UNIVERSIDAD NACIONAL "PEDRO RUIZ GALLO"

FACULTAD DE CIENCIAS HISTÓRICO SOCIALES Y EDUCACIÓN

UNIDAD DE POSGRADO

MAESTRÍA EN CIENCIAS DE LA EDUCACIÓN

ESTRATEGIAS DIDÁCTICAS PARA FACILITAR EL APRENDIZAJE DEL DISEÑO Y CONSTRUCCIÓN DE ALGORITMOS DE LOS ESTUDIANTES DEL III CICLO DE COMPUTACIÓN E INFORMÁTICA DEL I.E.S.T. PÚBLICO "PERÚ – JAPÓN", CHACHAPOYAS - 2015

TESIS

PARA OBTENER EL GRADO ACADÉMICO DE MAESTRA EN CIENCIAS DE LA EDUCACIÓN CON MENCIÓN EN INVESTIGACIÓN Y DOCENCIA

AUTORA:
Ing. DANNY MAGALLYT CHUQUIZUTA HERRERA

ASESOR:
M.Sc. MIGUEL ALFARO BARRANTES

LAMBAYEQUE – PERÚ
2016

UNIVERSIDAD NACIONAL "PEDRO RUIZ GALLO"

FACULTAD DE CIENCIAS HISTÓRICO SOCIALES Y EDUCACIÓN

UNIDAD DE POSGRADO

MAESTRÍA EN CIENCIAS DE LA EDUCACIÓN

ESTRATEGIAS DIDÁCTICAS PARA FACILITAR EL APRENDIZAJE DEL DISEÑO Y CONSTRUCCIÓN DE ALGORITMOS DE LOS ESTUDIANTES DEL III CICLO DE COMPUTACIÓN E INFORMÁTICA DEL I.E.S.T. PÚBLICO "PERÚ – JAPÓN", CHACHAPOYAS - 2015

TESIS

PARA OBTENER EL GRADO ACADÉMICO DE MAESTRA EN CIENCIAS DE LA EDUCACIÓN CON MENCIÓN EN INVESTIGACIÓN Y DOCENCIA

ING. DANNY M. CHUQUIZUTA HERRERA
AUTOR

M.Sc. MIGUEL ALFARO BARRANTES
ASESOR

APROBADA POR:

M. Sc. CARLOS REYES APONTE
PRESIDENTE DEL JURADO

Dra. ROSA ELENA SÁNCHEZ RAMÍREZ
SECRETARIA DEL JURADO

M. Sc. LUIS PÉREZ CABREJOS
VOCAL DEL JURADO

ACTA DE SUSTENTACIÓN DE TESIS

Siendo las 9:00 horas del día 11 de noviembre del año dos mil dieciséis, en la Sala de Sustentaciones de la Facultad de Ciencias Histórico Sociales y Educación de la Universidad Nacional "Pedro Ruiz Gallo" de Lambayeque, se reunieron los miembros del jurado, designados mediante Resolución N° 1145-2016-UP-D-FACHSE de fecha 23-06-16 conformado por:

M.Sc. Lazaro Reyes Aponte PRESIDENTE(A)
Dra. Rosa Elena Sánchez Ramírez SECRETARIO(A)
M.Sc. Luis Pérez Calderón VOCAL
M.Sc. Miguel Alfaro Barrantes ASESOR(A)

con la finalidad de evaluar la tesis titulada Estrategias Didácticas para facilitar el aprendizaje del Diseño y Construcción de Algoritmos de los estudiantes del III ciclo de Computación e Informática del I.E.S.T. Público "Perú - Japón", Chiclayo - 2015

presentado por (el)(la)(los)(las) tesista(s) Danny Magallyt Chuquiguita Herrera

sustentación que es autorizada mediante Resolución N° 2205-2016-UP-D-FACHSE

de fecha 07 de Noviembre 2016

El Presidente del jurado autorizó el inicio del acto académico y después de la sustentación, los señores miembros del jurado formularon las observaciones y preguntas correspondientes, las mismas que fueron absueltas por (el) (la) (los)(las) sustentante(s), quien(es) obtuvo (obtuvieron) 80 puntos que equivale al calificativo de BUENO

En consecuencia (el) (la) (los)(las) sustentante(s) queda(n) apto (s) para obtener el Grado Académico de Maestra en Ciencias de la Educación con mención en Investigación y Docencia

Siendo las 10:30 a.m. horas del mismo día, se da por concluido el acto académico, firmando la presente acta.

PRESIDENTE

SECRETARIO

VOCAL

ASESOR

Observaciones:

En el Acta de Sustentación de Tesis se evidencia el proceso de sustentación de tesis. La misma que ha sido refrendada por el jurado conformado por el presidente, secretario y vocal, mas no se registra la firma del asesor, cuya labor efectiva es durante el proceso de elaboración de tesis y su presencia en el acto de sustentación de la tesis es voluntaria. Por tanto, su ausencia no invalida el acto de sustentación.

El/la sustentante cumple con los requisitos para la emisión de su grado académico correspondiente.

Lambayeque, 28 de febrero de 2019.

UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO
FACHSE
UNIDAD DE POSGRADO
LAMBAYEQUE

Dr. JOSÉ MÁXIMO MAQUEN CASTRO
DIRECTOR UP-FACHSE

DECLARACIÓN JURADA DE ORIGINALIDAD

Ing. DANNY MAGALLYT CHUQUIZUTA HERRERA, investigadora principal y **Mg. MIGUEL ALFARO BARRANTES,** asesor de la Tesis denominada: **ESTRATEGIAS DIDÁCTICAS PARA FACILITAR EL APRENDIZAJE DEL DISEÑO Y CONSTRUCCIÓN DE ALGORITMOS DE LOS ESTUDIANTES DEL III CICLO DE COMPUTACIÓN E INFORMÁTICA DEL I.E.S.T. PÚBLICO "PERÚ – JAPÓN", CHACHAPOYAS - 2015,** declaramos bajo juramento que este trabajo no ha sido plagiado, ni contiene datos falsos. En caso se demostrara lo contrario, asumimos responsablemente la anulación de este informe y por ende el proceso administrativo, a que hubiera lugar. Que puede conducir a la anulación del título o grado emitido como consecuencia de este informe.

Ing. DANNY MAGALLYT CHUQUIZUTA HERRERA
INVESTIGADORA

Mg. MIGUEL ALFARO BARRANTES
ASESORA

DEDICATORIA

A Dios, mi señor por darme la vida.

A mi madre Maura, mi padre Luis y **mis hermanos Dionicio, Ana, Sonia, Luis y María** por su apoyo incondicional, a mis mágicas princesas **Daniela Nicole y Alanis Stacy** por su inmenso amor y comprensión y a ustedes mis estimados maestros por sus conocimientos impartidos en cada sesión de aprendizaje.

Danny M. Chuquizuta H.

AGRADECIMIENTO

Aún a pesar de saber que mi memoria me presentará como un ser injusto pues tal vez, involuntariamente, se me escapen algunas personas a quienes debería nombrar, quiero agradecer a todos mis estudiantes quienes, a lo largo de muchos años con sus inquietudes y sus dudas, me han enseñado más de lo que yo pude haberles enseñado. Fueron sus permanentes inquietudes las que poco a poco inspiraron esta investigación.

También quiero presentar mis agradecimientos a mi maestra Julia Esther Santa Cruz Mío, quien me brindó su apoyo profesional y a toda la plana docente del Programa de Maestría de la FACHSE, quienes me entregaron herramientas para entender que el mundo no hay que dejarlo ser simplemente, sino que hay que analizarlo y tratar de hacer un aporte en ese devenir dialéctico de la especie humana que justifique nuestra propia existencia. También quiero agradecer a mis amigos, tantos aportes académicos no podían quedar en vano.

Igualmente quiero agradecer a mis hijas quienes pacientemente han esperado durante muchos días y con gran expectativa la finalización de esta tesis y a mi familia por los ánimos que me infundieron para terminarlo.

Danny M. Chuquizuta H.

RESUMEN

El problema en este estudio de investigación surge al observar deficiencias en la capacidad del diseño y construcción de algoritmos en lógica de programación, en los estudiantes del III ciclo de computación e informática del I.E.S.T. Público "Perú - Japón" de Chachapoyas, lo que se evidencia en la incapacidad para diseñar algoritmos sencillos utilizando las diferentes estructuras, presentando dificultades para generar los diagramas de flujo, así como para realizar las pruebas de escritorio.

El objetivo fue diseñar, ejecutar y aplicar estrategias didácticas basadas en el uso del software PSeInt, el Aprendizaje por Descubrimiento de BRUNER, la Teoría del Conocimiento Operante de SKINNER, el Aprendizaje Significativo de Ausubel, la Teoría de PIAGET y el arte de la programación de KUNTH, para desarrollar la capacidad del diseño y construcción de algoritmos. La hipótesis considera que "Si, se diseñan, elaboran y ejecutan estrategias didácticas con base en el uso del software PSeInt, en el diseño y construcción de algoritmos, entonces los estudiantes del III ciclo de computación e informática del I.E.S.T Público "Perú – Japón", lograrán elevar la calidad de la enseñanza y aprendizaje en dicha unidad didáctica y temática.

El estudio es significativo, considerando que no existen trabajos que aborden de modo específico el asunto, por lo que el docente de la unidad didáctica tendrá a su alcance un material que le sirva de guía y oriente su trabajo pedagógico. Se trabajó con el diseño cuasi experimental de un grupo; la muestra estuvo constituida por 35 estudiantes.

Se concluye que los estudiantes antes de la aplicación del programa de estrategias didácticas, tenían un nivel deficiente en el diseño de algoritmos de lógica de programación, pero luego de aplicar el programa, los estudiantes mejoraron significativamente.

Palabras clave: software libre, PSeInt, estrategias, algoritmo, lógica de programación.

ABSTRACT

The problem is this research study arises when observing deficiencies in the capacity of the design and construction of algorithms in programming logic, in the students of the III cycle of computing and informatics of the public school Peru – Japan of Chachapoyas, which is evidenced in the incapacity to design simple algorithms using the different structures, presenting difficulties to generate the flowcharts, as well as to perform the desktop tests.

The objective was to design, execute and apply didactic strategies based on the use of PSeInt software, Bruner's discovery learning, Skinner operant knowledge theory, Ausubel's significant learning, Piaget's theory and the art of programming Kunth, to develop the capacity of the design and construction of algorithms. The hypothesis considers that if didactic strategies are designed, elaborated and executed based on the use of the PSeInt software, in the design and construction of algorithms, then the students of the III cycle of computation and informatics of the lest, will be able to raise the quality of the teaching and learning in said didactic and thematic unit.

The study is significant, considering that there are no jobs that specific issues the matter, so the teaching of the didactic unit will have at its fingertucting a material that will guide you and guides your pedagogical work. I work with the quasi experimental design of a group, the sample was constituted by 35 students.

It was concluded that students before the application of the teaching strategies program, had a poor level in the design of logical logistics of logical, but after applying the program, students improved significantly.

Keywords: free software, PSeInt, strategies, algorithm, programming logic.

