research was developed with the purpose of demonstrating the influence of algorithms in the learning of computer programming with their respective dimensions, indicators and items proposed in the students of the III cycle of the Program of Studies Information Technology and Telecommunications subsidiary Yanahuanca - 2023. The methodology used corresponds to a study with a quantitative approach, an explanatory level where the independent variable has been manipulated, developing learning sessions emphasizing the dependent variable to verify the respective experiment, its design was pre-experimental with a single group, taking as a sample 16 students from the respective cycle, who directly participated in the learning sessions developed in the Programming Methodology course. The result obtained has shown that there is a significant influence due to the difference in means equivalent to 08.06 in the pre-test and 12.50 in the post-test, in addition the value of significance in the correlational and t-test is $0.000 < 0.05$, as well as the value of student's t was -4.886 compared to -1.753 in the table, with a significance value of $0.000 < 0.05$, therefore the research hypothesis is accepted and the null is rejected, pertinently demonstrating that the tools, resources and phases of the algorithms have allowed to develop and strengthen programming capabilities with the intense use of Pseint as a pseudocode checking tool.

Keywords:

Algorithm / Programming / Problem definition / Pseudocode development / Coding and testing /

INTRODUCCIÓN

La programación de ordenadores es una disciplina emergente en los tiempos actuales, los que están inmersos en esta tarea, han emprendido una multiplicidad de tareas y manejo de una diversidad de lenguajes de propósito general, orientado a objetos y en la actualidad programación web, las habilidades que deben poseer las personas dedicadas a esta tarea en definitiva implica una disciplina rigurosa para programar; las fases iniciales que permiten desarrollar las destrezas y habilidades de programación son los algoritmos, los que en sus fases iniciales son pertinentes el manejo adecuado de cada uno de los componentes que posee.

Los estudiantes del programa de estudio, desarrollan cursos generales, específicos y de especialidad; de acuerdo al plan de estudios del programa, se desarrolló un curso exclusivo de metodología de la programación, donde el desarrollo de sesiones está orientado fundamentalmente al trabajo de los algoritmos, como línea de base para la programación de ordenadores, el mencionado curso proporciona los fundamentos necesarios para programar ordenadores utilizando un lenguaje de programación respectivamente.

El estudio está dividido en cuatro capítulos:

CAPÍTULO I: Conformado por el problema de investigación, identificación y determinación del problema, delimitación de la investigación, formulación del problema, objetivos generales y específicos, importancia y alcances de la investigación, donde se encuentra argumentos claros sobre el propósito, las metas y el impacto de la investigación en estricta relación con las variables: Algoritmos y programación informática.

CAPÍTULO II: Contiene información que da a conocer los antecedentes de estudio a nivel internacional, nacional y local, las bases teóricas científicas en estricta relación con las variables y dimensiones de cada una de ellas con sus respectivos

sustentos y constructos teóricos científicos, finalmente, la definición de términos, el planteamiento de las hipótesis, las variables y su respectiva operacionalización.

CAPÍTULO III: Considera la metodología y técnicas de estudio conformado por el tipo de investigación, diseño de la investigación, población y muestra, métodos de investigación, técnicas e instrumentos de recolección de datos, validación de los instrumentos, técnicas de procesamiento de datos y la selección y validación de instrumentos, el planteamiento de la hipótesis general, específica y nula, el sistema de variables y su correspondiente operacionalización que contiene la definición conceptual y operacional de las variables que muestra las dimensiones, indicadores e ítems que han permitido elaborar los instrumentos para el recojo de la información.

CAPÍTULO IV: Conformado por toda la información concerniente al trabajo de campo, presentando los resultados, tablas, gráficos, etc., su interpretación correspondiente y la prueba de hipótesis con la aplicación estadística correspondiente, así como la discusión de los resultados que considera una comparación directa con las investigaciones realizadas con anterioridad.

LOS AUTORES

ÍNDICE

DEDICATORIA

AGRADECIMIENTO

RESUMEN

ABSTRACT

INTRODUCCIÓN

ÍNDICE

INDICE DE TABLAS

INDICE DE FIGURAS

CAPÍTULO I

PROBLEMA DE INVESTIGACIÓN

1.1.	Identificación y determinación del problema	1
1.2.	Delimitación de la investigación	4
1.2.1.	Delimitación Espacial:	4
1.2.2.	Delimitación temporal:.....	4
1.2.3.	Delimitación Social:.....	5
1.3.	Formulación del problema.....	5
1.3.1.	Problema general.....	5
1.3.2.	Problemas específicos	5
1.4.	Formulación de objetivos	6
1.4.1.	Objetivo General:	6
1.4.2.	Objetivos Específicos:.....	6

1.5.	Justificación de la investigación	6
1.6.	Limitaciones de la investigación	7

CAPÍTULO II

MARCO TEÓRICO

2.1.	Antecedentes de estudio	9
2.2.	Bases teóricas – científicas.	17
2.2.1.	Algoritmo	17
2.2.2.	Definición del problema.....	19
2.2.3.	Elaboración del algoritmo	20
2.2.4.	Codificación del algoritmo.....	21
2.2.5.	Programación informática	22
2.2.6.	Lenguaje de programación	22
2.2.7.	Codificación y escritura de instrucciones	24
2.2.8.	Elaboración de algoritmos o programas.....	25
2.2.9.	Comprobación del funcionamiento de programas	26
2.3.	Definición de términos básicos.....	26
2.4.	Formulación de Hipótesis	29
2.4.1.	Hipótesis General	29
2.4.2.	Hipótesis Específicas	29
2.4.3.	Hipótesis nula.....	30
2.5.	Identificación de Variables.....	30

2.5.1.	Variable independiente.....	30
2.5.2.	Variable dependiente.....	30
2.6.	Definición Operacionalización de variables e indicadores.....	31
2.6.1.	Definición conceptual	31
2.6.2.	Definición Operacional.....	31

CAPÍTULO III

METODOLOGÍA Y TÉCNICAS DE INVESTIGACIÓN

3.1.	Tipo de investigación.....	34
3.2.	Nivel de investigación	34
3.3.	Métodos de investigación	34
3.4.	Diseño de investigación.....	35
3.5.	Población y muestra.....	36
3.5.1.	Población.....	36
3.5.2.	Muestra.....	36
3.6.	Técnicas e instrumentos de recolección de datos	37
3.6.1.	Técnicas.....	37
3.6.2.	Instrumentos	37
3.7.	Selección, validación y confiabilidad de los instrumentos de investigación.....	38
3.7.1.	Selección de instrumentos	38
3.7.2.	Validación de instrumentos	38
3.7.3.	Confiabilidad de instrumentos	38

3.8.	Técnicas de procesamiento y análisis de datos.....	39
3.8.1.	Procesamiento manual	39
3.8.2.	Procesamiento electrónico	39
3.9.	Tratamiento estadístico	39
3.10.	Orientación ética filosófica y epistémica.....	40

CAPÍTULO IV

RESULTADOS Y DISCUSIÓN

4.1.	Descripción del trabajo de campo.....	41
4.2.	Presentación, análisis e interpretación de resultados.....	42
4.3.	Prueba de Hipótesis	46
4.4.	Discusión de resultados	49

CONCLUSIONES

RECOMENDACIONES

REFERENCIAS BIBLIOGRÁFICAS

ANEXOS

INDICE DE TABLAS

Tabla 1.	Variable independiente: Algoritmos.....	32
Tabla 2.	Variable dependiente: Programación informática.....	33
Tabla 3.	Población de estudiantes de la filial.....	36
Tabla 4.	Resultados obtenidos en la prueba pedagógica.....	38
Tabla 5.	Resultados de prueba pedagógica.....	39
Tabla 6.	Notas obtenidas en la prueba de entrada.....	42
Tabla 7.	Notas obtenidas en las sesiones de aprendizaje.....	43
Tabla 8.	Notas obtenidas en la posprueba.....	45
Tabla 9.	Diferencia de notas de pre test y post test de la muestra de estudio.....	47
Tabla 10.	Estadísticas de muestras emparejadas.....	47
Tabla 11.	Correlaciones de muestras emparejadas.....	48
Tabla 12.	Prueba T de Student para muestras emparejadas.....	48

INDICE DE FIGURAS

Figura 1.	Resultados de la prueba de entrada – III ciclo.....	43
Figura 2.	Resultados de las sesiones de aprendizaje.....	44
Figura 3.	Resultados de la prueba de salida.....	45
Figura 4.	Prueba t con una sola cola.....	49

CAPÍTULO I

PROBLEMA DE INVESTIGACIÓN

1.1. Identificación y determinación del problema

La resolución de problemas se ha constituido en una actividad de gran importancia en la formación universitaria, por lo que, en cada disciplina de formación profesional se encuentran las asignaturas como cursos generales que van enfatizando, desarrollando o fortaleciendo habilidades para resolver problemas sin dificultades, según Nazario y otros (2009): “la matemática como instrumento que posibilita resolver diferentes problemas del entorno sociocultural, su valor formativo al contribuir al desarrollo intelectual e integral de la personalidad y la matemática como lenguaje universal de las ciencias.”

Al mismo tiempo, en investigaciones vinculadas a las tecnologías de la información se puede distinguir que las propuestas están vinculadas a la resolución de problemas con diversidad de herramientas digitales, como lo argumenta Gutiérrez, C. (2018): “La investigación asumió el Entorno Virtual de Aprendizaje, EVA, como una herramienta didáctica en el aula, con el ánimo de estimular la competencia de interpretación y solución de problemas.”

Asimismo, la interacción dada con la diversidad de recursos informáticos enfatiza que surgen nuevos entornos y uso de diversidad de herramientas para realizar múltiples actividades relacionadas con tareas académicas y de formación, al respecto, Gutiérrez, C. (2018) manifiesta:

La multialfabetización entendida como un aprendizaje que integre las diversas formas y lenguajes de representación y comunicación de la información: textual, icónica, sonora, hipertextual, audiovisual y tridimensional, entre otras, a través de distintas tecnologías; promover la innovación curricular, orientada al mejoramiento de la enseñanza y del aprendizaje (Salinas, 2011). Las TIC también impactan la motivación y concentración del estudiante en el proceso de enseñanza-aprendizaje, por las posibilidades de interacción y animación a la hora de presentar los conceptos y las actividades, lo cual permite aumentar la permanencia de los estudiantes en el aula (Claro, 2010). (p. 282).

Las competencias digitales son saberes que se deben ir fortaleciendo a partir de la formación básica o profesional a partir del uso de diversas herramientas, donde los programas informáticos sirvan para elaborar otras aplicaciones a partir de la programación como área de creación de recursos digitales, al respecto, Flores, J. (2018) argumenta:

“Un algoritmo es un conjunto prescrito de instrucciones o reglas bien definidas, ordenadas y finitas que permite realizar una actividad mediante pasos sucesivos que no generen dudas a quien deba realizar dicha actividad. En un escenario con un estado inicial y una entrada, siguiendo los pasos sucesivos se llega a un estado final y se obtiene una solución.” (p.6).

El término “pensamiento algorítmico” ha sido empleado con frecuencia por varios autores como una de habilidades relevantes para la formación en Informática (Snyder, 2000, pag.150).

Futschek (2006), en relación a la algoritmia argumenta:

El pensamiento algorítmico está compuesto por habilidades conectadas a la construcción y comprensión de algoritmos, tales como: analizar el problema dado, especificar un problema con precisión, encontrar acciones básicas adecuadas al problema dado, construir el algoritmo correcto para resolver el problema dado a partir de las acciones básicas, pensar todos los casos posibles de solución del problema dado y comprobar la eficiencia de un algoritmo (p. 160).

Los algoritmos en la enseñanza han sido empleados como herramientas de análisis e instrumentos para facilitar el aprendizaje, según Jonsson et al (2014, pág. 114). De acuerdo a Edmonds (2008):

Al momento de desarrollar los códigos de programación el estudiante se encuentra con muchas dificultades, de las cuales para poder evitarlas son necesarios hacer el análisis para ver el problema, esto permitirá una mejor comprensión. Para ello, contará con otras herramientas para la resolución de problemas. (pag.151)

El internet de las cosas es una realidad, y seguirá creciendo mucho más la necesidad de tener programadores y personas capacitadas para resolver problemas, aportar soluciones e innovar en todos los ámbitos. (Stallman, R. 2013).

Hoy en la actualidad en nuestro país sería muy importante la enseñanza de la programación en las instituciones educativas, ya que esto ayudaría a tener estudiantes con mayor capacidad y creatividad en el mundo de las tecnologías,

asimismo en la parte política más se deberían enfocar en la educación de los estudiantes, si buscamos tener una mejora en la investigación e innovación. Todo ello con los maestros preparados, la infraestructura excelente y generando interés para el desarrollo condicional del estudiante.

Los resultados obtenidos en la asignatura de Metodología de la Programación durante sus primeras unidades, en el año 2021; de 5 estudiantes han aprobado el curso 3 y desaprobado 2, donde la mayor nota es 12 y la menor es 6; al mismo tiempo, en el año 2022, de 13 estudiantes han desaprobado 3 cuyas notas fluctúan de 06 a 10; han aprobado 10, cuyas notas se ubican de 11 a 18; lo que demuestra que la capacidad de resolución de problemas se encuentra en un nivel de inicio, asimismo, es importante que los estudiantes desarrollen la programación informática y sigan aprendiendo cada día nuevas estrategias basadas en la resolución de problemas con algoritmos a partir del uso de herramientas como los pseudocódigos, los diagramas de flujo para incorporarlos al área de programación utilizando lenguajes posteriormente

1.2. Delimitación de la investigación

1.2.1. Delimitación Espacial:

El desarrollo de la investigación se ha realizado en el Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca del Distrito de Yanahuanca, Provincia de Daniel Alcides Carrión, Región Pasco.

1.2.2. Delimitación temporal:

El periodo para ejecutar la presente investigación fue desde el mes de abril hasta junio de 2023.

1.2.3. Delimitación Social:

La investigación se realizó en estudiantes del III ciclo del programa de estudios de Tecnología Informática y Telecomunicaciones, UNDAC, Filial Yanahuanca.

1.3. Formulación del problema

1.3.1. Problema general

¿Cuál es la influencia de la aplicación de los algoritmos para el aprendizaje de la programación informática en estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023?

1.3.2. Problemas específicos

- a) ¿Cuál es el nivel de aprendizaje de la programación informática antes de la aplicación de algoritmos en estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023?
- b) ¿Qué características presenta la aplicación de los algoritmos para el aprendizaje de la programación informática en estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023?
- c) ¿Cuál es el nivel de aprendizaje de la programación informática después de la aplicación de los algoritmos en estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023?

1.4. Formulación de objetivos

1.4.1. Objetivo General:

Determinar la influencia de los algoritmos para el aprendizaje de la programación informática, en estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023.

1.4.2. Objetivos Específicos:

- a) Diagnosticar el nivel aprendizaje de la programación informática antes de la aplicación de algoritmos en estudiantes del III ciclo del Programa de Estudios. de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023.
- b) Aplicar los algoritmos durante el aprendizaje de la programación informática, en estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023.
- c) Evaluar el nivel de aprendizaje de la programación informática después de la aplicación de los algoritmos en estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023.

1.5. Justificación de la investigación

El proceso de la investigación pretende encontrar los efectos, los resultados que hay entre el desarrollo de la lógica utilizando un pseudocódigo de programación y el pensamiento innovador, generando en el alumno el interés, la motivación para resolver algoritmos utilizando un software informático que permite conseguir soluciones a determinados problemas designados. Como

también, procura alcanzar en los estudiantes las distintas maneras de pensamiento y la capacidad de entender el funcionamiento de los algoritmos y aplicar distintas técnicas en la lógica computacional, adquiriendo una motivación permanente hacia los procesamientos de datos mediante el uso de los programas tecnológicos. La investigación posibilitará fortalecer los conocimientos fundamentales para resolver problemas informáticos con la aplicación de los algoritmos como conocimiento transversal para el desarrollo tecnológico de las sociedades. Asimismo, se va a trabajar por buscar un mejoramiento en la programación informática en los estudiantes de la Universidad Nacional Daniel Alcides Carrión, haciendo el uso de los recursos tecnológicos que brinda la institución, como también se escoge algoritmos como nuestro tema porque es una base necesaria en el desarrollo de nuestra lógica como programadores.

- **Nivel teórico:** La investigación permitió analizar sobre los fundamentos de una programación informática elaborando algoritmos utilizando Pseint como herramienta de representación de algoritmos.
- **Nivel práctico:** Las estrategias utilizadas de la aplicación del Pseint en el aprendizaje de la programación informática fueron pertinentes, de manera que, los estudiantes sean capaces de aplicar dichas estrategias en mejorar los conocimientos y las capacidades en la programación de ordenadores.

1.6. Limitaciones de la investigación

En el proceso de investigación realizado, se ha tenido algunas limitaciones como:

- **Limitación de tipo bibliográfico:** No todos los sitios son fáciles de acceder a la información necesaria debido a que no todos están bien actualizados, asimismo, no son de todo confiables.