ÍNDICE

RESUMEN…….....VIII

ABSTRACT……...IX

INTRODUCCIÓN……...XII

CAPÍTULO I: ANÁLISIS DEL OBJETO DE ESTUDIO…….....18

1.1. UBICACIÓN……..18

1.2. ENFOQUE HISTÓRICO TENDENCIAL DEL PROBLEMA…….……22

1.3. METODOLOGÍA…….……24

CAPÍTULO II: MARCO TEÓRICO.....……...28

2.1. ANTECEDENTES…….29

2.2. BASE TEÓRICA……...31

2.2.1. Teorías del Aprendizaje Aplicadas a las TIC……...31

2.2.1.1. Teoría de PIAGET…….....31

2.2.1.2. Aprendizaje Significativo de Ausubel…….31

2.2.1.3. Teoría del Conocimiento Operante de SKINNER……...32

2.2.1.4 Aprendizaje por Descubrimiento de BRUNER……...32

2.2.1.5. Constructivismo de PAPERT……...33

2.2.1.6. Teoría del Control Adaptativo del Pensamiento (ACT) de Anderson……..34

2.3. MARCO CONCEPTUAL……..37

2.3.1. FUNDAMENTOS Y ENFOQUE DE LA LÓGICA DE PROGRAMACIÓN.....37

- *Acerca de la lógica*…….37
- *Algoritmo*……...37
- *Programación*…….....38
- *La programación como un arte*……..38
- *Pseudocódigo*……38
- *Diagramas de Flujo*……..39
- *Unidad Didáctica de Lógica de Programación*……..39

2.3.2. ASPECTOS GENERALES DE LAS TIC EN EDUCACIÓN……35

2.3.3. ALCANCES TEÓRICOS GENERALES RESPECTO A ESTRATEGIAS DIDÁCTICAS……40

CAPÍTULO III: RESULTADOS DE LA INVESTIGACIÓN Y PROPUESTA TEÓRICA……...47
3.1. ANÁLISIS E INTERPRETACIÓN DE DATOS……..48
3.2. GRÁFICO DEL MODELO TEÓRICO DE LA PROPUESTA……61
3.3. PROPUESTA TEÓRICA……...62
CAPÍTULO IV: CONCLUSIONES, RECOMENDACIONES, REFERENCIAS BIBLIOGRÁFICAS Y ANEXOS…… 87
4.1. CONCLUSIONES…….88
4.2. RECOMENDACIONES……89
4.3. REFERENCIAS BIBLIOGRÁFICAS……...90
4.4. ANEXOS……93
4.4.1. Anexo 1: GUÍA DE OBSERVACIÓN……93
4.4.2. Anexo 2: TEST PARA MEDIR EL NIVEL DE DESARROLLO DE LA CAPACIDAD DEL "DISEÑO Y CONSTRUCCIÓN DE ALGORITMOS" EN LA UNIDAD DIDÁCTICA LÓGICA DE PROGRAMACIÓN……..95
4.4.3. Anexo 3: Sílabo……...96
4.4.4. Anexo 4: Tabla de Evaluación de Expertos……...103
4.4.5. Anexo 5: Constancia……104

INTRODUCCIÓN

El desarrollo de las capacidades de diseño y construcción de algoritmos, es materia de preocupación en los diferentes sistemas educativos de todo el mundo; esto en razón a que casi únicamente estas capacidades son consideradas indispensables en el contexto del aprendizaje de los estudiantes de la carrera profesional de computación e informática.

A nivel internacional existen centros de estudios que convocan concursos en el desarrollo de software de calidad. Esta competencia de desarrolladores de software atrae a muchos jóvenes estudiantes y no estudiantes a poner en evidencia la capacidad del diseño y construcción de algoritmos. Los ganadores siempre son de Rusia y Japón; el Perú no compite aun en ningún nivel convocado, quizás porque no tenemos representatividad en el nivel educativo.

A nivel nacional tampoco hay concursos y convocatorias sobre desarrollo de software de calidad, en el ámbito de la lógica de programación tanto en el nivel universitario y técnico.

A nivel regional, la muestra más evidente de querer superar estas deficiencias relacionada con la baja calidad educativa, está expresadas en el Proyecto educativo Regional, donde se plantean ciertas lineamientos claros a fin de que todos los actores educativos tengan cierta lucidez respecto a cómo proceder para mejorar la problemática en mención.

A nivel local, en las instituciones de educación superior pública y privada, continuamente se brindan capacitaciones a los d

ocentes, porque son conscientes de que si no se da importancia a este tema, ningún estudiante será capaz de enfrentar los contantes cambios del mundo globalizado.

A pesar de los esfuerzos bien intencionados por parte del estado peruano a través del ministerio de educación. Así como por parte de los gobiernos regionales, la problemática en torno al desarrollo de capacidad básica en los estudiantes, tales como diseño y construcción de algoritmos, sigue siendo preocupación de todos los actores vinculados con la educación, y particularmente de los docentes.

Así pues esta problemática tiene plena vigencia en el ámbito nacional, por tanto existe la necesidad de abordarla desde diferentes frentes, uno de los cuales lo constituye este trabajo de investigación, a través del cual se busca aportar con un grano de arena en aras de superar las deficiencias observadas.

La problemática específica, materia de abordaje en este estudio se ubica en el I.E.S.T. Público "Perú - Japón" de Chachapoyas, donde se ha observado en los estudiantes del III ciclo de computación e informática, en la unidad didáctica de lógica de programación, deficiencias en la capacidad del diseño y construcción de algoritmos, proponiéndose como alternativa de solución el diseño y aplicación de estrategias didácticas basada en el uso del software PSeInt.

Como antecedentes de este estudio se mencionan los siguientes trabajos:

(M. C. Ricardo Vargas de Basterra, 2006) en su tesis Modelo para la Construcción de Algoritmos Apoyados en Heurísticas, este modelo se enfoca en la parte microscópica del área de ingeniería de software concentrándose específicamente en el análisis, diseño, construcción, pruebas y validación de los algoritmos inmersos en los métodos y funciones de las clases y otros componentes de software abordando la capa más específica y concreta de la producción de software.

Tesis de maestría de la Universidad de Guayaquil, de Ortega Washington presentada en el mes de julio del 2012, cuyo título es: "El uso de la tecnología de información y comunicación en el aprendizaje significativo de los estudiantes". El autor sostiene que el uso de la tecnología en la actualidad es casi imprescindible en todos los campos de la ciencia, no solo en el campo de la educación, convirtiéndose en una herramienta importante para cada uno de los profesionales.

El problema en estudio se ha formulado de la siguiente manera ¿En qué medida la aplicación de estrategias didácticas basada en el uso del software PSeInt, desarrolla la capacidad del diseño y construcción de algoritmos en la unidad didáctica de lógica de programación, en los estudiantes del III ciclo de computación e informática – Chachapoyas 2015?

El **objeto de estudio** es: el Proceso de enseñanza aprendizaje, y el **campo de acción** las estrategias didácticas basadas en el uso del software PSeInt para desarrollar la capacidad del diseño y construcción de algoritmos.

El **Objetivo General** que persigue esta investigación es Diseñar, elaborar y ejecutar estrategias didácticas basadas en el uso del software PSeInt, el Aprendizaje por Descubrimiento de BRUNER, la Teoría del Conocimiento Operante de SKINNER, el Aprendizaje Significativo de Ausubel, la Teoría de PIAGET y el arte de la programación de KUNTH, para desarrollar la capacidad del diseño y construcción de algoritmos en la unidad didáctica de lógica de programación, en los estudiantes del III ciclo de computación e informática del I.E.S.T. Público "Perú - Japón", Chachapoyas 2015.

Los objetivos específicos son:

- Determinar el nivel de desarrollo actual de la capacidad del diseño y construcción de algoritmos en la unidad didáctica de lógica de programación, en los estudiantes del III ciclo de computación e informática del I.E.S.T. Público "Perú - Japón", Chachapoyas 2015.
- Identificar el nivel de dominio del software PSeInt por parte de los estudiantes del III ciclo de computación e informática del I.E.S.T. Público "Perú - Japón", Chachapoyas 2015.
- Fundamentar teóricamente el desarrollo de este trabajo de investigación, a través de la revisión, lectura y análisis de la literatura pertinente, que hagan posible la consolidación de argumentos que le sirvan de base y sustento.
- Diseñar las estrategias didácticas con base en el uso del software PSeInt, orientadas al desarrollo de la capacidad del diseño y construcción de algoritmos, en los estudiantes del III ciclo de computación e informática del I.E.S.T. Público "Perú - Japón", Chachapoyas 2015.
- Aplica estrategias didácticas diseñadas para el desarrollo de la capacidad del diseño y construcción de algoritmos en la unidad didáctica de lógica de programación.

La hipótesis de formula en respuesta al problema y al objetivo de la siguiente manera: "Si, se diseñan, elaboran y ejecutan estrategias didácticas con base en el uso del software PSeInt, en el diseño y construcción de algoritmos, entonces los estudiantes del

III ciclo de computación e informática del I.E.S.T Público “Perú – Japón”, lograrán elevar la calidad de la enseñanza y aprendizaje en dicha unidad didáctica y temática, entonces se desarrollara significativamente la capacidad del diseño y construcción de algoritmos en la unidad didáctica de lógica de programación en los estudiantes del III ciclo de computación e informática del I.E.S.T. Público “Perú - Japón”, Chachapoyas 2015.

La justificación e importancia de este estudio se enfoca desde tres puntos de vista:

Desde el punto de vista pedagógico, trata de constituirse en un foco de luz respecto a cómo desarrollar la capacidad del diseño y construcción de algoritmos en la unidad didáctica de lógica de programación, considerando que no existen trabajos que aborden de modo especifico el tema. Así pues el docente de la unidad didáctica tendrá a su alcance un material que le sirva como guía y oriente su trabajo pedagógico para elevar el nivel de desarrollo de dicha capacidad.

Desde el punto de vista político, enmarcado en la política educativa del ministerio de educación, trata de contribuir al desarrollo de capacidades de los estudiantes, más que de la educación tradicional. Así pues, con este trabajo de investigación se espera que los estudiantes se pongan en contacto y se familiaricen con las nuevas tecnologías de información y comunicación. Todo esto guarda estrecha relación con los propósitos educativos al 2021. De conformidad con lo que se plantea en el Diseño Curricular Básico (DCB). El proyecto Educativo Nacional (PEN) el Proyecto Educativo Regional (PER).

Desde el punto de vista social permitirá, por un lado, el trabajo en equipos por los estudiantes, permitiendo la interacción dinámica de los mismos dentro del proceso enseñanza aprendizaje: pero por otro lado beneficiara a los diversos actores educativos: docentes, alumnos, personal administrativo y jerárquico, padres de familia y sociedad en general, puesto que este trabajo estará disponible en el centro de cómputo del instituto, así como en la página web de la institución.

Este trabajo de investigación está estructurado en cuatro capítulos y se detalla a continuación:

El **capítulo I** comprende el **Análisis del objeto de estudio**; aquí se describe la ubicación del instituto en la que se ha aplicado este estudio; se analiza cómo surge el

problema en los diferentes ámbitos; se señalan con precisión las características y manifestaciones del problema; finalmente se describe la metodología utilizada.

El **capítulo II** comprende el marco teórico, donde se abordan la base teórica y el marco conceptual. En la base teórica se enfocan los elementos básicos de las teorías que sustentan este trabajo, que son el Aprendizaje por Descubrimiento de BRUNER, la Teoría del Conocimiento Operante de SKINNER, el Aprendizaje Significativo de Ausubel, la Teoría de PIAGET y el arte de la programación de KUNTH.

En la base conceptual se aborda los fundamentos y enfoque de la lógica de programación en el nivel técnico, en donde se precisan las particularidades más relevantes del trabajo de la unidad didáctica, resaltando la competencia del diseño y construcción de algoritmos por ser un componente fundamental en este estudio. También se señalan los aspectos generales de las TIC en el contexto educativo, dentro de cuya temática se encuentra inmerso el software PSeInt, y por último se abordan los elementos teóricos básicos en cuanto a estrategias didácticas.

El **capítulo III** comprende, la presentación de los resultados de la investigación; allí se analizan, interpretan y discuten los resultados obtenidos. También se presenta en este capítulo la propuesta o aporte de la investigadora, que vienen a ser las estrategias didácticas diseñadas y aplicadas en aras de solucionas la problemática encontrada.

Por último el **capítulo IV** comprende como corolario conclusiones y recomendaciones a las que se han llegado a partir de los resultados de todo el proceso de investigación; por último se señala la biografía y los anexos correspondientes.

CAPÍTULO I

Análisis del Objeto de Estudio

CAPÍTULO I
ANÁLISIS DEL OBJETO DE ESTUDIO

1.1. UBICACIÓN

El presente trabajo de investigación se desarrolló en el Instituto de Educación Superior Tecnológico Público "Perú - Japón", ubicado en el Jr. Amazonas # 120 del barrio "la Laguna", zona urbana del distrito y provincia de Chachapoyas, departamento de Amazonas.

Chachapoyas fue fundada como **San Juan de la Frontera de los Chachapoyas** por el capitán realista Alonso de Alvarado, el 5 de septiembre de 1538, es una ciudad del norte del Perú, capital de la provincia de Chachapoyas y del departamento de Amazonas, se encuentra ubicada a una altitud de 2,334 msnm, en el nororiente peruano, cerca al río Utcubamba. Se encuentra a 56 Km desde Pedro Ruiz, 236 km desde Moyobamba, 441 km desde Chiclayo y 1222 km desde Lima vía Chiclayo que se recorren en 21 horas.

Chachapoyas proviene del vocablo nativo Sachapuyos que significa "hombres de la neblina" atribuyéndoles este nombre por la densa neblina que habitualmente cubre el cerro de Puma Urco, el cual se encuentra en las inmediaciones de la ciudad; aunque otros autores afirman que proviene del Aymara: chacha "gente" y Phuyu "nube".