- **Limitación de tipo económico:** La parte económica, fundamental en toda investigación para la aplicación de instrumentos y conducción de sesiones de aprendizaje con los alumnos de la muestra de estudio.
- **Limitación de tiempo:** Considerando el trabajo que se va realizar con los alumnos se necesita de un tiempo más prolongado para enfatizar los procesos de resolución de problemas con algoritmos.

CAPÍTULO II

MARCO TEÓRICO

2.1. Antecedentes de estudio

A nivel local

Alvarado, M. (2018). *La programación de las computadoras con micromundos pro y los procesos cognitivos básicos de los alumnos del vi ciclo de la institución educativa José Carlos Mariátegui Lachira de Michivilca – Tapuc 2017*. Cuyas conclusiones son:

Los resultados obtenidos muestran que existe una relación importante y de gran relevancia entre la programación con Micromundos Pro y los procesos cognitivos básicos de los estudiantes del ciclo VI del Institución José Carlos Mariátegui Lachira de Michivilca, ya que la correlación de Pearson dio un valor de 0.85.

Se encontró una relación adecuada y apropiada entre programar con Micromundos Pro y analizar información de diversas fuentes, teniendo en cuenta formación académica, actitudes positivas, uso de herramientas, realización de

diferentes actividades y uso de herramientas informáticas para encontrar diferentes soluciones. sobre el problema que presenta la situación.

Arias, W. & Timoteo W. (2019) “*Programación de aplicativos para equipos móviles en el mejoramiento del aprendizaje en los estudiantes del IV grado de educación secundaria del área de educación para el trabajo de la institución educativa san miguel, distrito de Acobamba provincia de tarma – región Junín*”. Cuyo resumen es:

La educación moderna requiere trabajar en el campo de la programación de tecnología móvil, ya que tiene que adaptarse a las tendencias en la investigación tecnológica y las necesidades de los estudiantes. Por lo tanto, es importante considerar la posibilidad de acceder a la información en cualquier momento y lugar a través de una variedad de dispositivos o dispositivos móviles, incluidas las soluciones informáticas para instituciones educativas y computadoras de escritorio. Se convierte en una parte importante del funcionamiento del currículo en todos los niveles. Esta tesis tiene como objetivo conocer y comprender las principales características de la programación de aplicaciones para ordenadores o dispositivos móviles. Este módulo de capacitación se desarrolló como una aplicación AppInventor para este proceso de investigación, lo que permite desarrollar una capacitación básica para demostrar cómo construirlo en la práctica. Características de las aplicaciones para Android. Desde este punto de vista, desarrollamos esta tesis teniendo en cuenta la estructura establecida por el Instituto del Ministerio de Educación y Ciencia, que se presenta para la selección del puesto ocupacional de Licenciado en Educación Secundaria, la cual consta de los siguientes capítulos: un acercamiento a un problema que explica y justifica el motivo de nuestra investigación mediante la identificación y

el análisis del problema, buscamos encontrar posibles soluciones donde es importante establecer metas y la justificación necesaria para alcanzarlas. , para estar al tanto de la validez de la investigación y las limitaciones presentadas en el desarrollo de la misma. En el Capítulo II: Marco teórico, examinamos investigaciones previas relevantes para nosotros para encontrar temas desarrollados con similitudes. De la misma forma se formularon las bases científicas y teóricas, así como las hipótesis y variables del sistema. Capítulo III: Metodología de la Investigación, nos ocupamos de un tipo de investigación cualitativa y nos limitamos al uso de métodos descriptivos-explicativos. El diseño del estudio es cuasi-experimental, considerando poblaciones y muestras para recolectar datos utilizando métodos apropiados y herramientas. Capítulo IV: Presentación y Discusión de Resultados: Se realizan procedimientos numéricos estadísticos e interpretación de datos para probar las hipótesis planteadas por el estudio, y los resultados se presentan en forma de tablas y gráficos estadísticos. Finalmente, espero que este estudio sirva como punto de partida para la investigación y desarrollo de nuevas estrategias de aprendizaje para el desarrollo de nuevas tecnologías, y al mismo tiempo, se espera que este estudio sirva como una oportunidad para impulsar el desarrollo y desarrollo. mediante la corrección de errores que se produzcan durante el desarrollo de este estudio. (p. iv)

A nivel nacional

Lagos, M. (2020). *“Método de aprendizaje cooperativo y el aprendizaje de algoritmos de programación en estudiantes de la universidad nacional de Ayacucho”*. Cuyo resumen establece:

Como alternativa a los métodos tradicionales de enseñanza utilizados durante muchos años. Detrás hay métodos de aprendizaje colaborativo utilizados

en otros contextos. Currículo en diferentes niveles. el objetivo es Este estudio pretende determinar los efectos del uso de Métodos de aprendizaje cooperativo en el aprendizaje del diseño de algoritmos Universidad Nacional de San Cristóbal de Huamanga - 2019; mediante Pruebas previas y posteriores para diseños de estudios experimentales y subtipos cuasi-experimentales Grupo control y grupo experimental al que se le aplicó el método Métodos de aprendizaje tradicional y de aprendizaje cooperativo respectivamente. Qué En general, se encontró que había diferencias significativas en las notas. Poner a prueba los conocimientos del grupo experimental con respecto al grupo de control. además, En cuanto a la definición de indicadores y el uso y uso adecuado de las variables Selección de la estructura de control de flujo en el desarrollo de algoritmos: prueba posterior No hubo diferencia significativa entre el grupo experimental y el grupo control. y en Si se encuentra una diferencia significativa con respecto al uso adecuado, la otra parte Estructuras iterativas de control de flujo y combinaciones apropiadas Selección y estructura de control de flujo iterativo del desarrollo posterior al algoritmo Prueba del grupo experimental en comparación con el grupo de control. (p. vi)

Quispe N. (2015) *“Relación del pensamiento lógico y el pensamiento creativo en el aprendizaje de la lógica de programación en los estudiantes de computación e informática del IESTP Manuel Antonio Hierro Pozo de Ayacucho durante el periodo lectivo 2013”*. Cuyo resumen establece:

El propósito de este estudio fue investigar la relación entre el pensamiento lógico y el pensamiento creativo y el estudio de la programación lógica en estudiantes de informática y ciencias de la computación de Manuel Antonio Hierro Pozo de la Institución Educativa Superior Tecnológica del Estado en

Ayacucho. Una muestra de 50 estudiantes de los Módulos Especializados II y III, aplicando la herramienta Test de Pensamiento Lógico que consta de las tres dimensiones de Razonamiento e Inteligencia Lógica, Pensamiento Numérico y Pensamiento Abstracto; El test de pensamiento creativo basado en las 15 competencias/habilidades de las personas creativas propuesto por Ponty se categoriza en competencia psicológica, competencia conductual y competencia técnica. Pruebas de entrenamiento de lógica de programación en las etapas de aprendizaje tales como conocimiento conceptual, conocimiento procedimental y conocimiento conductual. A través del procesamiento estadístico y los resultados, se puede concluir que el pensamiento lógico y el pensamiento creativo no están relacionados con el aprendizaje lógico de programación, no existe una relación significativa entre el pensamiento lógico y el aprendizaje lógico de programación, y tampoco existe una relación significativa entre el aprendizaje. lógica de programación. Pensar e investigar la lógica de la programación entre estudiantes de informática de instituciones de educación superior técnica nacional. Manuel Antonio Hierro Pozo de la ciudad de Ayacucho. (p. iv)

A nivel internacional

Pérez, H. (2017) *“Uso de Scratch como herramienta para el desarrollo del pensamiento computacional en programación y de la carrera de Informática de la Universidad Central del Ecuador”*. Cuyo resumen establece:

La investigación realizada se ha utilizado como herramienta para desarrollar inferencias computacionales en dimensiones como identificación de patrones, uso de instrucciones, uso de variables, secuencias, uso de operadores, reutilización de código, detección de errores y abstracción, centrándose en el uso de Scratch. La población escogida fueron 80 estudiantes del primer semestre de

la carrera de informática de la Facultad de Filosofía de la Universidad Central del Ecuador, con una temática de programación. Va de explicación en explicación y se apoya en una metodología mixta para interpretar los resultados. Los objetivos del estudio son generales y específicos, e incluyen: Objetivo General: Utilizar la herramienta Scratch para analizar el desarrollo de las habilidades de pensamiento computacional de los estudiantes del primer semestre de la Facultad de Informática, Facultad de Filosofía, Universidad Central del Ecuador. Un recurso didáctico en su formación profesional. Objetivo específico: Diagnosticar las habilidades de pensamiento computacional de los estudiantes del primer semestre de la carrera de Ciencias de la Computación de la Universidad Central del Ecuador. Probamos diferencias significativas en el desarrollo de habilidades de pensamiento computacional entre estudiantes que aprenden a programar usando Scratch y aquellos que no usan Scratch, analizando antes y después de la aplicación del programa de intervención. Los resultados obtenidos en el grupo experimental y el grupo de control se ordenan por género y se comparan en relación con el desarrollo de la capacidad de pensamiento computacional para establecer la diferencia en el aprendizaje. Comparar los resultados obtenidos en el grupo experimental y el grupo control formado por la dimensión de pensamiento computacional para establecer el nivel de desarrollo de cada uno. Durante la fase de recolección de datos se utilizó un cuestionario de 30 preguntas sobre pensamiento computacional, el cual fue analizado mediante estadística descriptiva e inferencial mediante cuestionarios pre-test y post-test según el momento seguido de triangulación de los resultados. resultados similares. inspección. Entre los hallazgos más importantes encontramos diferencias en el

desarrollo de habilidades de pensamiento computacional en hombres y mujeres, y que Scratch promueve el reconocimiento de patrones en los estudiantes. (p. 12)

Caneo, O. (2002) *“Las practicas cooperativas como mejora del aprendizaje en programación de computadoras”*. Cuyo resumen establece:

Este estudio trata sobre problemas de aprendizaje y programación de computadoras. Detalla el análisis, desarrollo y aplicación de intervenciones basadas en acciones de aprendizaje y uso colaborativo destinadas a mejorar el aprendizaje de los estudiantes que cursan temas de fundamentos de programación en sus carreras de informática. Históricamente, la Universidad de Playa Ancha en Valparaíso, Chile, tiene una tasa de reprobación del 60 y 75% de los estudiantes. Aprender a programar computadoras es particularmente desafiante porque implica aprender y manejar múltiples conceptos, procedimientos, algoritmos y lenguajes que representan lenguajes de programación y metodologías de resolución de problemas. También, estrategias y usos y aplicación de estos conjuntos de elementos de manera integrada y la aplicación de estos elementos. Asegúrese de que se desarrollen e implementen soluciones efectivas para los 30 problemas que está tratando de resolver. Todos estos aspectos del aprendizaje son necesarios para el desarrollo y mejora de las habilidades que se pretenden en esta materia. La experiencia ha demostrado que el estudio de estas estrategias y, en consecuencia, el desarrollo y mejora de las habilidades para la resolución de problemas de programación no se logra principalmente mediante el uso de métodos tradicionales de enseñanza basados en lecciones explicativas y presentación de ejemplos. Dado que es imposible comunicar procesos mentales complejos a los estudiantes utilizando este formato, esto significa procesar las estrategias que hemos dictado para que los estudiantes puedan interiorizarlas. En

este contexto, se han desarrollado y aplicado intervenciones basadas en la integración de actividades de aprendizaje y trabajo colaborativo, que buscan mejorar y facilitar las condiciones en las que los estudiantes aprenden conceptos y procedimientos y desarrollan formas de pensamiento. Manejo de estrategias necesarias para resolver problemas de programación con el propósito de adquirir, desarrollar y mejorar la habilidad para desarrollar e implementar soluciones a problemas de procesamiento de datos. Para evaluar el resultado de la intervención, para los grupos experimental y control, se consideró el rendimiento académico de los alumnos aprobados e insatisfechos y se define para tal fin. Según esta estimación, el grupo experimental mostró una mejora significativa con respecto al grupo control, que alcanzó el 75,59% de los que aprobaron la primera ronda y el 39,84% de los que aprobaron la segunda. Por otro lado, en el caso del grupo de intervención, a pesar de los avances en el desarrollo de las actividades cooperativas, la mayoría de los grupos mostraron una mejora significativa en la calidad del trabajo cooperativo que desarrollaban, lo que se reflejó de forma gradual y progresiva. Mejora de los resultados de la evaluación individual del grupo experimental en comparación con el grupo control. Estas mejoras también se reflejan en la buena percepción de los alumnos sobre el tipo de intervención, tal y como demuestran los resultados de una encuesta de satisfacción que se les realizó al final de la intervención. (p. 13)

2.2. Bases teóricas – científicas.

2.2.1. Algoritmo

Joyanes L. (2008) Un algoritmo se puede definir como una secuencia inequívoca de pasos que proporciona una solución a un problema dado y se expresa en lenguaje natural.

Para desarrollar un algoritmo, primero debe identificar las tareas más importantes para resolver un problema y organizarlas en el orden en que deben realizarse. Un algoritmo se puede definir como un conjunto de instrucciones que representa un modelo de solución a un tipo particular de problema. O como una serie de comandos, ejecutados en secuencia, proporcionarán una solución al problema.

Para construir un programa, es conveniente desarrollar o predefinir un algoritmo.

El desarrollo de algoritmos requiere creatividad y un conocimiento profundo de las habilidades de programación. Luis Joyanes, programador profesional y autor de muchos libros sobre lógica y programación, dice: “en la ciencia de la computación y en la programación, los algoritmos son más importantes que los lenguajes de programación o las computadoras. Un lenguaje de programación es sólo un medio para expresar un algoritmo y una computadora es sólo un procesador para ejecutarlo”.

Características del algoritmo

Las principales características que deben cumplir todos los algoritmos son:

- Los algoritmos deben ser precisos e indicar el orden en que se realiza cada paso.

- Los algoritmos deben estar claramente definidos. Si ejecuta el algoritmo dos veces, debería obtener el mismo resultado cada vez.
- Los algoritmos deben ser finitos. Si sigues el algoritmo, debes terminar en un punto determinado. Es decir, debe haber un número finito de pasos.

Pasos en la metodología de resolución de problemas

El proceso de resolver un problema con una computadora da como resultado escribir un programa y ejecutarlo en la computadora. Aunque el proceso de desarrollo de software es un proceso creativo por naturaleza, puede considerarse como un conjunto general de pasos o pasos que todos los programadores deberían seguir en general.

Pasos para solucionar los problemas de su computadora:

- Análisis del problema.
- Desarrollo de algoritmos.
- Codificación.
- Editar y ejecutar.
- Verificación.
- Depuración.
- mantener.
- Documentación.

Medios de Expresión del algoritmo

Los algoritmos se pueden expresar de diversas formas, incluido el lenguaje natural, el pseudocódigo, los diagramas de flujo y los lenguajes de programación. Las descripciones en lenguaje natural tienden a ser vagas y extensas. Se pueden evitar muchas ambigüedades en el lenguaje natural usando pseudocódigo y diagramas de flujo. Estas expresiones son una forma más

estructurada de expresar algoritmos, pero son independientes de cualquier lenguaje de programación en particular. La descripción de un algoritmo se suele hacer en tres niveles.

Descripción de alto nivel. Se formula un problema, se elige un modelo matemático, se describe verbalmente un algoritmo, se pueden omitir las descripciones y se pueden omitir los detalles.

Descripción formal. El pseudocódigo se utiliza para describir una secuencia de pasos para encontrar una solución.

Implementación. Un algoritmo se expresa como algún objeto capaz de ejecutar un lenguaje de programación o comando específico. También puede incluir un teorema que demuestre la corrección de un algoritmo, un análisis de su complejidad o ambos.

2.2.2. Definición del problema

Esta fase enuncia el problema asignado y procede a analizarlo considerando los datos de entrada y salida, al respecto Pérez, I. et al (s.f) argumenta: "... trata sobre definir y acotar adecuadamente el problema a resolver, en correspondencia con una solución que se puede construir por computadora; a continuación, se procede a definir las entradas y salidas que deberá tener el algoritmo a diseñar." (p. 15). Esta fase considera la especificación de los datos de entrada, lo que consiste en asignar valores o variables a cada uno de los enunciados que se leen en el enunciado del problema, a cada una de las cuales se les asigna un identificador; posterior a este proceso se determina los datos de salida, que consiste en establecer la operación numérica, lógica o de relación, que el problema obtendrá al resolverlo, Galindo, J. y Vilaplana, J. (2009) manifiestan: "hay que realizar un análisis del problema, que puede ser laborioso, ya que a veces

el problema es difícil de comprender. Una vez que se ha aclarado el problema, se realiza un análisis minucioso de los requerimientos que ayudarán a definir el problema que queremos resolver.” (p. 13).

El proceso de definición del problema le genera los argumentos suficientes para entender de lo que se trata el problema, sus datos que considera como entrada y finalmente los datos de salida, esta información, permitirá al programador contar con los datos necesarios para resolverlo.

2.2.3. Elaboración del algoritmo

En esta fase, se establece las instrucciones por cada uno de los procesos para la construcción del algoritmo, al respecto, Pérez, I. et al (s.f), argumenta:

... aborda la definición de los elementos particulares que deberá llevar el proceso del algoritmo; esto se realiza inicialmente como una jerarquización de los pasos del algoritmo a diseñar (diseño descendente); después se trata de representar de manera más detallada cada una de las instrucciones o pasos del algoritmo, de manera parecida al pseudocódigo. Finalmente, se utiliza alguna de las herramientas de diagramación de algoritmos, para hacer una representación gráfica de él, con la finalidad de lograr un mejor entendimiento y una adecuada documentación del mismo. (p. 16)

En esta fase se utiliza una herramienta de programación con el fin de representarlo, puede ser un pseudocódigo, un diagrama de flujo o un diagrama de N-S, cualquiera de ellos, permite entender su resolución de manera práctica y de fácil comprensión, al respecto, Galindo, J. y Vilaplana, J. (2009) afirma: “A partir del enunciado debemos realizar un diseño que nos lleve a la solución deseada.”, estos procesos, nos brindan la oportunidad para realizar las correcciones

necesarias antes de probar el funcionamiento en un ordenador, utilizando un lenguaje, en la actualidad existen herramientas que permiten establecer cada uno de los procesos con el fin de comprender el algoritmo y resolver el problema.

2.2.4. Codificación del algoritmo

En esta fase, el propósito es trasladar a un lenguaje de especificación que puede ser Pseint para codificar el algoritmo y conocer los aciertos o errores que se pueden haber cometido durante su elaboración, al respecto, Pérez, I. et al (s.f) manifiesta: “... codificación del algoritmo en un lenguaje de programación específico, su compilación y ejecución, así como de la depuración de los errores que pudiese tener. Esta fase es la que construye un producto de software como tal.” (p.17)

Es pertinente, utilizar un lenguaje de programación para codificar el pseudocódigo, de manera que se puede comprobar su funcionamiento, estableciendo las correcciones lógicas o de especificación acerca de las sintaxis generadas por el programa, al respecto, Galindo, J. y Vilaplana, J. (2009) argumentan:

Implementar significa llevar a cabo. El resultado del diseño es un algoritmo, pero, para ejecutarlo deberá traducirse el diseño realizado en lenguaje algorítmico a un lenguaje de programación que el ordenador entienda. Esta etapa es sencilla, puesto que sólo se trata de una traducción bastante mecánica del diseño ya realizado para obtener el programa correspondiente, que podremos ejecutar en el ordenador.” (p.14)

2.2.5. Programación informática

La programación informática es la rama de la tecnología de la información que se encarga de desarrollar y escribir instrucciones o declaraciones que una computadora debe ejecutar para completar una tarea o resolver un problema. Se denomina proceso al conjunto de acciones que realiza un ordenador para proporcionar un resultado concreto, y al conjunto de procesos que realizan una tarea común, y por tanto programa, que forma un todo. Por ejemplo, un proceso podría ser la suma de los importes que componen una línea de factura. el otro, el cálculo del impuesto que se aplicará sobre el monto de la factura; Recuperar los detalles del cliente al que desea enviar una factura es un proceso diferente. Cuando se integran todos estos y otros procesos similares, se crea un programa de facturación. Además, si tiene un proceso para calcular la ruta y la distancia de un vehículo de una empresa de transporte, puede agregarlo a su antiguo programa. Debe estar en un programa diferente en relación con la empresa de logística de envío. De esta forma, aumentaremos nuestra productividad de ejecución y lograremos un conjunto de programas más estructurados y enfocados a resolver problemas específicos.

2.2.6. Lenguaje de programación

En informática, los lenguajes de codificación se conocen como programas diseñados para crear otros programas informáticos. El nombre es un lenguaje formal diseñado para constituir algoritmos y procesos lógicos, ejecutados por computadoras o sistemas informáticos para controlar su comportamiento físico y lógico y habilitar formas de comunicarse con los humanos.

Según Wilson (1993, p. 75), Un lenguaje de programación es un lenguaje artificial diseñado para expresar cálculos que una máquina como una

computadora puede realizar, asimismo desarrollar la lógica para crear programas informáticos, aplicaciones.

Olarte, L. (2018) Un lenguaje de programación proporciona los elementos de lenguaje necesarios que son para traducir los pasos de un pseudocódigo en formato comprensible de la máquina, lo cual esto permite desarrollar la solución del problema a un algoritmo.

Los lenguajes de codificación proporcionan una transformación de un pseudocódigo legible por humanos a instrucciones legibles por máquina, los cuales esto facilite la mejor comprensión al desarrollar un algoritmo.

Tipos de lenguaje de programación. Hay tres tipos de lenguaje de programación de las cuales estos son:

a) Lenguaje de Máquina

Es un número binario o conjunto de bits que un ordenador lee e interpreta.

El lenguaje de máquina es el único lenguaje que una computadora puede entender. Un lenguaje de bajo nivel está más cerca del lenguaje utilizado por las computadoras, y un lenguaje de alto nivel está más cerca del lenguaje humano.

b) Lenguaje de bajo nivel:

Según: (Macias, 2016) El código escrito en ensamblador es algo difícil de entender porque su estructura es cercana al lenguaje máquina, es decir, un lenguaje de bajo nivel, lo cual consiste en dos valores numéricos de las cuales estos son 1 y 0, como también lo conocen números binarios

c) Lenguaje de alto nivel:

Los lenguajes de programación de alto nivel se caracterizan por expresar algoritmos de una manera que coincide con las capacidades cognitivas

humanas en lugar de la capacidad de una máquina para ejecutarlos, entendiéndolo mejor son los lenguajes de programación para desarrollar la codificación en el ordenador.

2.2.7. Codificación y escritura de instrucciones

El proceso de codificación consiste en trasladar cada una de las líneas de instrucción de los pseudocódigos a un lenguaje de programación determinado con el fin de conocer los posibles errores en su elaboración que pueden estar vinculados a los datos, operaciones lógicas, condiciones, instrucciones diversas, etc., lo que permite al programador trabajar en cada una de las instrucciones para analizar y corregir los errores vinculados al sistema, este proceso de codificación se considera un software inicial, al respecto Castrillon, M. et al (2012) afirma:

La implementación, consiste en transformar los elementos y acciones en una secuencia ordenada de instrucciones y sentencias que conforman, dan cuerpo y definen el algoritmo, que resuelve el trabajo planteado, ... para concluir esta fase, y acometer la ejecución del algoritmo, ésta debe transcribirse en instrucciones entendibles por el ordenador, al que se denomina programa. (p. 95-96)

Los libros de programación por lo general utilizan pseudocódigos para ilustrar sus ejemplos de modo que todos los programadores que puedan acceder a él puedan entenderlo, sin necesidad de tener conocimiento detallado de la sintaxis propia del lenguaje a utilizar, los que posteriormente deben seguir los procedimientos para ejecutar y corregir los errores que pueden presentar.

Las instrucciones que se transcriben, en realidad presentan tareas detalladas cada una de ellas, las que al ser codificadas son parte del programa que

se ha ido elaborando, la respuesta de cada tarea forma una instrucción del programa.

2.2.8. Elaboración de algoritmos o programas

La elaboración de programas implica el desarrollo de cada componente de software que los programadores desarrollan a partir de un algoritmo, las misma que debe transcribirse en un lenguaje de programación para ser ejecutado por las computadoras, considerando que este equipo, “dispondrá de un repertorio de acciones, es decir, las instrucciones que puede llevar a cabo el hardware de un ordenador y en las que se tendrá que basar cualquier programa que se desea ejecutar.” (Castrillon, M. et al, 2021, p. 96)

Si se revisa con detalle la teoría de la programación se va encontrar que todos ello parte de la especificación con detalle de un algoritmo que solucione el problema, seguidamente, implementación del algoritmo en un lenguaje de programación y finalmente la depuración del programa resultante, por ello Loor, J. (2014) argumenta:

Tradicionalmente, los cursos introductorios de programación se han enfocado a la sintaxis y semántica de un lenguaje de computadora. Sin embargo, la investigación ha mostrado que el verdadero problema que tienen los estudiantes radica en “juntar las piezas”. Los programadores expertos conocen mucho más que sintaxis y semántica; saben cómo resolver una gran variedad de problemas, y cómo coordinar y aplicar varias soluciones para resolver un problema complejo. Soloway afirma que un programa no debe ser únicamente un mecanismo que dice a la computadora cómo resolver un problema, sino también una explicación que dice al programador de qué manera lo resuelve. Así, aprender a programar consiste en aprender a

construir mecanismos y explicaciones; esto trasciende el campo de la programación, pues cotidianamente es necesario hacer lo anterior para resolver problemas. (p. 3)

2.2.9. Comprobación del funcionamiento de programas

Para el proceso de funcionamiento de programas es pertinente, establecer cada una de las secuencias que conforma dicha solución, para la ejecución de dichos procesos, el programador deberá transcribir el pseudocódigo a un lenguaje de programación considerando sus sintaxis y estructuración correspondiente, posterior a este proceso se realiza la ejecución del programa, para comprobar su funcionamiento, estableciendo lineamientos concretos para realizar las correcciones del programa respectivamente, al respecto, Galindo, J. (2012) precisa: “En la etapa de pruebas se trata de probar el programa resultante con diferentes datos de entrada que reciben el nombre de juegos de pruebas. El éxito de estas pruebas dependerá en gran medida de la calidad del diseño realizado antes.” (p. 14). Asimismo, en la etapa de compilación del programa pueden surgir errores de sintaxis del lenguaje de programación en uso, y se hará necesario editarlo para corregirlos, una vez realizado el proceso, el programa se ejecutará sin problemas encontrando las soluciones que se ha pretendido al iniciar la programación utilizando los algoritmos.

2.3. Definición de términos básicos.

Algoritmo

López, J. (2007): Un algoritmo (trazar un plan) que lo resuelva por medio de pasos sucesivos y organizados en secuencia lógica.

Sánchez, J. (2008): Los algoritmos son una serie de pasos que proporciona una solución a un problema. El lenguaje algorítmico es aquel que implementa una solución teórica a un problema que indica que las operaciones se deben realizar y el orden en el que se deben realizar.

Juganaru, M. (2014): Un algoritmo es una lista finita de tareas bien definida y ordenada que permite encontrar una solución a un problema dado. Dado un estado inicial y unos datos de entrada, se llega a un estado final en el que se obtiene una decisión o solución mediante pasos secuenciales y bien definidos.

Aprendizaje de la programación

Barrera, L. (2015): La Programación es básicamente la traducción de los algoritmos en sentencias (comandos) que el computador pueda entender y ejecutar.

Zapata, C. (2006): La programación es el acto de codificar o escribir estas instrucciones en un medio llamado lenguaje de programación para crear una aplicación de software o simplemente un programa. Desde aplicaciones para el control directo de circuitos electrónicos, sistemas de nómina, hasta software para registro y compras por Internet, existe un lenguaje de programación que se adapta a casi cualquier necesidad. Incluso dispositivos móviles muy comunes hoy en día usan programas para buscar nombres en una lista de teléfonos, jugar y muchas otras opciones dependiendo del tipo de dispositivo que tengas.

Sánchez, J. (2008): La programación consiste en traducir un algoritmo a algún lenguaje informático para que pueda ser entendido por una computadora. La programación de computadoras comenzó en la década de 1950 y ha evolucionado a través de varias etapas. La programación se puede hacer usando

una variedad de métodos o técnicas. Estos métodos definen diferentes tipos de programaciones.

Era digital

Gira en torno a las nuevas tecnologías e Internet y está llevando a cabo cambios profundos y transformaciones de una sociedad que se mueve en un mundo globalizado. Estos cambios profundos suponen una verdadera revolución que nos toca vivir; a pesar de que todavía muchas personas no se hayan dado cuenta de lo que “tenemos encima”.

Lenguaje de programación

Es un idioma artificial diseñado para que se fácilmente entendible por el ser humano e interpretado por la máquina. Consta de una serie de reglas y de un conjunto de ordenes o instrucciones. Cada una de las instrucciones realiza una tarea determinada. A través de la secuencia de instrucciones se pueden indicar a la computadora que se ejecute algo o los procesos que debe seguir para solucionar un problema. (Jiménez, A. 2021)

Dispositivo de entrada – salida

Son los encargados de suministrar al ordenador los datos que debe procesar y las instrucciones necesarias para ello. Estos dispositivos incluyen entrada de datos, salida y dispositivos de almacenamiento.

Programas

Conjunto de órdenes (instrucciones y sentencias) creados y diseñados a través del razonamiento lógico y almacenamiento en ficheros de texto, respetando la sintaxis de un determinado lenguaje de programación. Estos conjuntos de órdenes se transmiten al ordenador para la realización y ejecución de tareas concretas. (Quero, E. 2003)

Proceso

Es básicamente un programa en ejecución. La información relacionada a un proceso se almacena en una tabla de procesos, que consta de una lista ligada de estructuras con información sobre cada una de las tareas existentes a realizar. En el mundo de la programación se encuentra procesos secuenciales, selectivos y repetitivos.

2.4. Formulación de Hipótesis

2.4.1. Hipótesis General

Los algoritmos influyen significativamente en el aprendizaje de la programación informática en los estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023.

2.4.2. Hipótesis Específicas

- a. El nivel de aprendizaje de la programación antes de la aplicación de los algoritmos en estudiantes del III ciclo del P.E. de tecnología informática y telecomunicaciones – filial Yanahuanca, 2023 es bajo.
- b. Las herramientas y recursos de los algoritmos para el desarrollo de la programación informática en estudiantes del III ciclo del Programa de Estudios Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023 es óptimo.
- c. El nivel de desarrollo de la programación informática después de la aplicación de los algoritmos en estudiantes del III ciclo del Programa de Estudios Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023 es alto.

2.4.3. Hipótesis nula

General:

Los algoritmos no influyen en el aprendizaje de la programación informática en los estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023.

Específicas:

- a. El nivel de aprendizaje de la programación antes de la aplicación de los algoritmos en estudiantes del III ciclo del P.E. de tecnología informática y telecomunicaciones – filial Yanahuanca, 2023 es alto.
- b. Las herramientas y recursos de los algoritmos para el desarrollo de la programación informática en estudiantes del III ciclo del Programa de Estudios Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023 no es óptimo.
- c. El nivel de desarrollo de la programación informática después de la aplicación de los algoritmos en estudiantes del III ciclo del Programa de Estudios Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023 es bajo.

2.5. Identificación de Variables.

2.5.1. Variable independiente

Algoritmos

2.5.2. Variable dependiente

Programación informática

2.6. Definición Operacionalización de variables e indicadores

2.6.1. Definición conceptual

Variable independiente:

Algoritmos: “Un algoritmo constituye una lista bien definida, ordenada y finita de operaciones, que permite encontrar la solución a un problema determinado. Dado un estado inicial y una entrada, es a través de pasos sucesivos y bien definidos que se llega a un estado final, en el que se obtiene una solución” (Juganuru, 2014, p. 2)

Variable dependiente:

Programación informática: “Consiste en escribir la lista de instrucciones que un computador debe ejecutar la que se denomina programa, los que piden algunos datos de entrada de manera que pueda devolver resultados al final del proceso, previa verificación de cada línea de instrucción considerada para la resolución de un problema” (Llanos, 2010, p. 9)

2.6.2. Definición Operacional

El siguiente cuadro grafica las variables, dimensiones y los indicadores correspondientes.

Tabla 1. *Variable independiente: Algoritmos*

Variable	Dimensiones	Indicadores	Ítems	Instrumento
Variable Independiente	Definición del problema	✓ Utiliza las estructuras de control de flujo.	03	Prueba pedagógica
Algoritmos	Elaboración del algoritmo	✓ Diseño de algoritmos, entrada, proceso y salida de datos.	03	
	Codificación del algoritmo	✓ Utiliza los tipos de datos.	04	

Tabla 2. *Variable dependiente: Programación informática*

Variable	Dimensiones	Indicadores	Ítems	Instrumento
Variable Dependiente Programación informática.	Codificación y escritura de instrucciones.	✓ Identifica y usa las variables correctas para el diseño de algoritmo de programación.	02 sesiones de aprendizaje	Ficha de observación Guías de práctica
	Elaboración de aplicaciones o programas.	✓ Utiliza el orden de control de flujo de selección para el diseño de algoritmos.	02 sesiones de aprendizaje	
	Comprobación del funcionamiento	✓ Utiliza los Operadores matemáticos para la programación.	02 sesiones de aprendizaje	
		✓ Utiliza los tipos de datos para programar.	02 sesiones de aprendizaje	

CAPÍTULO III

METODOLOGÍA Y TÉCNICAS DE INVESTIGACIÓN

3.1. Tipo de investigación

El presente trabajo de investigación se caracteriza por ser una investigación de tipo aplicada, en el nivel explicativo y enfoque cuantitativo. (Hernández et al, 2018).