De acuerdo con la información del Instituto Nacional de Estadística e Informática (INEI) del censo peruano de 2007, la población estimada era de 23,202 habitantes. Presenta un clima templado, moderadamente lluvioso y con amplitud térmica moderada. La media anual de temperatura máxima y mínima (periodo 1960 – 1991) es 19,8° C y 9,2° C, respectivamente.

Es una de las ciudades más antiguas del Perú y una de las pocas que aún mantiene su aire señorial e influencia hispánica. Tierra de orquídeas, de caña de azúcar y café, de peleas de gallo y exquisitas comidas, es capaz de impresionar a todo aquel que la visite.

Algunas batallas que se libraron fueron. La batalla de ***Higos Urco***: los pobladores de Chachapoyas se incorporaron a la causa de la independencia peruana. En abril de 1821, secundaron la acción del ejército libertados de San Martin, desconocieron a las

autoridades españolas y desterraron al subdelegado Francisco Baquedano y al obispo de Maynas, Hipólito Sánchez, quienes luchaban contra la independencia.

En medio de la causa de la independencia peruana, el 6 de junio de 1821 se libró la Batalla de Higos Urco en las pampas de Higos Urco, ciudad de Chachapoyas. El jefe militar de Moyobamba, coronel José Matos, organizó un ejército de 600 hombres para sorprender a los patriotas. La organización y disciplina de los españoles no pudo nada ante el heroísmo de los patriotas que ganaron la batalla.

Aquí destacaron por su heroísmo los tenientes *José Félix Castro* (trujillano), francisco *Salas* (chileno) y *Evaristo Tafur* (chachapoyano). La mujer amazónica estuvo representada por la heroína ***Matea o Matiaza Rimachi***, que ha pasado a la posteridad como la heroína de Higos Urco. Ese mismo día el 6 de junio, después de triunfo de Higos Urco, se juró definitivamente la independencia de Chachapoyas.

Sus principales actividades económicas donde se desempeña la PEA, son la agricultura, actividad que desarrollaron en extensos campos de cultivo cercanos a sus pobladores. Por el clima templado de Chachapoyas, se obtienen buenas cosechas de cereales, caña de azúcar, café y yuca además presenta buena cabaña ganadera.

En educación cuenta con colegios públicos y privados, la ***Universidad Nacional Toribio Rodríguez de Mendoza de Amazonas***, el ***Instituto de Educación Superior Tecnológico Público "Perú - Japón"***, el ***Instituto Superior Pedagógico Toribio Rodríguez de Mendoza*** y otros centros superiores.

Es una ciudad turística visitada por gran cantidad de extranjeros y peruanos por su gran legado histórico como: ***plaza de armas,*** ubicado en el centro de la ciudad, donde resalta la ***pileta de bronce*** de origen colonial. De entre las edificaciones de la periferia, destaca la ***sede del Arzobispado***, que fuera la casa donde nació el precursor de la independencia, don Alejandro Toribio Rodríguez de Mendoza Collantes. ***Pozo de yanayacu***, fue construido en el mismo lugar en el que, según la leyenda, Santo Toribio de Mogrovejo hizo brotar agua de una roca e hizo que la gente se convierta en cecina al beber de ella, acabando con la sequía y el hambre que asolaba la región. ***Pampas de Higos Urco***, fue en este lugar que, durante las luchas por la independencia de España, se libró la batalla de higos uro, en el cual los valerosos soldados lucharon por la independencia

de Chachapoyas. ***Catarata de Gocta***, la tercera más alta del mundo se encuentra en Amazonas, al oriente peruano. ***Kuélap***, sitio arqueológico ubicado a 32 km al sur este de Chachapoyas sobre un alto promontorio a 3080 m.s.n.m. fue construida entre los años 900 y 1000, albergaba 420 casas circulares que se estiman alojaban a unas 3500 personas. Túpac Yupanqui llegó con sus tropas en 1470, pero el ejército inca fue incapaz de doblegar a los Chachapoyas. ***Sarcófagos de Karajía***, están ubicados a 48 km al nor este de Chachapoyas. Este sitio arqueológico contiene el mejor ejemplo del estilo funerario de este reino. Se trata de un conjunto de 7 sarcófagos antropomorfos de 2.5 m de alto, conocidos como *purunmachus* construido de una mezcla de arcilla, piedras y palos. ***Revash***, sitio arqueológico ubicado 65 km de Chachapoyas, también es una construcción funeraria de piedra llamadas *chullpas* que son como casas pequeñas pintadas de rojo y blanco.

Las festividades de Chachapoyas son: ***carnaval chachapoyano/ febrero – marzo***, mezcal de costumbres celendinas y chachapoyanas. Se celebra con la plantación de "humishas", grandes árboles que se adornan con globos, serpentinas prendas y otros objetos; además, se realiza un concurso de carros alegóricos, la elección de reinas de carnaval, bailes sociales y populares, juegos con agua y colores. ***Semana turística*** de Chachapoyas**/ junio 1 al 7**, actividades creativas, sociales y culturales. Destaca el raymillacta de la Chachapoyas, en el que las comunidades de la región reúnen a sus danzantes para mostrar lo mejor de su folclore.

Fiesta patronal ***Virgen de Asunta***/ **agosto 7 al 15**, la imagen de la Virgen recorre las calles de la ciudad de Chachapoyas que se encuentran tapizada con alfombras florales multicolores. Se realizan actividades culturales, ferias artesanales y agropecuarias, danzas, degustación de platos típicos y presentaciones de Caballo de paso peruano.

El presente trabajo está dirigido a los estudiantes del III ciclo de computación e informática del I.E.S.T Público "Perú – Japón", el local de la institución está ubicado en Jr. Amazonas Nº 120 – Chachapoyas.

El I.E.S.T Público "Perú – Japón" de Chachapoyas es una institución educativa estatal que forma profesionales técnicos de calidad, con ética en el desempeño laboral y responsabilidad demandadas en las diferentes carreras profesionales en condiciones de

interactuar en el ámbito nacional e internacional, contribuyendo al desarrollo sostenible del entorno socio – económico de la región y del país.

Fue creado el ***10 de enero de 1980 con Resolución Suprema N° - 017-80-ED*** y revalidado el ***15 de febrero del 2006 con Resolución Directoral N° - 0120-2006-ED***.

En el año ***1982 con R. M. N° 00026-82-ED***, asume la categoría de ESEP segundo ciclo e inclusive se incrementa otras áreas como: secretariado ejecutivo y de gerencia y tecnóloga sanitaria rural ofertando el título de especialista profesional. En el año de 1983, se promulga una nueva ley general de educación N° 23384, que estructura el sistema educativo peruano en etapas, dentro de los que considera a los Institutos Superiores Tecnológicos en la segunda etapa.

La ***ESEP*** segundo ciclo con fecha 13 de febrero de 1984 cambia su denominación según resolución ministerial N° 110-84-ED a Instituto Superior Tecnológico.

Adecuándose a dicho sistema el Instituto Superior Tecnológico "Perú -Japón" de conformidad a la Resolución Suprema N° 131-83-ED del 19 de marzo de 1983 y en este sistema se otorga el título a nombre de la nación como profesional técnico, mención que se otorga hasta la actualidad.

El Instituto Superior Tecnológico Público "Perú - Japón" fue revalidado por el ministerio de educación mediante ***R.D. N° 0120-2006-ED*** de fecha 15 de febrero del 2006.

El local actualmente cuenta con una infraestructura moderna de dos pisos, su área es de 500 m^2 contado con estudiantes de ambos sexos, divididos en dos turnos, por la mañana todos los primeros ciclos y por la tarde el tercer y quinto ciclo de las siete carreras; el promedio de estudiantes por aula es de 35. Cuenta con una plana docente prestigiosa los cuales han formado varias generaciones de profesionales que en la actualidad desempeñan cargos en diferentes entidades ya sean públicas o privadas.

Se ofertan siete carreras profesionales técnicas como: técnica en laboratorio clínico, enfermería técnica, mecánica automotriz, electrónica industrial, producción agropecuaria, secretariado ejecutivo y computación e informática con una población de 550 estudiantes, 58 docentes especialistas y 25 administrativos. Se trabaja con un diseño curricular básico, sistema modular donde los estudiantes después de haber culminado el primer y segundo

ciclo y haber realizado su práctica del primer módulo se certifican por lo cual están aptos para laborar.

La carrera profesional técnica de computación e informática es una carrera líder en la formación de profesionales técnicos en el dominio de las Tecnologías de información y comunicación, con sólidos valores humanísticos y ética al servicio de la sociedad. Profesionales formados a través del desarrollo del diseño curricular dividido en tres módulos profesionales como son:

Módulo I: Gestión de soporte técnico, seguridad y tecnologías de la información y comunicación. ***Módulo II:*** Desarrollo de software y gestión de base de datos. ***Módulo III***: Gestión de aplicaciones para internet y producción multimedia, además de participar en investigación e innovación y proyección social.

Cuenta con una infraestructura propia para cada carrera profesional técnica juntamente con laboratorios bien implementados con lo último en tecnologías de información especialmente la carrera de computación e informática.

- 10 proyectores multimedia (Hitachi, Epson).
- 85 PC de escritorio entre Core i3, i5.
- 3 laptops convencionales (Compaq).
- 2 pizarras digitales.
- Conexión a internet.

En este contexto donde se realizó este trabajo de investigación, el cual se enmarca dentro del plano didáctico, y de manera más específica dentro del proceso de enseñanza y aprendizaje de la lógica de programación. Se busca con esta investigación aportar a la mejora del proceso de enseñanza y aprendizaje de esta unidad didáctica dentro del nivel de educación superior técnica; para lo que se plantean diversas estrategias didácticas basadas en el uso del software PSeInt.

1.2. ENFOQUE HISTÓRICO TENDENCIAL DEL PROBLEMA

El problema materia de abordaje en este trabajo de investigación se ubica en el I.E.S.T. Público “Perú - Japón” de Chachapoyas, particularmente en los estudiantes del III ciclo de computación e informática, donde se ha observado un nivel muy bajo en el diseño y construcción de algoritmos en la unidad didáctica de lógica de programación.

El diseño y construcción de algoritmos, como capacidad básica a ser desarrollada por los estudiantes a partir de las áreas de lógica de programación, viene teniendo un tratamiento prioritario en el sistema educativo; debido a los cambios significativos que se vienen experimentando en el diseño curricular básico hasta la actualidad.

En el marco de esta nueva óptica con que se mira al diseño y construcción de algoritmos, se han desarrollado diversos estudios e investigaciones orientados a la solución de problemas concretos en esta capacidad, que dicho sea de paso no es una problemática nueva, ni en el instituto "Perú - Japón", ni en el contexto nacional.

En el instituto "Perú - Japón" de Chachapoyas, el problema en torno al diseño y construcción de algoritmos en la unidad didáctica de lógica de programación está presente en casi todos los ciclos del nivel técnico de computación e informática, acentuándose en uno más que en otros. Es particularmente en el III ciclo de computación e informática donde requiere urgente un tratamiento especial por ser la base de la lógica de programación.

En este ciclo, el problema se evidencia en la incapacidad para diseñar y construir algoritmos sencillos utilizando el software PSeInt, serias dificultades para organizar sus ideas y expresarlas por escrito de manera coherente y entendible, cometen frecuentes errores del uso de estructuras, tienen dificultad para expresar su lógica y empezar a programar en cualquier lenguaje de programación, temor a la unidad didáctica, así como dificultades para crear sus diagramas de flujo.

Se considera que estas deficiencias se deben fundamentalmente a la utilización por parte de los docentes de la unidad didáctica. De metodologías inapropiadas, así como recursos educativos que no motivan al estudiante para aprender y en consecuencia no se alcanzan los objetivos propios de la unidad didáctica. Por tal razón, en este trabajo de investigación se plantea la utilización del software PSeInt como alternativa de solución a la problemática encontrada.

Para facilitar el aprendizaje de la lógica de programación, así como también para asegurarse que los estudiantes aprenderán de una forma interesante y entretenida, se ha creado material didáctico para apoyar la enseñanza, entre estos materiales está el software PSeInt. Los docentes deberían usar esta nueva tecnología para presentarles a sus

estudiantes, nuevas formas de aprender un idioma ya que necesitan adaptarse a los cambios y aportes tecnológicos que la ciencia nos regala.