3.2. Nivel de investigación

Nivel explicativo.

3.3. Métodos de investigación

Científico

Según **Tamayo, M. (2012)**, “El método científico es un conjunto de procedimientos por los cuales se plantean los problemas científicos y se ponen a prueba las hipótesis y los instrumentos de trabajo investigativo” (p.30).

Experimental

Según **Tamayo, M. (2012)** la investigación experimental es aquella que tiene como el propósito de determinar la causalidad. Para ello, uno o más grupos

se exponen a estímulos experimentales y el comportamiento resultante se compara con el comportamiento de los otros grupos.

Asimismo, Consiste en organizar las condiciones para el estudio de posibles relaciones causales mediante la exposición de uno o más grupos experimentales según un plan preestablecido.

3.4. Diseño de investigación.

Kerlinger (2002) sostiene que generalmente se llama diseño de investigación al plan y a la estructura de un estudio.

El diseño de la presente investigación es el Pre-experimental.

Pre-experimental con un solo grupo, formulado como un plan para lograr los objetivos de la investigación. Esta es una secuencia de todos los pasos para realizar un experimento. Aquí, la variable independiente (X) es la causa y la variable dependiente (Y) es el efecto.

Su representación es:

GE: O ₁ X O ₂

Donde:

GE: Grupo experimental

O1 = Pre test

X = Aplicación de la variable experimental

O2 = Post test

3.5. Población y muestra

3.5.1. Población

Estuvo conformado por los estudiantes del I al IX ciclo de la especialidad Tecnología Informática y Telecomunicaciones de la Universidad Nacional Daniel Alcides Carrión, filial Yanahuanca, que suman 75 estudiantes.

Tabla 3. *Población de estudiantes de la filial*

Nº	DESCRIPCIÓN	CANTIDAD
1	I Semestre	22
2	III Semestre	16
3	V Semestre	12
4	VII Semestre	6
5	X Semestre	19
TOTAL		75

3.5.2. Muestra

Para el cálculo de la muestra se aplicó el muestreo no probabilístico basado en el acceso, se ha determinado la muestra completa, intencional, no estadística con grupo completo conformado por 16 estudiantes del III semestre para el grupo experimental y control respectivamente, a quienes se les aplicó un pre test y post test para medir el nivel de aprendizaje de los algoritmos, Iborra et al (2010)

3.6. Técnicas e instrumentos de recolección de datos

3.6.1. Técnicas

Encuesta: Hernández et al (2006, p. 310) Definir la encuesta como la herramienta de adquisición de datos más utilizada. La encuesta consiste en una serie de preguntas sobre una o más variables que se miden. Dicho cuestionario debe ser elaborado con preguntas dirigidas a la información que se espera obtener.

Observación directa: consiste en obtener datos en el mismo lugar de la aplicación, observan diversos procesos y trabajos desarrollados por los estudiantes. Este método de recolección de datos consiste en un registro sistemático, válido y confiable del comportamiento y las circunstancias observadas. Ander-Egg (1996), se refiere a la observación como “Uno de los métodos de recopilación y revisión de datos sobre realidades socioculturales, comunidades o grupos sociales específicos.” (p.197).

3.6.2. Instrumentos

Prueba pedagógica: Instrumento con una cantidad de interrogantes, que permite recoger información, Según Hernández Sampieri (1997), nos menciona que el cuestionario es tal vez el más utilizado para la recolección de datos; este consiste en un conjunto de preguntas respecto a una o más variables a medir.

Ficha de observación: Instrumento de recolección de datos, que permite el análisis minucioso de una situación determinada, o el comportamiento y características de una persona.

Guías de práctica: Son documentos que permiten comprobar el aprendizaje de una fase de programación aplicando los procedimientos para su desarrollo respectivo.

3.7. Selección, validación y confiabilidad de los instrumentos de investigación

3.7.1. Selección de instrumentos

Se ha procedido seleccionar dos pruebas, la primera antes del desarrollo de sesiones de aprendizaje y la otra después del desarrollo de la investigación, para la recopilación de información, estrictamente con cada una de las variables de la presente investigación.

3.7.2. Validación de instrumentos

Para realizar este proceso se ha utilizado el Alpha de Cronbach

3.7.3. Confiabilidad de instrumentos

Se ha procedido a dar confiabilidad a cada instrumento en relación a cada variable mediante el Alpha de Cronbach, realizando una aplicación de prueba piloto para establecer la validez de contenido, constructo y de criterio, posterior a ello se ha procedido a utilizar SPSS para realizar los procesos de fiabilidad de los instrumentos a ser aplicados, por lo ejecutado se ha tenido los siguientes resultados:

Tabla 4. *Resultados obtenidos en la prueba pedagógica*

Alum Item	1	2	3	4	5	6	7	8	9	10	Σ
A1	0	0	0	0	0	0	2	0	0	0	02
A2	2	2	2	2	2	0	2	2	0	0	14
A3	2	2	2	0	2	0	0	2	0	2	12
A4	2	2	2	2	2	0	2	2	0	2	16
A5	0	0	0	0	0	0	2	0	0	2	04
A6	2	2	2	2	2	2	2	2	2	2	20
A7	0	0	0	2	0	2	2	0	2	2	10
A8	0	0	0	2	0	2	0	0	2	2	08
A9	2	0	2	0	2	2	0	2	2	0	12
A10	2	2	2	2	2	0	2	2	0	2	16
A11	0	0	0	2	0	0	2	0	0	0	04
A12	0	0	0	2	0	0	0	0	0	2	04
A13	2	0	2	2	2	0	2	2	0	2	14
A14	2	2	2	2	2	0	2	2	0	2	16

Fuente: Proceso realizado por los autores

Tabla 5. *Resultados de prueba pedagógica*

Estadísticos de fiabilidad	
Alfa de Cronbach	N de elementos
.773	14

Fuente: Proceso realizado por los autores

Los resultados muestran que el instrumento de la prueba pedagógica al ser procesado con SPSS 25 se ha obtenido 0.773 lo que indica que posee alta fiabilidad por lo que se aplica para el pre y post test, además el desarrollo de las sesiones de aprendizaje para comprobar lo propuesto en las respectivas hipótesis de investigación.

3.8. Técnicas de procesamiento y análisis de datos

3.8.1. Procesamiento manual

Se ha utilizado el conteo para determinar la cantidad de respuestas encontradas en función a las preguntas realizadas, teniendo en cuenta las respuestas brindadas en relación a las dimensiones e indicadores propuestos.

3.8.2. Procesamiento electrónico

Se ha utilizado el paquete estadístico SPSS y Microsoft Excel, para encontrar los valores respectivos en estricta relación con la investigación. Para la prueba de hipótesis se ha utilizado la prueba T de Student para comprobar la pertinencia de trabajo académico al manipular la variable independiente a partir de la dependiente con sus respectivas dimensiones.

3.9. Tratamiento estadístico

El análisis estadístico de esta investigación se realizó mediante tablas y gráficos con distribución de frecuencias, por cada una de las variables y

dimensiones propuestas, los datos obtenidos que se han obtenidos al aplicar los instrumentos por cada variable han permitido representar a través de las matrices los resultados por variable y dimensión. Para determinar la confiabilidad de los instrumentos se ha aplicado una prueba piloto y se ha procedido aplicar la prueba Alpha de Cronbach. Para realizar la prueba de hipótesis, se ha utilizado la prueba T de Student para comprobar la relación de medias, por tratarse de datos cuantitativos.

3.10. Orientación ética filosófica y epistémica

Se procede a detallar cada una de las consideraciones que se van a tener en cuenta durante el proceso de investigación y su posterior informe de tesis:

- Respeto a la autoría de las fuentes de información, referenciando adecuadamente los artículos, libros o investigaciones utilizadas como consulta o fuente de información utilizada en la investigación.
- Parfrasear la información obtenida como producto de consulta de fuentes de información relacionado con las variables y dimensiones de la investigación.
- Respetar la confidencialidad del desarrollo académico de cada uno de los estudiantes que formaron parte de la investigación, sin difundir por ningún medio sus aciertos o errores durante el desarrollo del trabajo académico.

CAPÍTULO IV

RESULTADOS Y DISCUSIÓN

4.1. Descripción del trabajo de campo.

Para realizar los procesos de investigación relacionados con el presente estudio, se ha aplicado dos pruebas, antes y después de la realización de los procesos de trabajo académico, las interrogantes planteadas en cada una de las pruebas están relacionadas con la variable independiente: algoritmos con sus respectivas dimensiones definición del problema con su indicador utiliza las estructuras de control de flujo, para lo cual se ha planteado interrogantes relacionados con el control de flujo de un algoritmo: estructuras secuenciales, condicionales y repetitivas; la elaboración del algoritmo con su indicador diseño de algoritmos de entrada, proceso y salida de datos; en la que se ha trabajado con la representación de las instrucciones y determinación de cada tarea que va realizar el algoritmo; finalmente la dimensión: codificación del algoritmo con su indicador utiliza los tipos de datos, en la que se ha enfatizado el uso de la aplicación Pseint para la representación del pseudocódigo y comprobar su funcionamiento, reconociendo y corrigiendo los errores de sintaxis y de estructura

al codificarlos; la diversidad de procesos que se ha desarrollado incluyen las 08 sesiones de aprendizaje que se han desarrollado, las misma que están adjuntas en los anexos con los resultados respectivos relacionados con los logros de los estudiantes conformantes de la muestra de investigación.

4.2. Presentación, análisis e interpretación de resultados.

Tablas, gráficas e interpretación de datos

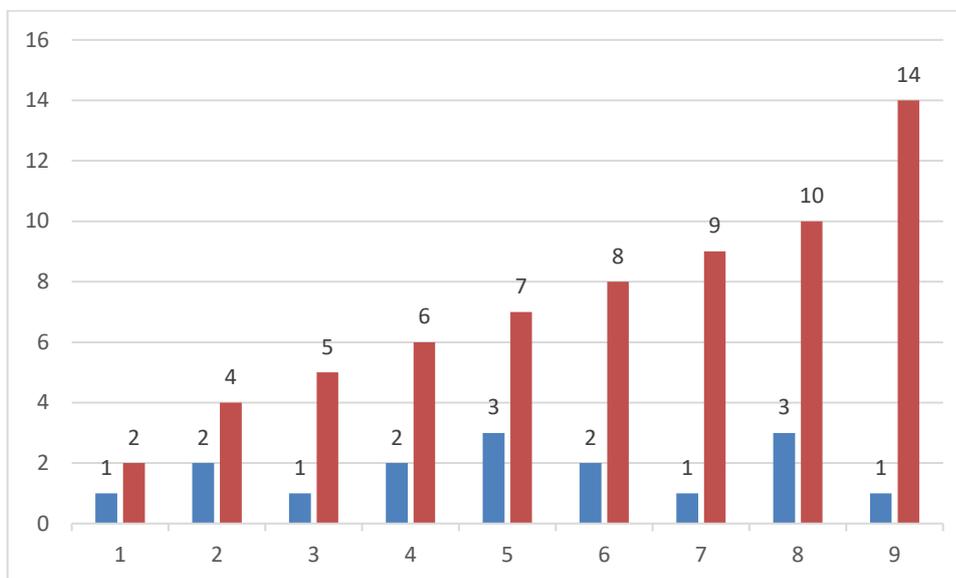
Prueba pedagógica de entrada

Tabla 6. *Notas obtenidas en la prueba de entrada*

Semestre de estudios	Cantidad de estudiantes	Nota obtenida	Porcentaje
	1	02	6.25
	2	04	12.5
	1	05	6.25
	2	06	12.5
III	3	07	18.75
	2	08	12.5
	1	09	6.25
	3	10	18.75
	1	14	6.25

Fuente: Matriz elaborado por los autores

Figura 1. *Resultados de la prueba de entrada – III ciclo*



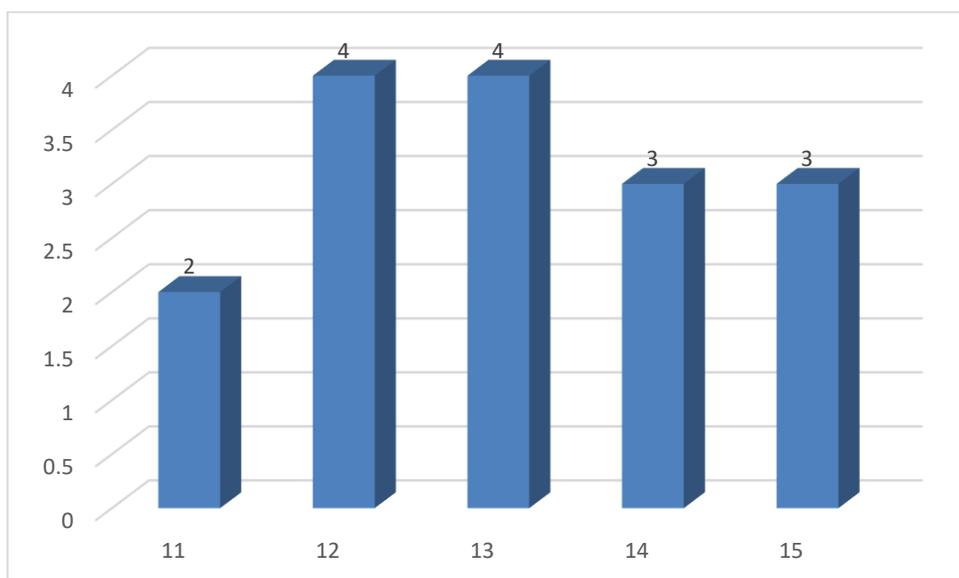
Interpretación: Al visualizar la matriz y la imagen respectiva se observa que un 6.25% (1) estudiante ha obtenido la nota de 02, 12.5% (2) la nota de 04; 6.25% (1) la nota de 05; 12.5% (2) la nota 06; 18.75% (3) la nota 07; 12.75% (2) la nota de 08; 6.25% (1) la nota de 09; 18.75% (3) la nota de 10 y 6.25% (1) la nota de 14; lo que indica que la mayoría de los estudiantes se encuentran en inicio del desarrollo de la competencia prevista por lo que es pertinente realizar las actividades pedagógicas implementando los trabajos académicos con el uso intenso de herramientas

Tabla 7. *Notas obtenidas en las sesiones de aprendizaje*

Semestre de estudios	Cantidad de estudiantes	Nota obtenida	Porcentaje
III	02	11	12.5
	04	12	25
	04	13	25
	03	14	18.75
	03	15	18.75

Fuente: Matriz elaborado por los autores

Figura 2. *Resultados de las sesiones de aprendizaje*



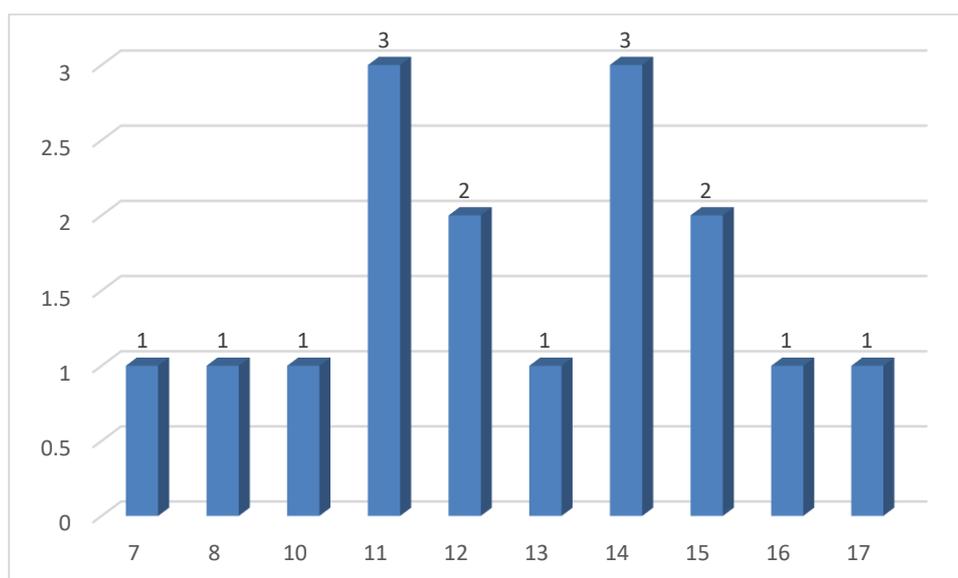
Interpretación: Al observar los resultados de la tabla 06, las notas obtenidas durante el desarrollo de sesiones de aprendizaje, un 12.5% (02) han obtenido la nota 11; un 25% (04) 12, un 25% (04) 13, un 18.75% (03) 14 y un 18.75% (03) 15; lo que demuestra que los estudiantes han venido incorporando los algoritmos para resolver la diversidad de problemas planteados en sus fases respectivas, utilizando la aplicación del Pseint como herramienta para representar sus pseudocódigos y ejecutarlo respectivamente.

Tabla 8. *Notas obtenidas en la posprueba*

Semestre de estudios	Nota obtenida	Cantidad de estudiantes	Porcentaje
III	07	01	6.25
	08	01	6.25
	10	01	6.25
	11	03	18.75
	12	02	12.5
	13	01	6.25
	14	03	18.75
	15	02	12.5
	16	01	6.25
	17	01	6.25

Fuente: Matriz elaborado por los autores

Figura 3. *Resultados de la prueba de salida*



Interpretación: Los resultados de la tabla y gráfico precedente nos muestra que un 6.25% (01) han obtenido 07 de nota; un 6.25% (01) 08; un 6.25% (01) 10, un 18.75% (03) 11, un 12.50% (02) 12, un 6.25% (01) 13, un 18.75%

(03) 14, un 12.50% (02) 15, un 6.25% (01) 16 y un 6.25% (01) 17; los resultados muestran que los estudiantes se encuentran en nivel de proceso y logro en su desarrollo académico en el área de estudio.

4.3. Prueba de Hipótesis

Para la realización de este proceso se ha aplicado una prueba de entrada y una prueba de salida considerando las dimensiones e indicadores al elaborar la prueba pedagógica, con los mencionados resultados se ha procedido a comprobar la prueba de hipótesis de acuerdo a los siguientes procedimientos:

Para la hipótesis general

Paso 1. Planteamiento de hipótesis nula (h_0) y la hipótesis alterna (h_1)

Hipótesis nula (h_0)

Los algoritmos no influyen en el aprendizaje de la programación informática en los estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023.

Hipótesis alterna (h_1)

Los algoritmos influyen significativamente en el aprendizaje de la programación informática en los estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023.

Paso 2. Selección del nivel de significancia

$$\alpha = 0,05$$

Paso 3. Escoger el estadístico de prueba

Con el propósito de obtener los resultados finales para aplicar la prueba T de Student se ha procedido a comparar los resultados del pre y postest:

Tabla 9. *Diferencia de notas de pre test y post test de la muestra de estudio*

pre test	Postest	Diferencia	
		D-A	D ²
09	14	05	25
06	10	04	16
07	15	08	64
14	14	00	00
02	11	09	81
04	08	04	16
10	12	02	04
14	11	-03	09
05	12	07	49
04	13	09	81
08	07	-01	01
09	15	06	36
08	16	08	64
14	17	03	09
07	14	07	49
08	11	03	09
		71	5041

Fuente: Matriz elaborada por los autores

Tabla 10. *Estadísticas de muestras emparejadas*

		Media	N	Desv. Desviación	Desv. Error promedio
Par 1	PRUEBA PEDAGOGICA DE ENTRADA	8,06	16	3,623	,906
	PRUEBA PEDAGOGICA DE SALIDA	12,50	16	2,781	,695

Fuente: Matriz elaborada por los autores

Interpretación: En la tabla precedente se observa que existe una diferencia significativa en las medias obtenidas en ambas pruebas, en la preprueba se ha obtenido 08.06 de promedio y en la posprueba se ha obtenido 12.50, encontrando una diferencia de 4.44 demostrando que hubo incremento significativo de saberes relacionados con la aplicación de algoritmos para resolver problemas en programación informática.

Tabla 11. *Correlaciones de muestras emparejadas*

		N	Correlación	Sig.
Par 1	PRUEBA PEDAGOGICA DE ENTRADA & PRUEBA PEDAGOGICA DE SALIDA	16	,380	,146

Interpretación: En la tabla precedente se comprueba que el nivel de correlación es 0.380 demostrando relación débil entre las pruebas administradas a los estudiantes de la muestra, asimismo el valor de significancia es $0.146 > 0.05$.

Tabla 12. *Prueba T de Student para muestras emparejadas*

Par		Media	Diferencias emparejadas			t	gl	Sig. (bilateral)
			Desv. Desviación	Desv. Error promedio	95% de intervalo de confianza de la diferencia Inferior Superior			
1	PRUEBA PEDAGOGICA DE ENTRADA - PRUEBA PEDAGOGICA DE SALIDA	-4,438	3,633	,908	-6,373 -2,502	-4,886	15	,000

Regla de decisión:

Si alfa (Sig) $> 0,05$; Se acepta la hipótesis nula

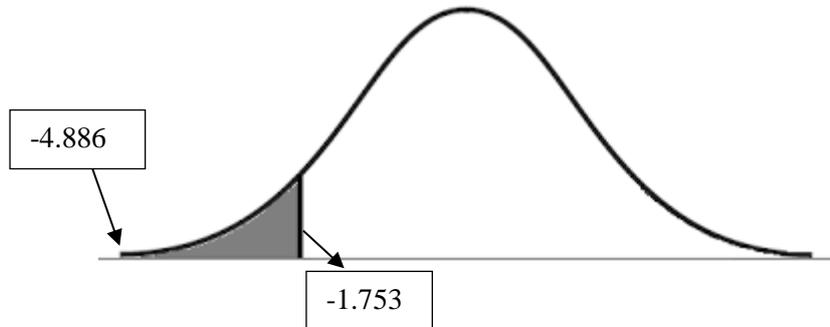
Si alfa (Sig) $< 0,05$; Se rechaza la hipótesis nula

Según se observa en la tabla 12, el valor de significancia obtenido entre los resultados de pre y posprueba es menor a 0,05 (Sig. = 0,000 $< 0,05$), por lo que se rechaza la hipótesis nula y se acepta la hipótesis alterna, asimismo el valor de la prueba t de Student con 15 grados de libertad es -4.886 y el valor de la tabla es -1.753 , entonces rechazamos la hipótesis nula (H_0) y aceptamos la hipótesis alterna (H_1); es decir, **“Los algoritmos influyen significativamente en el aprendizaje de la programación informática en los estudiantes del III ciclo**

**del Programa de Estudios de Tecnología Informática y Telecomunicaciones
– filial Yanahuanca, 2023”**

Paso 4. Diagrama

Figura 4. *Prueba t con una sola cola*



Fuente: Campana de Gauss elaborado por los autores

Paso 5. Toma de decisión

Según los resultados obtenidos a un nivel de confianza del 95%, se verifica que: el incremento de notas en los estudiantes de la muestra es significativo, en la preprueba es 8.06 y en la posprueba es 12.50, demostrando un incremento de 4.44 puntos, asimismo el valor de significancia es 0,000 menor a 0.05 por lo que se acepta la hipótesis de investigación y se rechaza la hipótesis nula, además el valor de t de student en la tabla de valores con nivel de significancia de 0.05 es -1.753 y el valor hallado fue -4.886 rechazando la hipótesis nula y validando la hipótesis de investigación.

4.4. Discusión de resultados

El trabajo de investigación ha permitido demostrar que el uso intensivo de los algoritmos con el uso de la aplicación Pseint en los estudiantes del III semestre del Programa de Estudios Tecnología Informática y Telecomunicaciones de la filial Yanahuanca, es significativo por el valor obtenido de -4.886 que demuestra que **“Los algoritmos influyen significativamente en el aprendizaje de la**

programación informática en los estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023”, lo que ha sido demostrado con evidencias concretas que hubo incremento considerable en las promedios del pre y postest en 4.44 lo que demuestra que el trabajo con los recursos y herramientas del pseint como aplicación que ha permitido desarrollar habilidades para la resolución de problemas desde la definición, elaboración y prueba del algoritmo como respuesta a la situación problemática planteada.

En lo referido a la investigación La programación de las computadoras con micromundos pro y los procesos cognitivos básicos de los alumnos del vi ciclo de la institución educativa José Carlos Mariátegui Lachira de Michivilca – Tapuc 2017 (Alvarado, M. 2018) se concluye que existe una relación importante y de gran relevancia entre la programación con Micromundos Pro y los procesos cognitivos básicos de los estudiantes por el valor de la correlación de Pearson en 0.85, coincidiendo con lo obtenido en nuestra investigación.

En referencia al estudio de Arias, W. & Timoteo W. (2019) “Programación de aplicativos para equipos móviles en el mejoramiento del aprendizaje en los estudiantes del IV grado de educación secundaria del área de educación para el trabajo de la institución educativa san miguel, distrito de Acobamba provincia de tarma – región Junín”; se concluye que la programación de aplicaciones para ordenadores o dispositivos móviles se desarrolló como una aplicación AppInventor, lo que permitió desarrollar una capacitación básica para demostrar cómo construirlo en la práctica. La investigación servirá para desarrollar nuevas estrategias de aprendizaje para el desarrollo de nuevas tecnologías, y al mismo tiempo, sirva como una oportunidad para impulsar el

desarrollo y desarrollo. mediante la corrección de errores que se produzcan durante el desarrollo del mismo, donde los resultados obtenidos demuestran que el manejo de los algoritmos permite desarrollar o fortalecer las capacidades de programación de computadoras a través del pseint como herramienta.

En relación al estudio de Lagos, M. (2020). “Método de aprendizaje cooperativo y el aprendizaje de algoritmos de programación en estudiantes de la universidad nacional de Ayacucho”. Cuyas conclusiones se relacionan a que hubo diferencias significativas en las notas. Poner a prueba los conocimientos del grupo experimental con respecto al grupo de control. además, En cuanto a la definición de indicadores y el uso adecuado de las variables, la selección de la estructura de control de flujo en el desarrollo de algoritmos: prueba posterior No hubo diferencia significativa entre el grupo experimental y el grupo control; se encuentra una diferencia significativa con respecto al uso adecuado, la otra parte Estructuras iterativas de control de flujo y combinaciones apropiadas; selección y estructura de control de flujo iterativo del desarrollo posterior al algoritmo como prueba del grupo experimental en comparación con el grupo de control.

En referencia a la investigación de Quispe N. (2015) “Relación del pensamiento lógico y el pensamiento creativo en el aprendizaje de la lógica de programación en los estudiantes de computación e informática del IESTP Manuel Antonio Hierro Pozo de Ayacucho durante el periodo lectivo 2013”. Concluye que, el test de pensamiento creativo basado en las 15 competencias/habilidades de las personas creativas propuesto por Ponty se categoriza en competencia psicológica, competencia conductual y competencia técnica. Pruebas de entrenamiento de lógica de programación en las etapas de aprendizaje tales como conocimiento conceptual, conocimiento procedimental y conocimiento

conductual. A través del procesamiento estadístico y los resultados, se puede concluir que el pensamiento lógico y el pensamiento creativo no están relacionados con el aprendizaje lógico de programación, no existe una relación significativa entre el pensamiento lógico y el aprendizaje lógico de programación, y tampoco existe una relación significativa entre el aprendizaje lógico de programación. Pensar e investigar la lógica de la programación entre estudiantes de informática de instituciones de educación superior técnica nacional.

En relación al estudio de Pérez, H. (2017) “Uso de Scratch como herramienta para el desarrollo del pensamiento computacional en programación y de la carrera de Informática de la Universidad Central del Ecuador” cuyas conclusiones son: Los resultados obtenidos en el grupo experimental y el grupo de control se ordenan por género y se comparan en relación con el desarrollo de la capacidad de pensamiento computacional para establecer la diferencia en el aprendizaje. Comparar los resultados obtenidos en el grupo experimental y el grupo control formado por la dimensión de pensamiento computacional para establecer el nivel de desarrollo de cada uno. Durante la fase de recolección de datos se utilizó un cuestionario de 30 preguntas sobre pensamiento computacional, el cual fue analizado mediante estadística descriptiva e inferencial mediante cuestionarios pre-test y post-test según el momento seguido de triangulación de los resultados. resultados similares. inspección. Entre los hallazgos más importantes encontramos diferencias en el desarrollo de habilidades de pensamiento computacional en hombres y mujeres, y que Scratch promueve el reconocimiento de patrones en los estudiantes.

CONCLUSIONES

PRIMERA: Los resultados obtenidos demuestran que la influencia de los algoritmos influyen positivamente en el aprendizaje de la programación informática en estudiantes del III ciclo del Programa de Estudios Tecnología Informática y Telecomunicaciones, considerando que se ha obtenido un promedio de 08.06 en la pre prueba y de 12.50 en la posprueba, con una diferencia de 4.44 (Tabla 9), al probar la hipótesis con la prueba t de Student se ha obtenido -4.886 frente al valor de la tabla de -1.753 con 15 grados de libertad, además el valor de significancia $0,000 < 0,05$ (tabla 11) rechazando la hipótesis nula y aceptando la hipótesis de investigación.

SEGUNDA: Se ha demostrado que la pre prueba aplicada a los estudiantes de la muestra ha alcanzado un promedio de 08.06, donde la máxima nota es 14 y la mínima es 02, demostrando que 13 estudiantes han obtenido notas desaproboratorias en la escala vigesimal entre 02 y 10, solamente 03 obtuvieron notas aprobatorias, demostrando que era necesario la aplicación de la diversidad de procesos para elaborar algoritmos para resolver problemas mediante las computadoras, enfatizando la definición de problemas, elaboración y prueba del pseudocódigo utilizando pseint.

TERCERA: Se han desarrollado 08 sesiones de aprendizaje con los estudiantes pertenecientes al III ciclo del programa de estudios respectivamente, empleando fundamentalmente herramientas de pseint con sus respectivos procedimientos, logrando una participación activa y efectivas de los alumnos, aplicando las fases de resolución de problemas informáticos, en cada sesión resolviendo guías de práctica calificada relacionadas con el tema en desarrollo, por cuyos procesos se ha obtenido el promedio de 13.06 que

indica que los estudiantes se encuentran en el nivel de proceso en sus aprendizajes en desarrollo.

CUARTA: Se ha alcanzado un nivel de desarrollo significativo en los estudiantes de la muestra de investigación con una diferencia de medias equivalente a 4.44 entre el pre y postest, asimismo los desempeños propuestos para la investigación se han desarrollado validando la investigación.

RECOMENDACIONES

PRIMERA: Fortalecer las actividades prácticas de los estudiantes relacionados con el uso intenso de algoritmos para la resolución de problemas, donde los estudiantes deben enfatizar las operaciones lógicas y relaciones concretas con el uso de datos respectivamente para representar pseudocódigos para comprobar su funcionamiento utilizando un lenguaje de programación.

SEGUNDA: Proponer que el aprendizaje de programación esté vinculado estrictamente al uso de algoritmos de manera detallada, estableciendo niveles de conocimiento relacionados con las fases de resolución de problemas informáticos, buscando la coherencia lógica de las instrucciones y los resultados que deben poseer.

REFERENCIAS BIBLIOGRÁFICAS

- Galindo, J. y Vilaplana, J. (2009). *Introducción a la Programación*. UOC. Universidad Oberta de Catalunya
- Jiménez, A. y Pérez, F. (2021). *Programación. Desarrollo de aplicaciones multimedia. Desarrollo de aplicaciones Web*. Editorial Paraninfo. Madrid – España.
- Llanos, D. (2007). *Fundamentos de informática y programación en C*. Editorial Paraninfo. Madrid – España.
- Pérez, I. y Moreno, S. (2001). *Estudio sobre la problemática en los enunciados de los problemas de programación*. Instituto de ciencias básicas e ingeniería. Universidad autónoma del Estado de Hidalgo. Recuperado de:
https://www.uaeh.edu.mx/investigacion/productos/4786/estudio_sobre_la_problematika_en_los_enunciados_de_los.pdf
- Quero, E. (2003). *Sistemas operativos y Lenguajes de Programación*. Editorial Paraninfo. Madrid – España.
- Gutiérrez, J., Gutiérrez, C. y Gutiérrez, J. (2018). *Estrategias metodológicas de enseñanza aprendizaje con un enfoque lúdico*. Revista de Educación y Desarrollo. México
- Flores, J. (2018). *Algoritmos, aplicaciones y Big data, nuevos paradigmas en el proceso de comunicación y de enseñanza-aprendizaje del periodismo de datos*. Revista de comunicación. México
- Snyder (2000). *Pensamiento algorítmico*. Universidad La Habana. Cuba
- Guinness, H. (2017). *Algoritmo un conjunto prescrito de instrucciones*

- Arias, W. & Timoteo W. (2019). *Programación de aplicativos para equipos móviles*
- Lagos, M. (2020). *Método de aprendizaje cooperativo y el aprendizaje de algoritmos de programación*
- Quispe N. (2015) “Relación del pensamiento lógico y el pensamiento creativo en el aprendizaje de la lógica de programación”.
- Pérez, H. (2017) “Uso de Scratch como herramienta para el desarrollo del pensamiento computacional en programación”
- Caneo, O. (2002) “Las practicas cooperativas como mejora del aprendizaje en programación de computadoras”
- Joyanes L. (2008). *Fundamentos de la programación*. Edit. McHill. México
- Olarte, L. (2018). *Lenguaje de programación*.
- López, J. (2007) Algoritmo, secuencia lógica.
- Sánchez, J. (2008) algoritmos una serie de pasos que permiten obtener la solución a un problema.
- Juganaru, M. (2014) Algoritmo una lista bien definida, ordenada y finita de operaciones
- Barrera, L. (2015) Programación traducción de algoritmos.
- Zapata, C. (2006) La programación acción de codificar.
- Sánchez, J. (2008) Algoritmos en un lenguaje de programación
- Hernández, Batista (2014) Metodología de la investigación Ed14. MacGraw &Hill. México

ANEXOS



Universidad Nacional Daniel Alcides Carrión

Facultad de Ciencias de la Educación

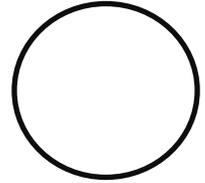
ESCUELA DE FORMACIÓN PROFESIONAL DE EDUCACIÓN SECUNDARIA

PROGRAMA DE ESTUDIOS: TECNOLOGÍA INFORMÁTICA Y TELECOMUNICACIONES

Evaluación de entrada

Apellidos y Nombres:Nota:

Semestre: Fecha:/...../.....