El hecho de querer entregar una mejor educación a los estudiantes y brindarles las herramientas necesarias para lograr el aprendizaje del diseño y construcción de algoritmos, nos llevan a descubrir uno de los factores que influyen en el aprendizaje de la lógica de programación.

De hecho, el Ministerio de Educación en estos últimos años está impulsando una mejor metodología para que los estudiantes sean parte de esta actividad, para ello está realizando capacitaciones de cursos y talleres en el uso de las TICs a los docentes de los institutos con el objetivo de lograr un aprendizaje significativo en los estudiantes.

Por estas razones se plantea en este trabajo de investigación la aplicación de estrategias didácticas con base en el uso del software PSeInt, con el fin de tomar más atractivas y motivadoras las clases para los estudiantes.

1.3. METODOLOGÍA

1.3.1. DISEÑO DE LA INVESTIGACIÓN

Para contrastar la hipótesis en este trabajo de investigación, se utilizó el diseño **cuasi experimental** de un solo grupo, con pre y post test. Vale decir que el grupo fue asignado en forma no aleatoria o al azar, sino que fue un grupo ya establecido, cuyo diagrama es el siguiente:

Grupo de estudio **O1** **X** **O2**

Donde:

O1 = observaciones o mediciones realizadas antes de aplicar el software PSeInt.

O2 = observaciones o mediciones realizadas después de aplicar el software PSeInt.

X = tratamiento experimental: software PSeInt.

El tipo de investigación es investigación **científica aplicada** en el nivel de producción de **tecnología sustantiva**, la cual se sustenta en las estrategias didácticas para facilitar el aprendizaje del diseño y construcción de algoritmos, además se presenta el diseño de la investigación y la relación entre sus variables dependiente e independiente y el análisis del resultado obtenido.

Los talleres se ejecutaron en las siete sesiones de clase en el laboratorio de cómputo, haciendo uso de computadoras, software y otros medios didácticos.

1.3.2. POBLACIÓN Y MUESTRA

1.3.2.1. POBLACIÓN: estuvo constituida por todos los 120 estudiantes de computación e informática del I.E.S.T. Publico "Perú - Japón" de Chachapoyas.

1.3.2.2. MUESTRA: estuvo constituida por 35 estudiantes del III ciclo de computación e informática, representando al grupo de estudio. La muestra fue no probabilística, se trabajó con el grupo intacto por conveniencia.

1.3.3. TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE DATOS

En este trabajo se utilizaron las siguientes técnicas e instrumentos:

- **Observación**: esta técnica permitió recoger información relevante a partir de la percepción, ilustrada y directa de los hechos. Se utilizó tanto la de tipo estructurada, mediante su correspondiente ficha o guía previamente elaborada, como la no estructurada, para hacer un reconocimiento de la caracterización de la problemática.
- **Guía de observación**: este instrumento sirvió para otra anotar los detalles de las variables, indicadores y subindicadores a observar.
- **Test (pre y post)**: para obtener datos con fines de diagnosticarla evaluación; debido a la cantidad y calidad de datos que proporciona, constituyo una herramienta indispensable en este trabajo de investigación.

1.3.4. MÉTODOS Y PROCEDIMIENTOS PARA LA RECOLECCIÓN DE DATOS

En este trabajo de investigación se utilizaron diversos métodos tanto teóricos como empíricos, lo que a continuación se detalla:

Los métodos empíricos: para diagnosticar o determinar la problemática dentro del instituto en cuyo espacio específico tuvo lugar esta investigación; pero también se utilizó en la fase de aplicación, vale decir en el trabajo de campo para

aplicar la variable independiente. Los métodos empíricos utilizados en este trabajo fueron: el experimento, la observación y la medición.

Los métodos teóricos: dentro del análisis teórico correspondiente se utilizó una metodología mixta, vale decir que se recurrió a diversos métodos, tales como: analístico – sintético, inductivo – deductivo, dialectico, critico, y holístico.

Para la recolección de datos se llevaron a cabo distintas acciones; tanto en lo que respecta al trabajo de gabinete, como en el de campo; a continuación se detalla:

a. TRABAJO DE GABINETE

- **Selección, lectura y análisis del material didáctico pertinente.** Considerando que existe gran cantidad de fuentes bibliográficas que guardan relación con la temática que se investigó, luego una revisión minuciosa, solamente se consideraron aquellas cuya relación fue más estrecha y que son más recientes. Se consideraron tanto las fuentes impresas como las virtuales.

- **Elaboración de instrumentos.** Se llevó a cabo con mucha minuciosidad, considerando las bases teóricas, la naturaleza del problema, las variables en estudio, así como los indicadores y subindicadores expresados en la operacionalización de las variables. Se puso especial atención en la elaboración del test para lo cual se recurrió al asesoramiento de expertos.

b. TRABAJO DE CAMPO

- **Aplicación de instrumentos.** Para recabar información de naturaleza facto perceptible se recurrió a la observación directa e intencional, utilizando para ello una guía de observación; la técnica y el instrumento se utilizó prácticamente durante todo el tiempo que duro la aplicación del software educativo, sin embargo se puso especial énfasis en el comienzo y final de dicha aplicación, esto con fines de diagnóstico y evaluación final respectivamente. Por su parte el test, se aplicó antes de incorporar la variable independiente (pre test) y al finalizar su aplicación (post test).

- **Aplicación del software:** Tuvo una duración de un mes y medio consecutivo, donde se realizó el control de la variable interviniente.

1.3.5. ANÁLISIS ESTADÍSTICO DE LOS DATOS

Para procesar estadísticamente el test, se elaboró una guía de evaluación e interpretación, a partir de las respuestas emitidas por los estudiantes, las que fueron agrupadas y luego convertidas a una escala de valoración.

Para el análisis de los resultados, luego de procesar la información obtenida mediante el test, se elaboraron cuadros estadísticos con frecuencias simples y porcentuales, orientadas a facilitar la interpretación en función al avance del grupo donde se aplicó la variable independiente (grupo de estudio).

Se compararon los resultados obtenidos mediante el pre test y el post test, a fin de determinar los logros obtenidos en el grupo de estudio.

CAPÍTULO II

Marco Teórico

CAPÍTULO II
MARCO TEÓRICO

2.1. ANTECEDENTES

(Álvaro Ignacio Morales Gonzáles, Jorge Luis Muñoz Montaño y Luis Eduardo Peláez Valencia, 2006) en su propuesta una experiencia pedagógica para programas de ingeniería: La Enseñanza de Algoritmos Mediada por la Lógica Cognitiva y la Elaboración de Juegos de Lógica, señala que la experiencia no pretende aportar en innovación tecnológica, sino contribuir desde la didáctica y la lúdica con estrategias para la enseñanza de la lógica y los algoritmos, áreas que exigen procesos de meta cognición en los que estudiantes y profesores están llamados a interpretar y proponer soluciones a problemas que implican el máximo desarrollo lógico.

Desde la modernidad, y en particular desde Hegel, es imposible pretender establecer un sistema de pensamiento totalizador de la realidad. En especial desde la economía, pero también desde la filosofía, la política y algunas "ciencias duras", se han intentado presentar "constructores teóricos" con pretensiones de universalidad, empero, las múltiples rupturas y contradicciones y las nuevas formas de ser "seres humanos" que ha traído la llamada "postmodernidad", ha llevado a comprender más que a aceptar, que se vive inmerso en pluralidad de paradigmas racionales y en "un sistema diverso" conformado, más que de leyes, de posibles; "un sistema" que se estructura desde la complejidad de pensamientos singulares (Garavito, 1990).

Bajo esta perspectiva, la educación se ve hoy entrecruzada, tal como plantea Morin (1998), por las "interretracciones" de la propia realidad y es imposible seguir pensando hoy en lógicas de única verdad, de único camino de solución puesto que las relaciones de los hombres se determinan ahora por lógicas diferenciadas en las que se busca, ante todo, el desarrollo de la creatividad de los actores inmersos en el proceso educativo.

Donald Knuth es uno de los más prestigiosos científicos en la ciencia de la computación. Ha escrito la serie de libros: The Art of Computer Programming, que es una referencia obligada a todo aquel que programe. Donald Knuth ha escrito hasta ahora cuatro libros de su serie sobre el arte de la programación:

Vol I: Algoritmos fundamentales (1986), Vol II: Algoritmos seminuméricos (1986), Vol III: Clasificación y Búsqueda (1986) y el Vol IV: Algoritmos combinatorios (1986).

(Omar Iván Trejos Barítica, 1999) en su libro La Esencia de la Lógica de Programación – Básico, recalca los conceptos fundamentales para aprender realmente a programar o sea lo que ha llamado la esencia de la lógica de programación, pues busca que las personas conozcan estos elementos conceptuales y, luego de dominarlos, se enfrente sin ningún problema no solo a cualquier objetivo que pueda ser alcanzable a través de computadores sino además a cualquier lenguaje de programación. Además, señala que los conceptos son sencillos para el diseño de soluciones muy (pero muy) lógicas.

(Nieva García, Omar Santiago; Arellano Pimentel, J. Jesús, 2009) en su documento Método de Enseñanza de Algoritmos Centrado en 2 Dimensiones, señalan que su trabajo se centra en dos dimensiones importantes del aprendizaje, por un lado, la capacidad de abstracción (no solamente enseñarle a escribir algoritmos) y por otro lado la capacidad de resolución de problemas. Para poder desarrollar ambas dimensiones, se hace uso del paradigma de enseñanza constructivista y de la heurística de resolución de problemas matemáticos de Polya.

(M.C. Ricardo Vargas de Basterra, 2006) en su tesis Modelo para la Construcción de Algoritmos Apoyados en Heurísticas, este modelo se enfoca en la parte microscópica del área de ingeniería de software concentrándose específicamente en el análisis, diseño, construcción, pruebas y validación de los algoritmos inmersos en los métodos y funciones de las clases y otros componentes de software abordando la capa más específica y concreta de la producción de software.

Tesis de maestría de la Universidad de Guayaquil, de Ortega Washington presentada en el mes de julio del 2012, cuyo título es: “El uso de la tecnología de información y comunicación en el aprendizaje significativo de los estudiantes”. El autor sostiene que el uso de la tecnología en la actualidad es casi imprescindible en todos los campos de la ciencia, no solo en el campo de la educación, convirtiéndose en una herramienta importante para cada uno de los profesionales.

2.2. BASE TEÓRICA

2.2.1. Teorías del Aprendizaje Aplicadas a las TIC

2.2.1.1. Teoría de PIAGET

La epistemología genética piagetiana se basa en el conocimiento del mundo a través de los sentidos, atendiendo a una perspectiva evolutiva.

El desarrollo de la inteligencia es la adaptación del individuo al medio, y en este desarrollo destacan 2 procesos básicos: a) adaptación (entrada de información), y b) organización (estructuración de la información).

Describe tres estadios básicos de desarrollo: sensomotor, operaciones concretas y operaciones formales.

Aboga por secuencias de instrucción con las siguientes características:

a. Debe ir ligada al desarrollo del individuo
b. Debe ser flexible
c. Debe considerar el aprendizaje como un proceso
d. La actividad tiene un papel relevante
e. Los medios deben estimular el aprendizaje
f. Hay que considerar la influencia del ambiente

Realmente, Piaget no fue muy partidario de la instrucción por ordenador, pero influyo en el constructivismo de PAPERT.

2.2.1.2. Aprendizaje Significativo de Ausubel

El aprendizaje significativo (en contra del memorístico o mecánico) es aquel en el que el contenido debe incorporar el conocimiento del sujeto en relación a aprendizajes previos. Influye en el diseño de software con limitaciones, ya que aunque reconoce como eficaz la Enseñanza Asistida por Ordenador (EAO), piensa que es mejor una enseñanza programada mediante libros. Critica la fragmentación de contenidos que puede darse en la EAO, y aboga por la necesidad del profesor como guía. Influirá en la teoría de la información de Gagné.

Ausubel señala que las TIC son medios eficaces para el proceso de enseñanza y aprendizaje.

Esta afirmación se debe a que a través del ordenador se pueden hacer simulaciones, se puede rescatar los saberes previos, se propicia el descubrimiento; sin embargo todo esto no puede reemplazar a la realidad.

2.2.1.3. Teoría del Conocimiento Operante de SKINNER

SKINNER fue pionero en la utilización de ordenadores para la enseñanza programada. Este proceso consistía en la presentación secuencial de preguntas y en la sanción correspondiente a las respuestas de los estudiantes.

Brunner afirma que se debe ayudar a los estudiantes a pasar progresivamente de un pensamiento concreto a un estadio de representación conceptual o simbólica más adecuada al pensamiento.

En esta realización puede resultar de gran utilidad el ordenador y algunos programas diseñados especialmente para este fin.

Se trataba de programas de ejercitación y práctica, basados en la repetición con secuencias de materiales lineales, y sanciones positivas o negativas a las respuestas del estudiante.

2.2.1.4. Aprendizaje por Descubrimiento de BRUNER

Los recursos TIC constituyen condiciones externas que pueden afectar positivamente las condiciones internas del estudiante para un buen aprendizaje.

Afirma que la resolución de problemas depende de cómo se presentan, de que supongan un reto que incite a su resolución, y propicie la transferencia.

Propone un currículo en espiral, que debe girar en torno a los grandes problemas, principios y valores de la sociedad.

En cuanto a su influencia en el software educativo, propone la estimulación cognitiva mediante materiales que entrenen las operaciones lógicas básicas.

Aboga por la creación de secuencias instructivas con las siguientes características:

- Hay que disponer la secuencia de forma que se pueda apreciar la estructura.
- Tiene que promover trasferencias.
- Hay que utilizar el contraste
- Se debe ir de lo concreto a lo abstracto
- Debe posibilitarse la experiencia de los estudiantes
- Se han de hacer revisiones periódicas de los conceptos aprendidos (currículo en espiral).

Respecto al proceso de enseñanza:

- Debe ser capaz de captar la atención
- Se debe analizar y presentar la estructura del contenido de forma adecuada.
- Es importante que el estudiante describa por sí mismo lo que es relevante para resolver un problema.
- Es esencial elaborar una secuencia efectiva.
- El refuerzo y la retroalimentación surgen del éxito.

2.2.1.5. Constructivismo de PAPERT

Papert intenta que mediante la computadora el niño pueda llegar a hacerse planteamiento acerca de su propio pensamiento, tarea difícilmente realizable sin su concurrencia.

Papert es el creador LOGO, primer lenguaje de programación para niños. Este sirve para que, mediante la programación, el niño piense sobre sus procesos cognitivos, sobre sus errores, y los aproveche para reformular sus programas, por lo que la programación servirá para favorecer las actividades meta cognitivas.

Considera que el ordenador reconfigura las condiciones de aprendizaje, y supone nuevas formas de aprender.

Parte de los postulados piagetianos que entienden el sujeto como agente activo y constructivo del aprendizaje, pero le da un carácter más intervencionista, incidiendo en las estructuras mentales potenciales, y en los ambientes de aprendizaje.

Aún con diferencias, toma de Piaget:

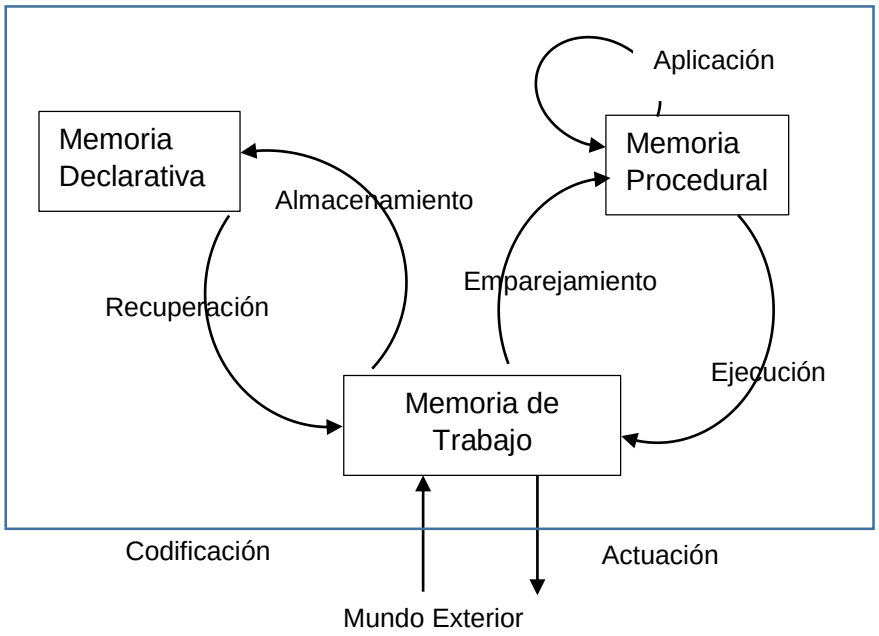
- La necesidad del análisis genético de los contenidos
- La defensa constructivista del conocimiento
- La defensa del aprendizaje del sujeto como ser activo que construye sus teorías sobre la realidad interactuando con esta
- La confrontación de las teorías con los hechos (conocimiento y aprendizaje frutos de la interacción entre sujeto y entorno).

Internet presenta rasgos de un entorno de aprendizaje constructivo en cuanto que permite la puesta en juego de los principios constructivistas. Es un sistema abierto guiado por el interés, iniciado por el aprendiz, e intelectual y conceptualmente provocador. La interacción será atractiva en la medida en que el diseño del entorno es percibido como soportador del interés.

2.2.1.6. Teoría del Control Adaptativo del Pensamiento (ACT) de Anderson

Es una teoría unitaria de procesamiento de información, ya que la idea básica que subyace a la teoría es que, según Anderson "todos los procesos cognitivos superiores, como memoria, lenguaje, solución de problemas imágenes, deducción e inducción son manifestaciones diferentes de un mismo sistema subyacente. Por tanto, los mecanismos de aprendizaje están en el CAT estrechamente relacionado con el resto de los procesos, especialmente con la forma en que se presenta la información en el sistema." (Pozo, 2006).

La teoría del control Adaptativo del Procesamiento de Anderson está totalmente traducida a un programa de computador la cual incorpora una teoría del pensamiento y su esquema se plantea en la figura 1, aquí se presentan tres tipos de memoria relacionadas: Memoria de trabajo, memoria declarativa y memoria procedural.



Memoria de trabajo. Es la que se comunica con el mundo exterior y procede por estímulos externos o por el propio sistema como consecuencia de la ejecución de una acción. Esta memoria se mueve con el concepto de Activación de un nodo que es central dentro de la teoría del ACT.

Activación. Hace referencia al proceso de como un nodo accede a la memoria de trabajo la cual puede producirse por estímulos externos o del propio sistema como consecuencia de la ejecución de una acción.

Según Anderson, "La activación cumple en el ACT la función heurístico asociativo relevante. Es decir, la activación mide lo estrechamente asociada que una pieza de información está con respecto a la información que actualmente usa" (Pozo, 2006) y es por eso que los nodos presentes en la memoria declarativa más frecuentemente usados son los que tienen mayor probabilidad de estar activos en la memoria de trabajo y la cantidad de ellos activos son limitados porque ésta memoria también es limitada.

Memoria declarativa. Almacena el conocimiento descriptivo sobre el mundo. Es la que determina el "SABER QUE". Se establece en ésta, como está organizado el mundo y lo que en él sucede. Características:

- Nunca desencadena acciones sobre el mundo directamente.
- Puede llegar a activar el conocimiento procedural responsable de esas acciones.
- Está organizada en forma de Red Jerárquica compuesta según Anderson de "Unidades Cognitivas o nodos y eslabones entre esos nodos" (Pozo, 2006). La memoria declarativa se asemeja a los modelos de memoria semántica.
- El conocimiento declarativo es estable y normalmente inactivo. Solo los nodos que se encuentran en la memoria de trabajo tendrán influencia sobre el conocimiento procedural.

Memoria procedural o de producciones. Contiene información para la ejecución de las destrezas que posee el sistema, representa el "SABER COMO" y se basa en sistemas de producción desarrollados por Newell y Simon en 1972. "En ésta, la idea básica de estos sistemas es que el conocimiento se almacena en forma de producciones o pares condición-acción. Las producciones adoptan la forma de un condicional <<si… entonces…>>…" (Pozo, 2006)

Ejemplo. Se plantea una situación en donde en la memoria de trabajo se tiene determinado el concepto de tigre, lo cual es una manifestación de la memoria declarativa que está activa en la memoria de trabajo.

Condición: Si aparece un tigre

Acción: Salir corriendo

Cuando el conocimiento declarativo activo en la memoria de trabajo satisfaga o "se empareje" con la condición de una producción, podrán activarse aquellas producciones cuyas condiciones estén parcialmente satisfechas. Tanto en el emparejamiento total como parcial, las producciones existentes en la memoria procedural, al igual que sucede con los nodos declarativos coinciden entre sí por ser activadas. Cuanto más frecuentemente se active una producción, mayor será su fuerza asociada. Entre más fuerte, se tiene mayor probabilidad de activación.

2.3. MARCO CONCEPTUAL

2.3.1. FUNDAMENTOS Y ENFOQUE DE LA LÓGICA DE PROGRAMACIÓN

- ***Acerca de la lógica***

Siempre que usted vaya a resolver un problema sea muy lógico, esto quiere decir que sencillamente guíese por sus mínimos razonamientos y busque siempre el camino más obvio y sencillo. No existe un problema que se resuelva con lógica cuya solución no sea sencilla. Antes de comenzar a pensar en la lógica de programación piense en su propia lógica. Diseñe las soluciones pensando en sus propias reglas y luego si ajústese a las reglas que la lógica de programación impone para facilitar la posterior codificación.

Es muy importante que usted poco a poco destine parte de su tiempo a resolver problemas, así no sean estrictamente de programación, dado que en esos momentos en donde usted se sienta a pensar detenidamente en la búsqueda de solución de un determinado problema, en esos momentos es donde usted realmente está utilizando su cerebro. Normalmente nuestro cerebro se va acostumbrando, y así es como lo orienta la educación formal, a buscar entre sus conocimientos las soluciones que se estén necesitando pero muchas veces tenemos que crear soluciones y es allí en donde nos encontramos que son muy pocas las veces en las que ponemos a funcionar nuestro cerebro.

- ***Algoritmo***

Un algoritmo es un conjunto ordenado y finito de operaciones que permite hallar la solución de un problema. Según Jahnsonbaugh es un conjunto finito de instrucciones que tienen las siguientes características:

(a) **Precisión**: todos los pasos son descritos con exactitud y sin ambigüedad.

(b) **Unicidad**: los resultados intermedios de la ejecución de cada paso son únicamente definidos y dependientes de los resultados y las entradas de los pasos anteriores.

(c) **Finito**: el algoritmo se detiene después de haber sido ejecutado varias instrucciones.

(d) **Entrada**: el algoritmo recibe entradas.

(e) **Salida**: al algoritmo produce salidas.

- ***Programación***

Levine (66) señala, "por programar se entiende un proceso mental complejo dividido en varias etapas". La finalidad de la programación, así entendida, es comprender con claridad el problema que va a resolverse o simularse por medio de la computadora, y entender también con detalle cual será el procedimiento mediante el cual la máquina llegará a la solución deseada. Para efectos de este proyecto nos centramos en las cuatro etapas que son el proceso de identificación del problema, el diseño algorítmico de la solución, la codificación en algún lenguaje y las pruebas correspondientes.

- ***La programación como un arte***

Según Budd (17), "es cierto que programar un computador es todavía una de las tareas más difíciles jamás emprendidas por la humanidad; llegar a ser un experto en programación requiere talento, creatividad, inteligencia, lógica, habilidad para construir y usar abstracciones, y experiencia, aun cuando se disponga de las mejores herramientas". Por otra parte Glenn (45) comenta "el proceso de descubrimiento de algoritmos y de resolución de problemas tienen aún íntima relación". En última instancia, nos gustaría reducir el mismo proceso de resolución de problemas a un algoritmo, pero se ha demostrado que esto es imposible. Así, la capacidad para resolver problemas sigue siendo más una aptitud artística que debe desarrollarse, que una ciencia precisa por aprender"

KUNTH (61), quizás uno de los programadores más notables, nombró a su libro el ARTE DE LA PROGRAMACIÓN subrayando que a pesar de todos los formalismos que se han creado, el diseño de algoritmos y su codificación sigue siendo un arte.

- ***Pseudocódigo***

El pseudocódigo es un lenguaje de alto nivel empleado para describir algoritmos, igual que los diagramas de flujo, aunque más cercano a la

computadora. El objetivo último del pseudocódigo es facilitar la comprensión del algoritmo.