1. Evaluar la expresión aritmética:

$$m = \frac{a + b + c + d + e}{5}$$

2. Hallar el resultado del siguiente ejercicio:

$$e = a^7 + (a + b^2)^4 .$$

3. Muestra en pantalla el resultado de las cuatro operaciones básicas (Suma, resta, multiplicación, división) de dos números.

4. Ingresar tres notas del estudiante y hallar el promedio de sus notas.

5. Hallar la siguiente expresión

$$\boxed{5 \times \left(\frac{3 + 4}{2} \right)}$$

6. Rony desea saber si puede entrar a una discoteca. Determinar si es mayor o menor de edad.

7. Determinar si un número es positivo o negativo.

8. Si los productos comprados son menores a 3 pagar en efectivo, caso contrario pagar con tarjeta.



Universidad Nacional Daniel Alcides Carrión

Facultad de Ciencias de la Educación

ESCUELA DE FORMACIÓN PROFESIONAL DE EDUCACIÓN SECUNDARIA
PROGRAMA DE ESTUDIOS: TECNOLOGÍA INFORMÁTICA Y TELECOMUNICACIONES

9. Si Juan compra sus productos el lunes o viernes tiene un descuento de 20%.

10. Comprobar si es verdadero o falso los problemas planteados:

$$R = (10 > 4) \text{ y } (2 = 1)$$

$$P = (8 < 4) \text{ o } (7 > 4)$$

$$Q = \sim (10 > 4)$$



Universidad Nacional Daniel Alcides Carrión

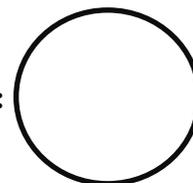
Facultad de Ciencias de la Educación

ESCUELA DE FORMACIÓN PROFESIONAL DE EDUCACIÓN SECUNDARIA
PROGRAMA DE ESTUDIOS: TECNOLOGÍA INFORMÁTICA Y TELECOMUNICACIONES

Evaluación de salida

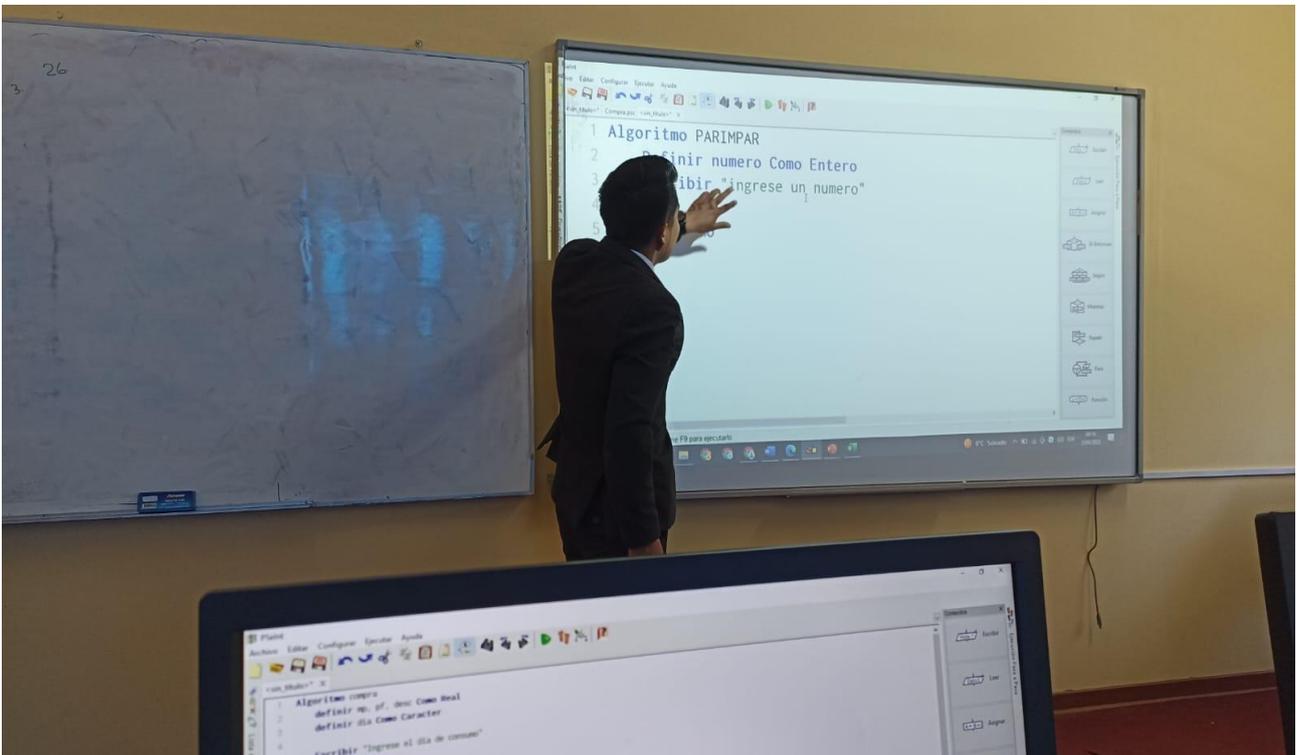
Apellidos y Nombres:.....Nota:

Semestre: Fecha:/...../.....



- ¿Cuál es el siguiente paso después de plantear el algoritmo y el pseudocódigo?**
 - Usar código fuente
 - Escribir en código de lenguaje
 - Escribirlo en lenguaje de programación
 - Representarlo mediante diagrama de flujo
- Como se llama el espacio donde es asignado un dato y que puede cambiar su valor.**
 - Constante
 - Declaración
 - Variable
 - Contador
- Definen una lógica estructurada en un algoritmo.**
 - Identificadores y variables cíclicos
 - Leer a=0, escribir a=entonces
 - Si entonces, mientras repetir
 - Ninguna de las anteriores
- ¿Qué función cumple el bucle?**
 - Permiten ejecutar un conjunto de instrucciones varias veces.
 - Permite imprimir datos de tipos numéricos muchas veces.
 - Permite evaluar las expresiones aritméticas, ejecuta los códigos las veces que se requiera.
 - Cumplen distintas funciones como reemplazar códigos varias veces en el programa.
 - N.A.
- La estructura si entonces fin si pertenecen a:**
 - Estructuras de interacción
 - Estructuras de repetición
 - Estructuras condicionales
 - Estructuras de control
- Ingresar los Nombres y Apellidos de un estudiante calcular su promedio de 3 notas, determinar si el estudiante aprueba o desaprueba
- La empresa "OLIVAS" por apertura del negocio está ofreciendo un descuento adicional a cualquier producto de compra, si el producto es mayor a tres, tiene descuento de 20%, si el producto es igual a uno no tiene descuento.
- Realizar un programa donde se ingresa los costos de un producto y el valor total de los productos. El programa se debe realizar indefinidamente, el usuario es quien decide cuando acaba.
- Hallar el perímetro y el área de las figuras geométricas (cuadrado, rectángulo y triangulo). El usuario debe elegir una figura geométrica, posteriormente a ello debe elegir que desea hallar.
- Mostrar la tabla de multiplicación desde el 1 al 10, para un número leído desde teclado.

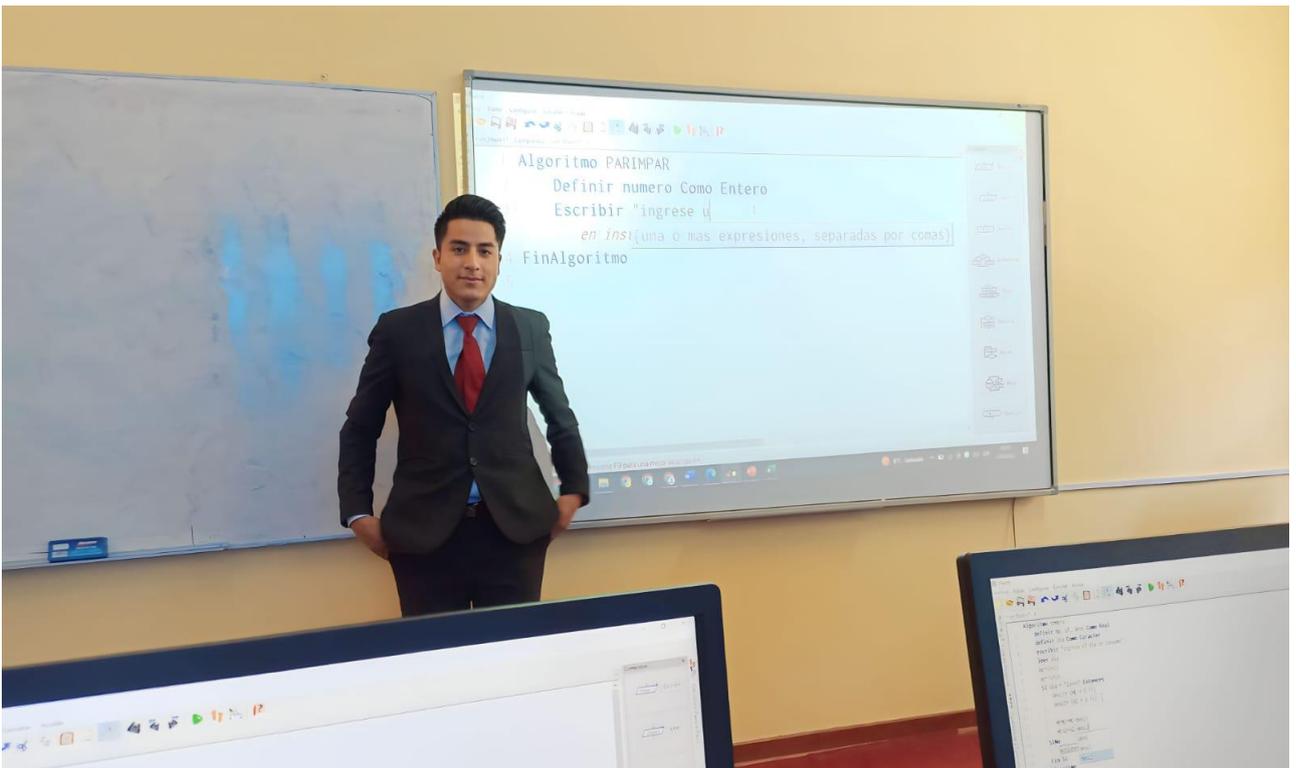
FOTOGRAFÍAS CON LOS ESTUDIANTES DE LA MUESTRA











Sesión de aprendizaje N° 01
Tema: Aprendemos sobre expresiones aritméticas y lógicas

I. DATOS INFORMATIVOS:

1.1. ASIGNATURA:	Metodología de la programación
1.2. SEMESTRE:	III
1.3. UNIDAD:	I: Procesos básicos para la resolución de problemas con ordenadores
1.4. SEMANA:	3ra. – abril 24 de 2023
1.5. DURACIÓN:	02 horas. De 8:00 am – 9:40 am
1.6. DOCENTE	Bach. Eliseo David BARRETO REQUIZ

II. PROPOSITOS DE APRENDIZAJE:

2.1. COMPETENCIA:	Aplica diversos procesos para desarrollar los fundamentos de la programación mediante los algoritmos, su representación y herramientas de desarrollo utilizando diferentes técnicas para dicho propósito y reconociendo la estructura general y básicas de control de un programa para la resolución de problemas computacionales diversos.
2.2. CAPACIDAD:	Identifica los procesos básicos para elaborar algoritmos con instrucciones adecuadas utilizando los tipos de datos, expresiones y operaciones de asignación en la resolución de problemas con ordenadores demostrando actitudes positivas en toda circunstancia.
2.3. INDICADOR (ES) LOGRO:	Desarrollan algoritmos y pasan a representarlos en Pseint.

III. SECUENCIA DIDÁCTICA

MOMENTOS	ACTIVIDADES	RECURSOS	TIEMPO
INICIO	<p>Motivación inicial El docente brinda una frase motivadora, saluda y da la bienvenida a los estudiantes.</p> <p>Exploración de saberes previos El docente da una introducción previa al tema que se va a desarrollar en la clase. Los estudiantes participan de manera voluntaria para generar nuevas ideas, estas ideas son aclaradas por el mediador.</p> <p>Anuncio de aprendizaje El docente presenta la diapositiva y da a conocer el tema y la actividad que se va a desarrollar.</p>	<ul style="list-style-type: none"> • Laptop • Multimedia • Diapositiva 	15 min
PROCESO	<p>El docente explica la manera de trabajo acerca de la evaluación de entrada a los estudiantes y les pide que deben desarrollarlos a mano escrita. Pasa a repartir la evaluación de entrada para que los estudiantes puedan desarrollarlos. El docente recoge la evaluación de entrada de los estudiantes y comienza a explicar acerca del algoritmo. Se brinda ejemplos para aclarar las dudas de los estudiantes. Se les pide participar a los estudiantes para entenderlo mejor acerca de los algoritmos. El docente da a conocer acerca las expresiones aritméticas, operadores relacionales, los operadores lógicos, estructura condicional. El docente reparte la ficha de los ejercicios propuestos para que los estudiantes puedan resolverlos de acuerdo a lo explicado por el docente. El docente monitorea el trabajo de los estudiantes, les brinda ayuda para que puedan salir de dudas, los ejercicios resueltos pasan a comprobar en el programa Pseint. El docente evalúa los trabajos a través de la lista de cotejo.</p>	<ul style="list-style-type: none"> • Laptop • Evaluación de entrada • Guía de trabajo • Hoja bond • Lapiceros • Plumones 	80 min

MOMENTOS	ACTIVIDADES	RECURSOS	TIEMPO
SALIDA	<p>El docente les deja como tarea culminar todos los ejercicios restantes de la actividad.</p> <p>Agradece a los estudiantes por su participación, les desea los éxitos y que sigan esforzándose.</p> <p>El docente pasa a retirarse del aula.</p>	<ul style="list-style-type: none"> Laptop Multimedia 	5 min

IV. EVALUACIÓN

4.1. TÉCNICA Presentación de productos

4.2. INSTRUMENTO Evaluación de entrada

V. REFERENCIAS

5.1. BIBLIOGRÁFICAS:

- Joyanes, L. (2009), Metodología de Fundamentos de Programación Orientado a Objetos, Madrid – España: Editorial Mcgraw – Hill
- Joyanes, L. (2013), Fundamentos de Programación. Libro de Problemas, Madrid – España: Editorial Mcgraw – Hill

5.2. ELECTRÓNICAS:

- Cabarcas, D. (2009), Algoritmos, Ask Experts Tutor, DSistem@s. Recuperado de: <https://www.coursehero.com/file/33942922/2492917-algoritmospdf/>
- Urbaz, W, (2012), El Algoritmo, una iniciación a la programación, Desarrollo Web. Recuperado de <https://desarrolloweb.com/manuales/67>

Yanahuanca, 24 de abril de 2023



Eliseo David BARRETO REQUIZ
Docente

Sesión de aprendizaje N° 02

Tema: Aprendemos sobre operaciones de asignación, expresiones aritméticas, lógicas y cadenas

I. DATOS INFORMATIVOS:

1.1. ASIGNATURA:	Metodología de la programación
1.2. SEMESTRE:	III
1.3. UNIDAD:	I: Procesos básicos para la resolución de problemas con ordenadores
1.4. SEMANA:	3ra. – abril 25 de 2023
1.5. DURACIÓN:	03 horas. De 8:00 am – 10:30 am
1.6. DOCENTE	Bach. Eliseo David BARRETO REQUIZ

II. PROPOSITOS DE APRENDIZAJE:

2.1. COMPETENCIA:	Aplica diversos procesos para desarrollar los fundamentos de la programación mediante los algoritmos, su representación y herramientas de desarrollo utilizando diferentes técnicas para dicho propósito y reconociendo la estructura general y básicas de control de un programa para la resolución de problemas computacionales diversos.
2.2. CAPACIDAD:	Identifica los procesos básicos para elaborar algoritmos con instrucciones adecuadas utilizando los tipos de datos, expresiones y operaciones de asignación en la resolución de problemas con ordenadores demostrando actitudes positivas en toda circunstancia.
2.3. INDICADOR (ES) LOGRO:	Escriben algoritmos y pasan a representarlos en Pseint.