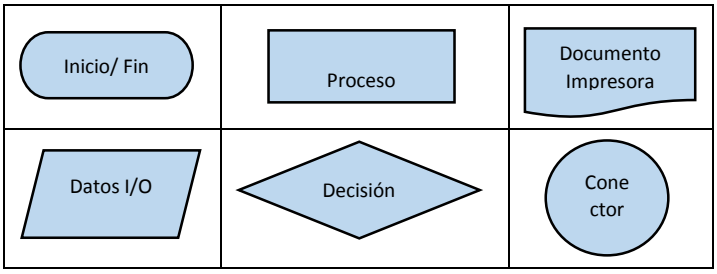
Concretamente, el pseudocódigo describe un algoritmo utilizando una mezcla de frases en lenguaje común, instrucciones de programación y palabras clave que definen las estructuras básicas.

- ***Diagramas de Flujo***

Los diagramas de flujo u organigramas son esquemas que se emplean para presentar gráficamente un algoritmo. Se basan en la utilización de diversos símbolos convenidos para representar operaciones específicas.

Los diagramas de flujo son representaciones gráficas del pseudocódigo. Los diagramas de flujo se dibujan generalmente usando símbolos estándares.

Los símbolos más comunes se muestran a continuación:



- ***Unidad Didáctica de Lógica de Programación***

La competencia de la unidad es: Planificar, implementar y gestionar el uso de las tecnologías de información y comunicación de una organización, a partir del análisis de sus requerimientos, teniendo en cuenta los criterios de calidad, seguridad y ética profesional propiciando el trabajo en equipo.

Su capacidad es realizar la lógica de la programación considerando procesos y secuencias; además están sus criterios de evaluación; realiza algoritmos de procesos y subprocesos, elabora y comprueba diagramas de flujo de los procedimientos requeridos. MINEDU – Diseño Curricular Básico (2010).

Esta unidad tiene como finalidad desarrollar en el estudiante la lógica de programación para resolver problemas a través de pseudocódigo y diagramas de flujo.

2.3.2. ASPECTOS GENERALES DE LAS TIC EN EDUCACIÓN

En la actualidad, nuestra sociedad se encuentra inmersa en la llamada era de la postmodernidad, caracterizada por la sucesión de continuos cambios sociales, económicas, políticos, culturales, ideológicas, tecnológicos, que plantean nuevas exigencias, demandas y desafíos al ámbito educativo, lo cual hace necesario redefinir el papel del profesorado, como piedra angular de la calidad de la educación, y a su vez, el planteamiento de nuevos enfoques metodológicos, herramientas y estrategias didácticas que den protagonismo al estudiante como sujeto activo, participativo, autónomo, creativo y reflexivo en la construcción de su propio conocimiento.

En este nuevo escenario, las Tics juegan un papel preponderante, ellas están presentes en las aulas, emergiendo nuevos modelos de aprendizaje en el estudiantado; en este contexto el profesorado debe asumir un nuevo rol como guía, mediador, facilitador de ese aprendizaje constructivo en el discente, creando el ambiente propicio para que se genere el aprendizaje significativo, relevante y funcional.

Tal como señala Moreno (2011),las Tics ofrecen posibilidades educativas sorprendentes para hacer posible un aprendizaje constructivo, participativo, activo, autónomo, creativo y reflexivo favoreciendo actitudes de búsqueda, investigación, exploración, descubrimiento, comunicación, intercambio y colaboración dentro de un nuevo modelo educativo en el que el discente cobra mayor importancia en el proceso educativo. Y de este modo, a través de las Tics, son posibles los aprendizajes colaborativos aprovechando la inteligencia colectiva.

Por tanto, partiendo de la idea de Castaño, Maíz, Palacio y Villarroel (2008), la misión del profesorado en entornos virtuales de enseñanza y aprendizaje es moderar y mantener vivos todos los espacios comunicativos facilitando el acceso a los contenidos haciendo posible el dialogo entre los participantes, el intercambio o confrontación entre puntos de vista y la reflexión, favoreciendo de este modo en los estudiantes, ese proceso de transformación de la información en conocimiento, haciendo posible, a su vez, esa construcción de la autonomía intelectual atendiendo a los postulados propuestos por Vygotsky sobre el aprendizaje sociocultural, así como la regulación reciproca que se produce entre los participantes.

EL SOFTWARE EDUCATIVO COMO APOYO AL PROCESO DE ENSEÑANZA APRENDIZAJE

Actualmente los cambios permanentes que se vienen dando en la educación, la información y en los software educativos, obligan a los docentes a nivel mundial a actualizarse permanentemente a fin de guiar y orientar a los estudiantes, acorde con los adelantos científicos y tecnológicos, de tal manera que les permita mejorar el proceso de enseñanza aprendizaje, logrando que los estudiantes desarrollen aprendizaje duraderos.

¿Pero que es un software educativo?, ¿Cuáles son sus funciones?, ¿Qué repercusiones tienen estos recursos en el campo educativo? Para dar respuesta a estas, entre otros interrogantes, a continuación se anotan de manera precisa algunos conceptos e ideas vertidas por diferentes académicos en torno al asunto en estudio.

A. SOFTWARE EDUCATIVO

Según Marqués (1997) los software educativos, son programas educativos y ordenador creados con la finalidad de ser utilizados, para facilitar los procesos de enseñanza y aprendizaje (medios didácticos).

Gros (2001) considera que el software educativo es cualquier producto basado en computadora con una finalidad educativa; mientras que Galvis (2000), sostiene que en el campo educativo se suele denominar software educativo a aquellos programas que permiten cumplir y apoyar funciones educativas.

En esta categoría entran tanto los que dan soporte al proceso de enseñanza y aprendizaje (un sistema para enseñar matemáticas, ortografía, contenidos o ciertas habilidades cognitivas), como los que apoyan la administración de procesos educacionales o de investigación.

Por ultimo Cataldi y García (2003) sostienen que el software educativo está relacionado con su grado de adaptación a un contexto en particular donde convergen una serie de variables, tales como las características curriculares, las edades del grupo destinatario, el estilo de aprendizaje y de enseñanza que se requieren para el análisis correcto.

B. FUNCIONES DEL SOFTWARE EDUCATIVO

Marques (1997) señala, que los software educativo realizan las funciones básicas propias de los medios didácticos en general y además, en algunos casos, según la forma de uso que determina el profesor, pueden proporcionar funcionalidades específicas. Las ventajas e inconvenientes que pueda comprobar su uso serán el resultado de las características del material, de su adecuación al contexto educativo al que se aplica y de la manera en que el profesor organice su utilización. Las funciones que puede realizar son: informativa, motivadora, evaluadora, instructiva, investigadora, expresiva, lúdica e innovadora.

C. SOFTWARE LIBRE

Da Rosa, y Heinz (2007) define al software o programa de computación que se refiere a la libertad de los usuarios para ejecutar, distribuir, copiar, cambiar, estudiar y mejorar el software. De modo más preciso, se refiere a cuatro libertades de los usuarios del software:

- La libertad de usar el programa, con cualquier propósito (libertad 0).
- La libertad de estudiar cómo funciona el programa, y adaptarlo a tus necesidades (libertad 1). El acceso al código fuente es una condición previa para esto.
- La libertad de distribuir copias, con lo que puedes ayudar a tu vecino (libertad 2).
- La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie, (libertad 3). El acceso al código fuente es un requisito previo para esto.

D. SOFTWARE LIBRE EN LA EDUCACIÓN

Según Da Rosa y Heinz (2007) sostienen que existen muchos motivos por los cuales es importante usar software libre en la educación. En opinión de los autores, tal vez lo fundamental sea tener en cuenta al estudiante como un ser integral y enseñarle la importancia de la libertad, guiarlo en el sentido de saber hacer uso de esa libertad.

En forma de conclusión, cabe señalar que sin duda el ordenador constituye hoy en día una herramienta de trabajo más para cualquier estudiante. Las tecnologías de la información y la comunicación (Tic) se han integrado en el ámbito

escolar y en los hogares como un instrumento complementario a los tradicionales canales de aprendizaje.

E. EL SOFTWARE PSEINT

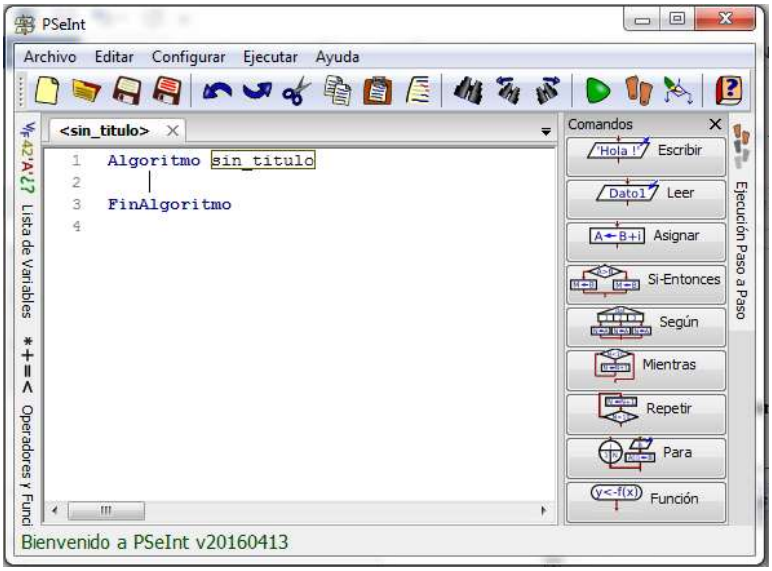
PSeInt es la abreviatura de los estados de computación de **Pse**udo **Int**erprete con licencia GNU, una herramienta educativa creada en Argentina, utilizada para aprender los fundamentos de la programación y el desarrollo de la lógica.

Fue lanzado en el 2003, desarrollado por Pablo Novara.

Sus características son:

- Lenguaje autocompletado
- Plantillas de comandos.
- Soporta procedimientos y funciones
- Indentado inteligente.
- Graficado, creación y edición de diagramas de flujo.
- Software multiplataforma.

Entorno gráfico del PSeInt



ACTIVIDADES QUE SE TRABAJAN CON PSeInt

PSeInt permite la creación de tres actividades diferentes para reforzar temáticas escolares.

Las actividades son:

1. Escritura de algoritmos.

2. Ejecución de algoritmos.
3. Dibujo del diagrama de flujo.

En conclusión, el software PSeInt cumple la mayoría de los requisitos para convertirse en una herramienta fundamental e indispensable en la enseñanza a todos los niveles.

La elección de PSeInt para este trabajo de investigación, se hizo intencionadamente como una manera de fomentar e integrara el uso de las nuevas tecnologías en el aula; pero además las ventajas significativas que ofrece han tenido cierto peso a la hora de optar por este programa.

Por el hecho de ser software de libre uso, de presentar una interfaz sencilla y fácil de aprender por un usuario con limitados conocimientos informáticos, lo que significa que los estudiantes del III ciclo de computación e informática no tendrán inconvenientes para adecuarse a su uso y manejo; el hecho de que su interfaz permite trabajar al estudiante de manera rápida y cómoda.

2.3.3. ALCANCES TEÓRICOS GENERALES RESPECTO A ESTRATEGIAS DIDÁCTICAS

La palabra estrategia deriva del latín ***Strategia***, que a su vez procede de dos términos griegos: ***stratos*** (“ejercito”) y ***agein*** (“conductor”, “guía”) por lo tanto, el significado primario o etimológico de estrategia es el arte de dirigir las operaciones militares. Sin embargo este concepto ha ido evolucionando a través del tiempo y hoy en día se aplica en diferentes contextos más allá del ámbito del tiempo; aunque su incorporación a nuestro lenguaje común es relativamente reciente, hasta hace algo más de medio siglo, su uso se restringía al campo militar y al mundo de la diplomacia.

En la actualidad el concepto estrategia se utiliza de manera general para referirse al plan ideado para dirigir un asunto y para designar al conjunto de reglas que aseguran la decisión óptima en cada momento.

En otras palabras, una estrategia es el proceso seleccionado a través del cual se prevé alcanzar un cierto estado futuro.

Según Gonzales (2010) las estrategias son formas específicas de organizar nuestros recursos (pensamientos, habilidades, tiempo, sentimientos, acciones) para obtener resultados consistentes al realizar algún trabajo. Añade que las estrategias siempre están orientadas hacia una meta positiva.

La estrategia supone dar respuestas a tres cuestiones básicas: qué, cómo y cuándo; en primer lugar, hay que responder a la cuestión del qué: que se pretende conseguir, cual es la meta que se persigue. En segundo lugar, se debe dar respuesta al cómo: cuáles serán los medios o acciones que permitirán alcanzar la meta. Finalmente, se ha de contestar al cuándo: en qué momento se llevaran a cabo las acciones y el periodo que supondrá realizarlas.

Estas tres preguntas se formulan siempre que una persona va a adoptar una decisión o llevar a cabo cualquier actividad. Sin embargo, no todas las decisiones pueden considerarse estratégicas.

En síntesis, una estrategia, de modo general, se puede definir como un modelo o plan que integra los principales objetivos, políticas y sucesión de acciones de una organización o de un proceso cualquiera en un todo coherente. Una estrategia bien formulada ayuda a ordenar y asignar los recursos de una organización, de una forma singular y viable.

Las estrategias didácticas se definen como un conjunto de procedimientos dirigidos a un objetivo determinado; el aprendizaje significativo de los estudiantes. Así pues este tipo de estrategias se orienta al logro de los propósitos planteados dentro del proceso enseñanza y aprendizaje, para lo cual se tienen que desarrollar diversas acciones debidamente planificadas y organizadas de conformidad con la naturaleza de la unidad didáctica, las características de los estudiantes y la concepción de aprendizaje.

Díaz y Hernández (1999) definen las estrategias didácticas como un conjunto de acciones dirigidas a la concesión de una meta, implicando pasos a realizar para obtener aprendizajes significativos, y así asegurar la concesión de un objetivo: toma en cuenta la capacidad de pensamiento que posibilita el avance en función de criterios de eficacia.

De otro lado las estrategias didácticas, específicamente las de aprendizaje implican más tiempo que los métodos tradicionales, una disposición ambiental en términos de mobiliario, materiales y de manera particular un trabajo más autónomo y por tanto con autorregulación del proceso, tarea nueva para el estudiante que posee una disposición natural hacia la clase magistral que implica solo trabajo del docente o la tradicional enseñanza.

En con conclusión cabe señalar que las estrategias didácticas buscan hacer más viable, entretenido y eficaz el proceso enseñanza aprendizaje, de tal manera que se alcancen los objetivos propuestos en la clase. Para ello se requiere la participación creativa del docente para diseñar y aplicar las estrategias; pero también la participación activa del estudiante como protagonista en el proceso docente educativo y responsable último de du aprendizaje.

Así pues las estrategias que se plantean en este trabajo de investigación, se orientan a la mejora del proceso enseñanza aprendizaje de la unidad didáctica de lógica de programación; en consecuencia el estudiante pueda lograr aprendizajes significativos en términos de desarrollo de sus capacidades, que le permitan ser competente no solamente en el aula sino fundamentalmente en la vida cotidiana.

CAPÍTULO III

Resultados de la Investigación y Propuesta Teórica

CAPÍTULO III
RESULTADOS DE LA INVESTIGACIÓN Y PROPUESTA TEÓRICA

3.1. ANÁLISIS E INTERPRETACIÓN DE DATOS

3.1.1. Resultados en relación a la aplicación de estrategias didácticas en el diseño y construcción de algoritmos, a partir del uso del software PSeInt.

En relación a las estrategias didácticas que utilizan los estudiantes para el diseño y construcción de algoritmos, a partir del uso del software PSeInt, y que en este estudio constituye un elemento importante de medición, cuyos resultados se presentan en el cuadros 1, hay que señalar que utilizando PSeInt se pueden plantear y aplicar diversas estrategias didácticas. A continuación se dan alcances generales para entender en qué consisten: sin embargo, más adelante en el acápite correspondiente a la presentación de la propuesta se describen y explican de manera detallada cada una de las estrategias desarrolladas e este estudio.

Una de estas estrategias es el diseño y construcción de algoritmos a partir de ***videos mostrados***, donde los estudiantes puedan entender el diseño de algoritmo, explicar cuáles son los pasos a seguir, describir la lógica empleada, qué estructuras se emplean en la solución del problema, etc. Todos estos videos deben almacenarse en una carpeta para ser compartidos y escritos en el software PSeInt.

De igual manera se puede trabajar con la estrategia en el diseño y construcción de algoritmos a partir de ***software libre***, consiste en usar el software libre como: LPP, NPP, etc. específicamente para la programación de algoritmos.

También se puede trabajar la estrategia en el diseño y construcción de algoritmos en relación a las ***dinámicas de grupo***; es decir se opta por trabajar en grupos de trabajo, por ejemplo "dinámicas para la reflexión grupal e individual: el dado, supervivencia en los andes", "dinámicas de confianza: el estatua, los sonidos de los animales", "dinámicas de trabajo en equipo: técnica 6.3.5, el globo aerostático, cosas en común".

Otra estrategia en el diseño y construcción de algoritmos a partir de ***ayudantes de catedra***; se elige un alumno que ayude en el asesoramiento durante la duración de la clase.

De igual manera se puede trabajar con la estrategia en el diseño y construcción de algoritmos a partir de ***material didáctico***, se entrega el material físico o digital para guiar la resolución de algoritmos.

También se puede trabajar la estrategia en el diseño y construcción de algoritmos en relación a un ***tema específico planteado***; es decir se opta por un tema que los estudiantes ya conocen, por ejemplo: "diseñar un algoritmo que realice las cuatro operaciones básicas", frente a este tema los estudiantes deben pensar lógicamente y escribir el pseudocódigo en el software PSeInt.

Por ultimo otra estrategia, que se plantea en este estudio como elemento de medición es el diseño y construcción de algoritmos a partir del ***desarrollo de actividades propias del programa***; cabe señalar que en PSeInt se pueden desarrollar muchas actividades como: escribir el algoritmo en pseudocódigo, ejecutar el programa, corregir errores y generar el diagrama de flujo.

A continuación se presentan el Cuadro 1, cuyos resultados se orientan al objetivo de determinar si los estudiantes aplican o no estrategias didácticas para diseñar y construir algoritmos, a partir del uso del ***software PSeInt***.

Cuadro 1:

Relación de estrategias didácticas contenidas en la Guía de observación, en el diseño y construcción de algoritmos por parte de los estudiantes, antes de su enseñanza.

Estrategias	Grupo de estudio					
	Si		No		Total	
	Nº	%	Nº	%	Nº	%
Diseña y construye algoritmos a partir de videos mostrados.			35	100	35	100
Diseña y construye algoritmos a partir de software libre.			35	100	35	100
Diseña y construye algoritmos a partir de dinámicas de grupo.			35	100	35	100
Diseña y construye algoritmos a partir de ayudantes de catedra.			35	100	35	100
Diseña y construye algoritmos a partir de material didáctico.			35	100	35	100
Diseña y construye algoritmos a partir de tema específico planteado.			35	100	35	100
Diseña y construye algoritmos a partir de desarrollo de actividades propias del programa.			35	100	35	100

***Fuente: Guía de observación a los estudiantes del III ciclo de computación e informática del I.E.S.T. Público "Perú - Japón"* – Chachapoyas.**

Se observa en el grupo de estudio, que los estudiantes, no conocen ni aplican ninguna estrategia para el diseño y construcción de algoritmos que tenga como base el uso del software PSeInt, ya que el 100% de los estudiantes no responden de manera positiva a los indicadores planteados.

Estos resultados ponen en evidencia que las estrategias planteadas en este trabajo de investigación para el diseño y construcción de algoritmos, teniendo como base el uso del software PSeInt, y que más adelante han sido aplicadas al grupo de estudio, han sido originales y novedosas para los estudiantes, elemento que ha servido de soporte para desarrollar la competencia del diseño y construcción de algoritmos.

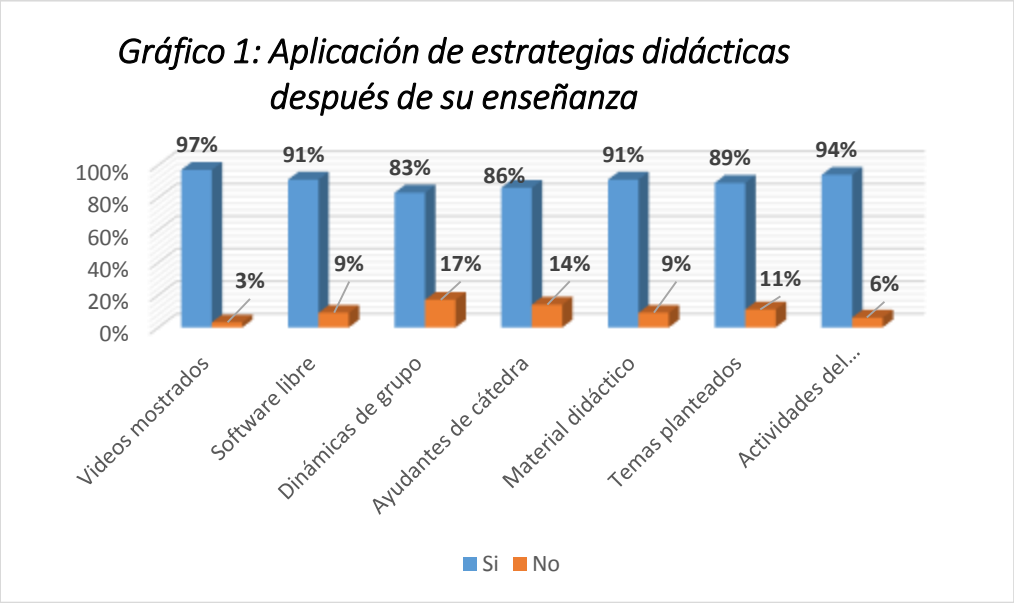
Luego de que los estudiantes del grupo de estudio pasaran por un proceso de enseñanza aprendizaje, en relación a las estrategias didácticas para el diseño y construcción de algoritmos a partir del uso del software PSeInt, los resultados varían significativamente en dicho grupo, tal como se observa en el Cuadro 2.

Cuadro 2:

Aplicación de las estrategias didácticas, en el diseño y construcción de algoritmos por parte de los estudiantes, después de su enseñanza.

Estrategias	**Grupo de estudio**					
	Si		**No**		**Total**	
	Nº	**%**	**Nº**	**%**	**Nº**	**%**
Diseña y construye algoritmos a partir de videos mostrados.	34	97	1	3	35	100
Diseña y construye algoritmos a partir de software libre.	32	91	3	9	35	100
Diseña y construye algoritmos a partir de dinámicas de grupo.	29	83	7	17	35	100
Diseña y construye algoritmos a partir de ayudantes de catedra.	30	86	5	14	35	100
Diseña y construye algoritmos a partir de material didáctico.	32	91	3	9	35	100
Diseña y construye algoritmos a partir de tema específico planteado.	31	89	4	11	35	100
Diseña y construye algoritmos a partir de desarrollo de actividades propias del programa.	33	94	2	6	35	100

***Fuente: Guía de observación a los estudiantes del III ciclo de computación e informática del I.E.S.T. Público "Perú - Japón"* – Chachapoyas.**



***Fuente: Guía de observación a los estudiantes del III ciclo de computación e informática del I.E.S.T. Público "Perú - Japón"* – Chachapoyas.**

Se observa en el Cuadro 2 y Gráfico 1 que, el 100% de los estudiantes del grupo de estudio, de conformidad con los resultados, evidencian un dominio significativo de dichas estrategias; ya que entre el 83% y 94% de estudiantes de este grupo, responden de manera positiva a los indicadores planteados, y solo un mínimo porcentaje que oscila entre el 3% y 17% no evidencia un manejo eficiente de dichas estrategias.

Se deduce a partir de los resultados, que el avance evidenciado en los estudiantes del grupo de estudio, se debe a la aplicación de las estrategias didácticas para diseñar y construir algoritmos, lo que se constituye a su vez en un indicador positivo del logro del objetivo general y la hipótesis planteados en este trabajo de investigación.

3.1.2. Resultados en relación al desarrollo de la capacidad del diseño y construcción de algoritmos.

Para el objetivo de determinar el nivel de desarrollo de la capacidad del diseño y construcción de algoritmos en la unidad didáctica lógica de programación, en los estudiantes del III ciclo de computación e informática del I.E.S.T. Publico "Perú - Japón" – Chachapoyas, teniendo en cuenta los indicadores para medir el nivel de calidad del software, así como los tipos de algoritmos diseñados y construidos, se presentan los Cuadros 3 y 4.