III. SECUENCIA DIDÁCTICA

MOMENTOS	ACTIVIDADES	RECURSOS	TIEMPO
INICIO	<p>Motivación inicial El docente brinda una frase motivadora, saluda y da la bienvenida a los estudiantes.</p> <p>Exploración de saberes previos El docente da una introducción previa al tema que se va a desarrollar en la clase. Los estudiantes participan de manera voluntaria para generar nuevas ideas, estas ideas son aclaradas por el mediador.</p> <p>Anuncio de aprendizaje El docente presenta la diapositiva y da a conocer el tema y la actividad que se va a desarrollar.</p>	<ul style="list-style-type: none"> • Laptop • Multimedia • Diapositiva 	15 min
PROCESO	<p>El docente explica la manera de trabajo acerca de las guías de trabajo, los pasos que van a desarrollar.</p> <p>El docente da a conocer acerca las expresiones aritméticas, operadores relacionales, los operadores lógicos, estructura condicional, cadenas, caracteres.</p> <p>Pasa a repartir las guías de trabajo para el desarrollo de los algoritmos Se brinda ejemplos para aclarar las dudas de los estudiantes, explica los comandos del programa Pseint.</p> <p>El docente muestra ejemplos en la pizarra, desarrolla un ejercicio para que los estudiantes puedan entenderlo mejor. Se les pide participar a los estudiantes para entenderlo mejor acerca de los algoritmos, lógicas de programación.</p> <p>El docente reparte la segunda ficha de los ejercicios propuestos para que los estudiantes puedan resolverlos de acuerdo a lo explicado en las clases Los estudiantes de manera voluntaria pasan a presentar sus algoritmos en el programa de Pseint para su respectiva calificación.</p> <p>El docente monitorea el trabajo de los estudiantes, les brinda ayuda para que puedan salir de dudas, los ejercicios resueltos pasan a comprobar en el programa Pseint.</p>	<ul style="list-style-type: none"> • Laptop • Guía de trabajo • Hoja bond • Lapiceros • Plumones 	120 min

MOMENTOS	ACTIVIDADES	RECURSOS	TIEMPO
SALIDA	<p>El docente les deja como tarea investigar sitios webs de fundamentos de la programación, les brinda canales de YouTube para ampliar sus conocimientos.</p> <p>El docente realiza la explicación breve de algunas dudas que los estudiantes no pudieron entender, les brinda ejemplos, comparten ideas.</p> <p>Agradece a los estudiantes por su participación, les desea los éxitos y que sigan esforzándose.</p> <p>El docente pasa a retirarse del aula.</p>	<ul style="list-style-type: none"> Laptop Multimedia 	15 min

IV. EVALUACIÓN	
4.1. TÉCNICA	Presentación de productos
4.2. INSTRUMENTO	Guía de practica

V. REFERENCIAS	
5.1. BIBLIOGRÁFICAS:	
<ul style="list-style-type: none"> Joyanes, L. (2009), Metodología de Fundamentos de Programación Orientado a Objetos, Madrid – España: Editorial Mcgraw – Hill Joyanes, L. (2013), Fundamentos de Programación. Libro de Problemas, Madrid – España: Editorial Mcgraw – Hill 	
5.2. ELECTRÓNICAS:	
<ul style="list-style-type: none"> Cabarcas, D. (2009), Algoritmos, Ask Experts Tutor, DSistem@s. Recuperado de: https://www.coursehero.com/file/33942922/2492917-algoritmospdf/ Urbaz, W, (2012), El Algoritmo, una iniciación a la programación, Desarrollo Web. Recuperado de https://desarrolloweb.com/manuales/67 	

Yanahuanca, 25 de abril de 2023



Eliseo David BARRETO REQUIZ
Docente

Sesión de aprendizaje N° 03
Tema: Aprendemos sobre las herramientas de programación

I. DATOS INFORMATIVOS:

1.1. ASIGNATURA:	Metodología de la programación
1.2. SEMESTRE:	III
1.3. UNIDAD:	I: Procesos básicos para la resolución de problemas con ordenadores
1.4. SEMANA:	4ta. – mayo 01 de 2023
1.5. DURACIÓN:	02 horas. De 8:00 am – 9:40 am
1.6. DOCENTE	Bach. Eliseo David BARRETO REQUIZ

II. PROPOSITOS DE APRENDIZAJE:

2.1. COMPETENCIA:	Aplica diversos procesos para desarrollar los fundamentos de la programación mediante los algoritmos, su representación y herramientas de desarrollo utilizando diferentes técnicas para dicho propósito y reconociendo la estructura general y básicas de control de un programa para la resolución de problemas computacionales diversos.
2.2. CAPACIDAD:	Comprende la estructura de un programa, instrucciones, elementos básicos y la escritura de algoritmos representando los algoritmos con las estructuras secuenciales, selectivas, repetitivas simples y anidadas valorando los procesos realizados.
2.3. INDICADOR (ES) LOGRO:	Escriben algoritmos y pasan a representarlos en Pseint.

III. SECUENCIA DIDÁCTICA

MOMENTOS	ACTIVIDADES	RECURSOS	TIEMPO
INICIO	<p>Motivación inicial El docente brinda una frase motivadora, saluda y da la bienvenida a los estudiantes.</p> <p>Exploración de saberes previos El docente da una introducción previa al tema que se va a desarrollar en la clase. Los estudiantes participan de manera voluntaria para generar nuevas ideas, estas ideas son aclaradas por el mediador.</p> <p>Anuncio de aprendizaje El docente presenta la diapositiva y da a conocer el tema y la actividad que se va a desarrollar.</p>	<ul style="list-style-type: none"> • Laptop • Multimedia • Diapositiva 	15 min
PROCESO	<ul style="list-style-type: none"> • El docente explica la manera de trabajo acerca de las guías de trabajo, los pasos que van a desarrollar. • El docente da a conocer acerca de las herramientas de programación, para que los estudiantes puedan entender acerca de las distintas funciones y las diferencias que cumplen cada uno de ellos. • Pasa a repartir las guías de trabajo para el desarrollo de los algoritmos • Se brinda ejemplos para aclarar las dudas de los estudiantes, explica los comandos del programa Pseint. • El docente muestra ejemplos en la pizarra, desarrolla un ejercicio para que los estudiantes puedan entenderlo mejor. • Se les pide participar a los estudiantes para entenderlo mejor acerca de los algoritmos, lógicas de programación. • Los estudiantes de manera voluntaria pasan a presentar sus algoritmos en el programa de Pseint para su respectiva calificación. • El docente califica sus trabajos presentados en pizarra de cada uno de los estudiantes. 	<ul style="list-style-type: none"> • Laptop • Guía de trabajo • Hoja bond • Lapiceros • Plumones 	80 min

MOMENTOS	ACTIVIDADES	RECURSOS	TIEMPO
SALIDA	<p>El docente realiza la explicación breve de algunas dudas que los estudiantes no pudieron entender, les brinda ejemplos, comparten ideas.</p> <p>Agradece a los estudiantes por su participación, les desea los éxitos y que sigan esforzándose.</p> <p>El docente pasa a retirarse del aula.</p>	<ul style="list-style-type: none"> Laptop Multimedia 	5 min

IV. EVALUACIÓN	
4.1. TÉCNICA	Presentación de productos
4.2. INSTRUMENTO	Guía de practica

V. REFERENCIAS	
5.1. BIBLIOGRÁFICAS:	
<ul style="list-style-type: none"> Joyanes, L. (2009), Metodología de Fundamentos de Programación Orientado a Objetos, Madrid – España: Editorial Mcgraw – Hill Joyanes, L. (2013), Fundamentos de Programación. Libro de Problemas, Madrid – España: Editorial Mcgraw – Hill 	
5.2. ELECTRÓNICAS:	
<ul style="list-style-type: none"> Cabarcas, D. (2009), Algoritmos, Ask Experts Tutor, DSistem@s. Recuperado de: https://www.coursehero.com/file/33942922/2492917-algoritmospdf/ Urbaz, W, (2012), El Algoritmo, una iniciación a la programación, Desarrollo Web. Recuperado de https://desarrolloweb.com/manuales/67 	

Yanahuanca, 01 de mayo de 2023



Eliseo David BARRETO REQUIZ
Docente

Sesión de aprendizaje N° 04

Tema: Aprendemos sobre la estructura de un programa, instrucciones, tipos

I. DATOS INFORMATIVOS:

1.1. ASIGNATURA:	Metodología de la programación
1.2. SEMESTRE:	III
1.3. UNIDAD:	I: Procesos básicos para la resolución de problemas con ordenadores
1.4. SEMANA:	4ta. – mayo 02 de 2023
1.5. DURACIÓN:	03 horas. De 8:00 am – 10:30 am
1.6. DOCENTE	Bach. Eliseo David BARRETO REQUIZ

II. PROPOSITOS DE APRENDIZAJE:

2.1. COMPETENCIA:	Aplica diversos procesos para desarrollar los fundamentos de la programación mediante los algoritmos, su representación y herramientas de desarrollo utilizando diferentes técnicas para dicho propósito y reconociendo la estructura general y básicas de control de un programa para la resolución de problemas computacionales diversos.
2.2. CAPACIDAD:	Comprende la estructura de un programa, instrucciones, elementos básicos y la escritura de algoritmos representando los algoritmos con las estructuras secuenciales, selectivas, repetitivas simples y anidadas valorando los procesos realizados.
2.3. INDICADOR (ES) LOGRO:	Escriben algoritmos y pasan a representarlos en Pseint.

III. SECUENCIA DIDÁCTICA

MOMENTOS	ACTIVIDADES	RECURSOS	TIEMPO
INICIO	<p>Motivación inicial El docente brinda una frase motivadora, saluda y da la bienvenida a los estudiantes.</p> <p>Exploración de saberes previos El docente da una introducción previa al tema que se va a desarrollar en la clase. Los estudiantes participan de manera voluntaria para generar nuevas ideas, estas ideas son aclaradas por el mediador.</p> <p>Anuncio de aprendizaje El docente presenta la diapositiva y da a conocer el tema y la actividad que se va a desarrollar.</p>	<ul style="list-style-type: none"> • Laptop • Multimedia • Diapositiva 	15 min
PROCESO	<ul style="list-style-type: none"> • El docente explica la manera de trabajo acerca de las guías de trabajo, los pasos que van a desarrollar. • El docente da a conocer acerca la estructura de un programa expresiones aritméticas, operadores relacionales, los operadores lógicos, estructura condicional, cadenas. • Pasa a repartir las guías de trabajo para el desarrollo de los algoritmos • Se brinda ejemplos para aclarar las dudas de los estudiantes, explica los comandos del programa Pseint. • El docente muestra ejemplos en la pizarra, desarrolla un ejercicio para que los estudiantes puedan entenderlo mejor. • Se les pide participar a los estudiantes para entenderlo mejor acerca de los algoritmos, lógicas de programación. • Los estudiantes de manera voluntaria pasan a presentar sus algoritmos en el programa de Pseint para su respectiva calificación. <p>El docente monitorea el trabajo de los estudiantes, les brinda ayuda para que puedan salir de dudas, los ejercicios resueltos pasan a comprobar en el programa Pseint.</p>	<ul style="list-style-type: none"> • Laptop • Guía de trabajo • Hoja bond • Lapiceros • Plumones 	120 min

MOMENTOS	ACTIVIDADES	RECURSOS	TIEMPO
SALIDA	<ul style="list-style-type: none"> El docente realiza la explicación breve de algunas dudas que los estudiantes no pudieron entender, les brinda ejemplos, comparten ideas. Agradece a los estudiantes por su participación, les desea los éxitos y que sigan esforzándose. El docente pasa a retirarse del aula. 	<ul style="list-style-type: none"> Laptop Multimedia 	15 min

IV. EVALUACIÓN	
4.1. TÉCNICA	Presentación de productos
4.2. INSTRUMENTO	Guía de practica

V. REFERENCIAS	
5.1. BIBLIOGRÁFICAS:	
<ul style="list-style-type: none"> Joyanes, L. (2009), Metodología de Fundamentos de Programación Orientado a Objetos, Madrid – España: Editorial Mcgraw – Hill Joyanes, L. (2013), Fundamentos de Programación. Libro de Problemas, Madrid – España: Editorial Mcgraw – Hill 	
5.2. ELECTRÓNICAS:	
<ul style="list-style-type: none"> Cabarcas, D. (2009), Algoritmos, Ask Experts Tutor, DSistem@s. Recuperado de: https://www.coursehero.com/file/33942922/2492917-algoritmospdf/ Urbaez, W, (2012), El Algoritmo, una incianción a la programación, Desarrollo Web. Recuperado de https://desarrolloweb.com/manuales/67 	

Yanahuanca, 02 de mayo de 2023



Eliseo David BARRETO REQUIZ
Docente

Sesión de aprendizaje N° 05

Tema: Aprendemos sobre elementos básicos de un programa.

I. DATOS INFORMATIVOS:

1.1. ASIGNATURA:	Metodología de la programación
1.2. SEMESTRE:	III
1.3. UNIDAD:	II: Programación estructurada. Estructuras de control
1.4. SEMANA:	5ta. – mayo 08 de 2023
1.5. DURACIÓN:	02 horas. De 8:00 am – 9:40 am
1.6. DOCENTE	Bach. Kevin OLIVAS ARRIETA

II. PROPOSITOS DE APRENDIZAJE:

2.1. COMPETENCIA:	Aplica diversos procesos para desarrollar los fundamentos de la programación mediante los algoritmos, su representación y herramientas de desarrollo utilizando diferentes técnicas para dicho propósito y reconociendo la estructura general y básicas de control de un programa para la resolución de problemas computacionales diversos.
2.2. CAPACIDAD:	Identifica los procesos básicos para elaborar algoritmos con instrucciones adecuadas utilizando los tipos de datos, expresiones y operaciones de asignación en la resolución de problemas con ordenadores demostrando actitudes positivas en toda circunstancia.
2.3. INDICADOR (ES) LOGRO:	Desarrollan algoritmos y pasan a representarlos en Pseint.

III. SECUENCIA DIDÁCTICA

MOMENTOS	ACTIVIDADES	RECURSOS	TIEMPO
INICIO	<p>Motivación inicial El docente se presenta, saluda y da la bienvenida a los estudiantes, brinda unas palabras de reflexión.</p> <p>Exploración de saberes previos El docente da una introducción previa al tema que se va a desarrollar en la clase. Los estudiantes participan de manera voluntaria para generar nuevas ideas, estas ideas son aclaradas por el mediador.</p> <p>Anuncio de aprendizaje El docente presenta la diapositiva y da a conocer el tema y la actividad que se va a desarrollar.</p>	<ul style="list-style-type: none"> • Laptop • Multimedia • Diapositiva 	15 min
PROCESO	<p>El docente explica la manera de trabajo acerca de las guías de trabajo, los pasos que van a desarrollar.</p> <p>El docente da a conocer acerca de los elementos básicos de un programa, con la ayuda de una diapositiva.</p> <p>El docente muestra ejemplos en la pizarra, desarrolla un ejercicio para que los estudiantes puedan entenderlo mejor.</p> <p>Se les pide participar a los estudiantes para entenderlo mejor acerca de los algoritmos, lógicas de programación.</p> <p>El docente reparte fichas de los ejercicios propuestos para que los estudiantes puedan resolverlos de acuerdo a lo explicado en las clases</p> <p>Los estudiantes de manera voluntaria pasan a presentar sus algoritmos en el programa de Pseint para su respectiva calificación.</p> <p>El docente monitorea el trabajo de los estudiantes, les brinda ayuda para que puedan salir de dudas, los ejercicios resueltos pasan a comprobar en el programa Pseint.</p>	<ul style="list-style-type: none"> • Laptop • Evaluación de entrada • Guía de trabajo • Hoja bond • Lapiceros • Plumones 	80 min

MOMENTOS	ACTIVIDADES	RECURSOS	TIEMPO
SALIDA	<p>El docente les deja como tarea culminar todos los ejercicios restantes de la actividad.</p> <p>Agradece a los estudiantes por su participación, les desea los éxitos y que sigan esforzándose.</p> <p>El docente pasa a retirarse del aula.</p>	<ul style="list-style-type: none"> Laptop Multimedia 	5 min

IV. EVALUACIÓN

4.1. TÉCNICA Presentación de productos

4.2. INSTRUMENTO Evaluación de entrada

V. REFERENCIAS

5.1. BIBLIOGRÁFICAS:

- Joyanes, L. (2009), Metodología de Fundamentos de Programación Orientado a Objetos, Madrid – España: Editorial Mcgraw – Hill
- Joyanes, L. (2013), Fundamentos de Programación. Libro de Problemas, Madrid – España: Editorial Mcgraw – Hill

5.2. ELECTRÓNICAS:

- Cabarcas, D. (2009), Algoritmos, Ask Experts Tutor, DSistem@s. Recuperado de: <https://www.coursehero.com/file/33942922/2492917-algoritmospdf/>
- Urbaz, W, (2012), El Algoritmo, una iniciación a la programación, Desarrollo Web. Recuperado de <https://desarrolloweb.com/manuales/67>

Yanahuanca, 08 de mayo de 2023



Kevin OLIVAS ARRIETA
Docente

Sesión de aprendizaje N° 06
Tema: Aprendemos sobre programación estructurada.

I. DATOS INFORMATIVOS:

1.1. ASIGNATURA:	Metodología de la programación
1.2. SEMESTRE:	III
1.3. UNIDAD:	II: Programación estructurada. Estructuras de control
1.4. SEMANA:	6ta. – mayo 09 de 2023
1.5. DURACIÓN:	03 horas. De 8:00 am – 10:30 am
1.6. DOCENTE	Bach. Kevin OLIVAS ARRIETA

II. PROPOSITOS DE APRENDIZAJE:

2.1. COMPETENCIA:	Aplica diversos procesos para desarrollar los fundamentos de la programación mediante los algoritmos, su representación y herramientas de desarrollo utilizando diferentes técnicas para dicho propósito y reconociendo la estructura general y básicas de control de un programa para la resolución de problemas computacionales diversos.
2.2. CAPACIDAD:	Identifica los procesos básicos para elaborar algoritmos con instrucciones adecuadas utilizando los tipos de datos, expresiones y operaciones de asignación en la resolución de problemas con ordenadores demostrando actitudes positivas en toda circunstancia.
2.3. INDICADOR (ES) LOGRO:	Escriben algoritmos y pasan a representarlos en Pseint.

III. SECUENCIA DIDÁCTICA

MOMENTOS	ACTIVIDADES	RECURSOS	TIEMPO
INICIO	<p>Motivación inicial El docente brinda una reflexión frente al horario, saluda y da la bienvenida a los estudiantes.</p> <p>Exploración de saberes previos El docente da una introducción previa al tema que se va a desarrollar en la clase. Los estudiantes participan de manera voluntaria para generar nuevas ideas, estas ideas son aclaradas por el mediador.</p> <p>Anuncio de aprendizaje El docente presenta la diapositiva y da a conocer el tema y la actividad que se va a desarrollar.</p>	<ul style="list-style-type: none"> • Laptop • Multimedia • Diapositiva 	15 min
PROCESO	<p>El docente explica la manera de trabajo acerca de las guías de trabajo, los pasos que van a desarrollar.</p> <p>El docente da a conocer acerca de la programación estructurada con los comandos si-entonces y según.</p> <p>Pasa a repartir las guías de trabajo para el desarrollo de los algoritmos Se brinda ejemplos para aclarar las dudas de los estudiantes, explica los comandos del programa Pseint.</p> <p>El docente muestra ejemplos en la pizarra, desarrolla un ejercicio para que los estudiantes puedan entenderlo mejor. Se les pide participar a los estudiantes para entenderlo mejor acerca de los algoritmos, lógicas de programación.</p> <p>El docente reparte la segunda ficha de los ejercicios propuestos para que los estudiantes puedan resolverlos de acuerdo a lo explicado en las clases Los estudiantes de manera voluntaria pasan a presentar sus algoritmos en el programa de Pseint para su respectiva calificación.</p> <p>El docente monitorea el trabajo de los estudiantes, les brinda ayuda para que puedan salir de dudas, los ejercicios resueltos pasan a comprobar en el programa Pseint.</p>	<ul style="list-style-type: none"> • Laptop • Guía de trabajo • Hoja bond • Lapiceros • Plumones 	120 min

MOMENTOS	ACTIVIDADES	RECURSOS	TIEMPO
SALIDA	<p>El docente les deja como tarea investigar sitios webs de fundamentos de la programación, les brinda canales de YouTube para ampliar sus conocimientos.</p> <p>El docente realiza la explicación breve de algunas dudas que los estudiantes no pudieron entender, les brinda ejemplos, comparten ideas.</p> <p>Agradece a los estudiantes por su participación, les desea los éxitos y que sigan esforzándose.</p> <p>El docente pasa a retirarse del aula.</p>	<ul style="list-style-type: none"> Laptop Multimedia 	15 min

IV. EVALUACIÓN	
4.1. TÉCNICA	Presentación de productos
4.2. INSTRUMENTO	Guía de practica

V. REFERENCIAS	
5.1. BIBLIOGRÁFICAS:	
<ul style="list-style-type: none"> Joyanes, L. (2009), Metodología de Fundamentos de Programación Orientado a Objetos, Madrid – España: Editorial Mcgraw – Hill Joyanes, L. (2013), Fundamentos de Programación. Libro de Problemas, Madrid – España: Editorial Mcgraw – Hill 	
5.2. ELECTRÓNICAS:	
<ul style="list-style-type: none"> Cabarcas, D. (2009), Algoritmos, Ask Experts Tutor, DSistem@s. Recuperado de: https://www.coursehero.com/file/33942922/2492917-algoritmospdf/ Urbaz, W, (2012), El Algoritmo, una inciación a la programación, Desarrollo Web. Recuperado de https://desarrolloweb.com/manuales/67 	

Yanahuanca, 09 de mayo de 2023



Kevin OLIVAS ARRIETA
Docente

Sesión de aprendizaje N° 07

Tema: Aprendemos sobre la Programación estructurada: Básica secuencial.

I. DATOS INFORMATIVOS:

1.1. ASIGNATURA:	Metodología de la programación
1.2. SEMESTRE:	III
1.3. UNIDAD:	II:
1.4. SEMANA:	7tma. – mayo 15 de 2023
1.5. DURACIÓN:	02 horas. De 8:00 am – 9:40 am
1.6. DOCENTE	Bach. Kevin OLIVAS ARRIETA

II. PROPOSITOS DE APRENDIZAJE:

2.1. COMPETENCIA:	Aplica diversos procesos para desarrollar los fundamentos de la programación mediante los algoritmos, su representación y herramientas de desarrollo utilizando diferentes técnicas para dicho propósito y reconociendo la estructura general y básicas de control de un programa para la resolución de problemas computacionales diversos.
2.2. CAPACIDAD:	Comprende la estructura de un programa, instrucciones, elementos básicos y la escritura de algoritmos representando los algoritmos con las estructuras secuenciales, selectivas, repetitivas simples y anidadas valorando los procesos realizados.
2.3. INDICADOR (ES) LOGRO:	Escriben algoritmos y pasan a representarlos en Pseint.

III. SECUENCIA DIDÁCTICA

MOMENTOS	ACTIVIDADES	RECURSOS	TIEMPO
INICIO	<p>Motivación inicial El docente brinda una frase motivadora, saluda y da la bienvenida a los estudiantes.</p> <p>Exploración de saberes previos El docente da una introducción previa al tema que se va a desarrollar en la clase. Los estudiantes participan de manera voluntaria para generar nuevas ideas, estas ideas son aclaradas por el mediador.</p> <p>Anuncio de aprendizaje El docente presenta la diapositiva y da a conocer el tema y la actividad que se va a desarrollar.</p>	<ul style="list-style-type: none"> • Laptop • Multimedia • Diapositiva 	15 min
PROCESO	<ul style="list-style-type: none"> • El docente explica la manera de trabajo acerca de las guías de trabajo, los pasos que van a desarrollar. • El docente da a conocer acerca de la programación estructurada con los comandos mientras, repetir y para. • Pasa a repartir las guías de trabajo para el desarrollo de los algoritmos • Se brinda ejemplos para aclarar las dudas de los estudiantes, explica los comandos del programa Pseint. • El docente muestra ejemplos en la pizarra, desarrolla un ejercicio para que los estudiantes puedan entenderlo mejor. • Se les pide participar a los estudiantes para entenderlo mejor acerca de los algoritmos, lógicas de programación. • Los estudiantes de manera voluntaria pasan a presentar sus algoritmos en el programa de Pseint para su respectiva calificación. • El docente califica sus trabajos presentados en pizarra de cada uno de los estudiantes. 	<ul style="list-style-type: none"> • Laptop • Guía de trabajo • Hoja bond • Lapiceros • Plumones 	80 min

MOMENTOS	ACTIVIDADES	RECURSOS	TIEMPO
SALIDA	<p>El docente realiza la explicación breve de algunas dudas que los estudiantes no pudieron entender, les brinda ejemplos, comparten ideas.</p> <p>Agradece a los estudiantes por su participación, les desea los éxitos y que sigan esforzándose.</p> <p>El docente pasa a retirarse del aula.</p>	<ul style="list-style-type: none"> Laptop Multimedia 	5 min

IV. EVALUACIÓN	
4.1. TÉCNICA	Presentación de productos
4.2. INSTRUMENTO	Guía de practica

V. REFERENCIAS	
5.1. BIBLIOGRÁFICAS:	
<ul style="list-style-type: none"> Joyanes, L. (2009), Metodología de Fundamentos de Programación Orientado a Objetos, Madrid – España: Editorial Mcgraw – Hill Joyanes, L. (2013), Fundamentos de Programación. Libro de Problemas, Madrid – España: Editorial Mcgraw – Hill 	
5.2. ELECTRÓNICAS:	
<ul style="list-style-type: none"> Cabarcas, D. (2009), Algoritmos, Ask Experts Tutor, DSistem@s. Recuperado de: https://www.coursehero.com/file/33942922/2492917-algoritmospdf/ Urbaz, W, (2012), El Algoritmo, una inciación a la programación, Desarrollo Web. Recuperado de https://desarrolloweb.com/manuales/67 	

Yanahuanca, 15 de mayo de 2023



Kevin OLIVAS ARRIETA
Docente

Sesión de aprendizaje N° 08

Tema: Aprendemos sobre programación estructurada repetitiva simple.

I. DATOS INFORMATIVOS:

1.1. ASIGNATURA:	Metodología de la programación
1.2. SEMESTRE:	III
1.3. UNIDAD:	II: Programación estructurada. Estructuras de control
1.4. SEMANA:	8va. – mayo 16 de 2023
1.5. DURACIÓN:	03 horas. De 8:00 am – 10:30 am
1.6. DOCENTE	Bach. Kevin OLIVAS ARRIETA

II. PROPÓSITOS DE APRENDIZAJE:

2.1. COMPETENCIA:	Aplica diversos procesos para desarrollar los fundamentos de la programación mediante los algoritmos, su representación y herramientas de desarrollo utilizando diferentes técnicas para dicho propósito y reconociendo la estructura general y básicas de control de un programa para la resolución de problemas computacionales diversos.
2.2. CAPACIDAD:	Comprende la estructura de un programa, instrucciones, elementos básicos y la escritura de algoritmos representando los algoritmos con las estructuras secuenciales, selectivas, repetitivas simples y anidadas valorando los procesos realizados.
2.3. INDICADOR (ES) LOGRO:	Escriben algoritmos y pasan a representarlos en Pseint.

III. SECUENCIA DIDÁCTICA

MOMENTOS	ACTIVIDADES	RECURSOS	TIEMPO
INICIO	<p>Motivación inicial Presentación de un problema de acuerdo a la guía de clases. Participan en una discusión controversial propuesta para la clase de acuerdo a los procesos desarrollados en la resolución del problema.</p> <p>Exploración de saberes previos El docente da una introducción previa al tema que se va a desarrollar en la clase. Los estudiantes participan de manera voluntaria para generar nuevas ideas, estas ideas son aclaradas por el mediador.</p>	<ul style="list-style-type: none"> • Laptop • Multimedia • Diapositiva 	15 min
PROCESO	<ul style="list-style-type: none"> • El docente presenta ejercicios propuestos concernientes al tema. • Los estudiantes de manera voluntaria pasan a presentar sus algoritmos en el programa de Pseint para su respectiva calificación. • El docente explica la manera de trabajo acerca de la evaluación de salida a los estudiantes y les pide que deben desarrollarlos a mano escrita y en el programa Pseint. • Pasa a repartir la evaluación de salida para que los estudiantes puedan desarrollarlos. • El docente recoge la evaluación de salida de los estudiantes y resuelve todas las preguntas brindadas en el examen. 	<ul style="list-style-type: none"> • Laptop • Guía de trabajo • Hoja bond • Lapiceros • Plumones 	120 min
SALIDA	<ul style="list-style-type: none"> • Agradece a los estudiantes por su participación, les desea los éxitos y que sigan esforzándose. • El docente pasa a retirarse del aula. 	<ul style="list-style-type: none"> • Laptop • Multimedia 	15 min

IV. EVALUACIÓN

4.1. TÉCNICA	Presentación de productos
---------------------	---------------------------

4.2. INSTRUMENTO	Evaluación de salida
-------------------------	----------------------

V. REFERENCIAS

5.1. BIBLIOGRÁFICAS:

- Joyanes, L. (2009), Metodología de Fundamentos de Programación Orientado a Objetos, Madrid – España: Editorial Mcgraw – Hill
- Joyanes, L. (2013), Fundamentos de Programación. Libro de Problemas, Madrid – España: Editorial Mcgraw – Hill

5.2. ELECTRÓNICAS:

- Cabarcas, D. (2009), Algoritmos, Ask Experts Tutor, DSistem@s. Recuperado de: <https://www.coursehero.com/file/33942922/2492917-algoritmospdf/>
- Urbaez, W, (2012), El Algoritmo, una iniciación a la programación, Desarrollo Web. Recuperado de <https://desarrolloweb.com/manuales/67>

Yanahuanca, 16 de mayo de 2023



Kevin OLIVAS ARRIETA
Docente

MATRIZ DE CONSISTENCIA

TITULO DE LA INVESTIGACIÓN: “Aplicación de algoritmos para el aprendizaje de la programación Informática en estudiantes universitarios de la Filial Yanahuanca, 2023”.

TESISTAS:

- **Bach. Eliseo David BARRETO REQUIZ**
- **Bach. Kevin OLIVAS ARRIETA**

PROBLEMAS	OBJETIVOS	HIPÓTESIS	VARIABLES	METODOLOGIA	POBLACION Y MUESTRA	TECNICAS E INSTRUMENTOS
<p>Problema General: ¿Cuál es la influencia de la aplicación de los algoritmos para el aprendizaje de la programación informática en estudiantes del III ciclo del Programa de Estudios de tecnología informática y telecomunicaciones – filial Yanahuanca, 2023?</p>	<p>Objetivo General: Determinar la influencia de los algoritmos para el aprendizaje de la programación informática, en estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023.</p>	<p>Hipótesis General: Los algoritmos influyen significativamente en el aprendizaje de la programación informática en los estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023.</p> <p>Hipótesis Nula: Los algoritmos no influyen significativamente en el aprendizaje de la programación informática en los estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023.</p>	<p>Variable independiente: Algoritmos</p> <p>Variable dependiente: Programación informática</p> <p>Variables intervinientes: Creatividad Innovación</p>	<p>Tipo de investigación: Aplicada</p> <p>Diseño de investigación: Pre experimental</p>	<p>Población: Está conformado por la totalidad de alumnos del programa de estudios de Tecnología Informática y Telecomunicacion es filial- Yanahuanca, que suman 75 estudiantes.</p> <p>Muestra: Se ha determinado tomar como muestra a los alumnos del tercer ciclo que suman 16 como grupo experimental y de control, respectivamente.</p>	<p>Técnicas: Encuesta Observación directa</p> <p>Instrumentos: Prueba pedagógica Ficha de observación Guía de practica</p>
<p>Problemas Específicos: ✓ ¿Cuál es el nivel de aprendizaje de la programación informática antes de la</p>	<p>Objetivos Específicos: ✓ Diagnosticar el nivel aprendizaje de la programación informática antes de la aplicación de</p>	<p>Hipótesis Especifica: ✓ El nivel de aprendizaje de la programación antes de la aplicación de los algoritmos en estudiantes del III ciclo</p>				

PROBLEMAS	OBJETIVOS	HIPÓTESIS	VARIABLES	METODOLOGIA	POBLACION Y MUESTRA	TECNICAS E INSTRUMENTOS
<p>aplicación de algoritmos en estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023?</p> <ul style="list-style-type: none"> • ¿Qué características presenta la aplicación de los algoritmos para el aprendizaje de la programación informática en estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023? ✓ ¿Cuál es el nivel de aprendizaje de la programación informática después de la aplicación de los algoritmos en estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y 	<p>algoritmos en estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023</p> <ul style="list-style-type: none"> ✓ Aplicar los algoritmos durante el aprendizaje de la programación informática, en estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023. ✓ Evaluar el nivel de aprendizaje de la programación informática después de la aplicación de los algoritmos en estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023. 	<p>del P.E. de tecnología informática y telecomunicaciones – filial Yanahuanca, 2023 es bajo.</p> <ul style="list-style-type: none"> ✓ Las herramientas y recursos de los algoritmos para el desarrollo de la programación informática en estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023 es óptimo. ✓ El nivel de desarrollo de la programación informática después de la aplicación de los algoritmos en estudiantes del III ciclo del Programa de Estudios de Tecnología Informática y Telecomunicaciones – filial Yanahuanca, 2023 es alto 				

PROBLEMAS	OBJETIVOS	HIPÓTESIS	VARIABLES	METODOLOGIA	POBLACION Y MUESTRA	TECNICAS E INSTRUMENTOS
Telecomunicaciones – filial Yanahuanca,2023?						