Los indicadores planteados en este estudio, para medir el nivel de calidad del software diseñado por los estudiantes se dio en seis indicadores como: la ***funcionalidad***, es decir la satisfacción de las necesidad del usuario; la ***confiabilidad*** que hace referencia a mantener su rendimiento bajo ciertas condiciones durante cierto periodo de tiempo; la ***usabilidad***, es decir basada en el esfuerzo necesario para utilizar el software por parte de un grupo de usuarios; la ***eficiencia*** que se basa en la relación entre el nivel de rendimiento del software y el volumen de recursos utilizado, bajo ciertas condiciones; la capacidad de ***mantenimiento***, es decir basada en el esfuerzo necesario para realizar modificaciones especificas; además de la ***portabilidad*** que hace referencia a la capacidad del software para ser transferido de un entorno a otro.

En el Cuadro 3 y Gráfico 2 se presentan los resultados que evidencian el nivel de calidad de los algoritmos diseñados y construidos por los estudiantes, antes de que pasaran por un proceso de enseñanza aprendizaje respecto de las estrategias didácticas.

Cuadro 3:

Relación de indicadores de medición del nivel de calidad del software, contenidas en la guía de observación, antes de su enseñanza.

Indicador	Grupo de estudio											
	Excelente		Muy bueno		Bueno		Regular		Deficiente		Total	
	Nº	%	Nº	%	Nº	%	Nº	%	Nº	%	Nº	%
Funcionalidad			1	3	4	11	10	29	20	57	35	100
Confiabilidad			2	6	6	17	5	14	22	63	35	100
Usabilidad			1	3	5	14	8	23	21	60	35	100
Eficiencia			1	3	3	9	7	20	24	69	35	100
Capacidad de mantenimiento			2	6	1	3	6	17	26	74	35	100
Portabilidad			1	3	2	6	4	11	28	80	35	100

***Fuente: Pre test aplicado a los estudiantes del III ciclo de computación e informática del I.E.S.T. Público "Perú - Japón"* – Chachapoyas.**

En el Cuadro 3, donde se muestran los resultados luego de haber aplicado el pre test a los estudiantes del grupo de estudio, se observa que el mayor porcentaje de estudiantes, se ubica en el nivel deficiente respecto de los ***indicadores de medición del nivel de calidad del software*** en el diseño y construcción de algoritmos, de conformidad con los indicadores establecidos para tal fin; puntualmente entre el 57% y el 80% de los estudiantes; solo un mínimo porcentaje de estudiantes que oscila entre el 3% y 6% se ubica en el nivel muy bueno.

Los resultados indican que antes de aplicar la variable independiente al grupo de estudio (estrategias didácticas para facilitar el diseño y construcción de algoritmos), la condición del grupo en relación a la calidad del software es deficiente, y que cualquier variación positiva del grupo de estudio en adelante es atribuible a la variable independiente.

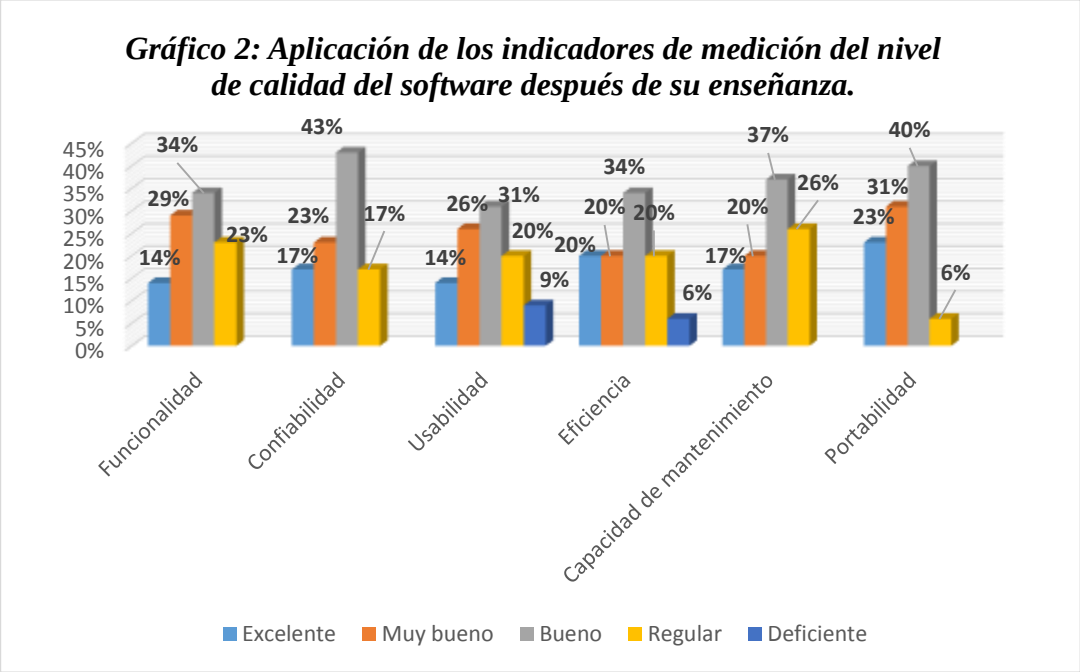
Luego de aplicar las estrategias didácticas a los estudiantes del grupo de estudio, los resultados muestran una mejora sustantiva, tal como se visualiza en el Cuadro 4.

Cuadro 4:

Aplicación de los indicadores de medición del nivel de calidad del software, después de su enseñanza.

	Grupo es estudio											
Indicador	**Excelente**		**Muy bueno**		**Bueno**		**Regular**		**Deficiente**		**Total**	
	Nº	**%**	**Nº**	**%**	**Nº**	**%**	**Nº**	**%**	**Nº**	**%**	**Nº**	**%**
Funcionalidad	5	14	10	29	12	34	8	23			35	100
Confiabilidad	6	17	8	23	15	43	6	17			35	100
Usabilidad	5	14	9	26	11	31	7	20	3	9	35	100
Eficiencia	7	20	7	20	12	34	7	20	2	6	35	100
Capacidad de mantenimiento	6	17	7	20	13	37	9	26			35	100
Portabilidad	8	23	11	31	14	40	2	6			35	100

Fuente: Pos test aplicado a los estudiantes del III ciclo de computación e informática del I.E.S.T. Público "Perú - Japón" – **Chachapoyas.**



Fuente: Pos test aplicado a los estudiantes del III ciclo de computación e informática del I.E.S.T. Público "Perú - Japón" – **Chachapoyas.**

En el Cuadro 4 y Gráfico 2, donde se muestran los resultados, se observa una variación positiva, donde los porcentajes más altos se ubican en los niveles bueno y muy bueno, cuyos porcentajes oscilan entre el 31% y 43% y entre los 20% y 31% respectivamente; solo cinco estudiantes que representan el 15% se ubica en el nivel deficiente, no logrando específicamente los criterios de usabilidad y eficiencia.

Se refiere a que esta variación positiva en el grupo de estudio se debe a la aplicación de las estrategias didácticas (variable independiente), lo que significa que dichas estrategias son eficaces y permiten el logro del objetivo planteado, y en consecuencia se logra solucionar el problema materia de investigación.

De conformidad con los resultados, se concluye que el grupo de estudio avanzó en el desarrollo de la capacidad del diseño y construcción de algoritmos en cuanto a la medición del nivel de calidad del software, en razón a la aplicación de las estrategias didácticas orientadas a este fin; por lo tanto se afirma que las estrategias diseñadas y aplicadas han permitido un avance decisivo en el desarrollo

de esta capacidad, lo que confirma el logro del objetivo general, y en consecuencia la solución del problema en estudio.

Asimismo, con la presentación, análisis e interpretación de los resultados, se corrobora la hipótesis de trabajo planteada en este estudio, la cual señala que "Si, se diseñan, elaboran y ejecutan estrategias didácticas con base en el uso del software PSeInt, en el diseño y construcción de algoritmos", entonces se desarrollará significativamente dicha capacidad.

3.1.3. Resultados en relación al dominio del software PSeInt.

El software PSeInt es una herramienta valiosa a la hora de diseñar y construir algoritmos, tanto por parte del estudiante como del profesor; es una aplicación gratuita que está dirigida fundamentalmente a la escritura de pseudocódigo. Permite escribir, ejecuta y diagramar un algoritmo.

Es uno de los programas de acceso libre más novedoso y versátil que existe en la actualidad para el diseño y construcción de algoritmos. La herramienta PSeInt es el resultado de varios proyectos innovadores que se han creado para utilizarse como editor de pseudocódigo.

Ante el objetivo de identificar el Dominio del Software PSeInt por parte de los estudiantes del III ciclo de computación e informática del I.E.S.T. Público "Perú - Japón" – Chachapoyas, antes y después de su enseñanza se presenta en los Cuadros 5, 6 y Figura 3.

Cuadro 5:

Relación de los indicadores del dominio del software PSeInt, contenidas en la guía de observación, antes de su enseñanza.

Dominio del software	Grupo de estudio					
	Si		No		Total	
	Nº	%	Nº	%	Nº	%
Instala correctamente el software.			35	100	35	100
Usa correctamente los comandos de las estructuras.			35	100	35	100
Ejecuta correctamente el algoritmo.			35	100	35	100
Depura correctamente los errores encontrados.			35	100	35	100
Genera correctamente el diagrama de flujo.			35	100	35	100

***Fuente: Guía de observación a los estudiantes del III ciclo de computación e informática del I.E.S.T. Público "Perú - Japón" –* Chachapoyas.**

Los resultados que se muestran en el Cuadro 5, indican que el 100% de todos los estudiantes, no tenían ningún conocimiento con relación al dominio del software PSeInt; es decir el 100% de los estudiantes no podían: instalar el software, usar comandos de las estructuras, ejecutar el algoritmo, depurar los errores encontrados y tampoco generar el diagrama de flujo.

El dominio del software PSeInt implica, por un lado, estar en condiciones de poder instalar el software, lo cual es bastante sencillo, puesto que se trata de un software liviano por lo que su instalación se realiza en poco tiempo; aunque para tal fin se requiere tener un mínimo conocimiento de computación en lo relacionado a la manipulación del mouse y el teclado y conocer las características básicas del sistema operativo que se utilice; luego sencillamente se sigue los pasos secuenciales que se van presentando en el proceso de instalación. Otro aspecto básico que implica el dominio del software PSeInt tiene que ver con poder escribir el algoritmo usando los comandos de estructuras; esto también es bastante simple, solamente hay que dar clic sobre el comando a usar.

De igual forma, el dominio del referido software implica poder ejecutar el algoritmo escrito; esta acción es sencillo basta con dar clic en ejecutar. Además este dominio tiene mucho que ver con corregir errores y darles solución para que el algoritmo sea de calidad y sin errores.

Finalmente, el dominio del software PSeInt implica estar en condiciones de crear los diagramas de flujo, lo cual se realiza dando clic en el generador de diagramas de flujo del referido software.

Después de que los estudiantes del grupo de control han pasado por un proceso de enseñanza aprendizaje del software PSeInt, los resultados fueron diferentes, tal como se observa en el cuadro 6 y gráfico 3.

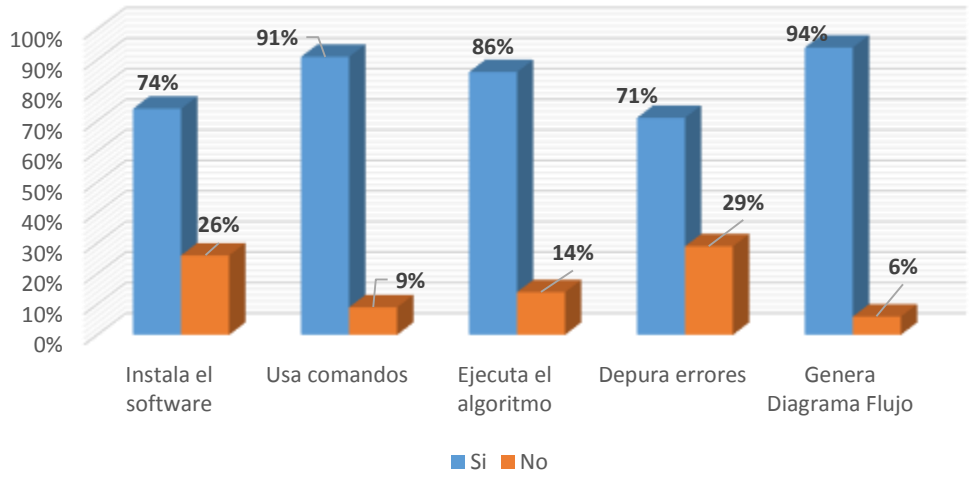
Cuadro 6:

Aplicación de los indicadores del dominio del software PSeInt, después de su enseñanza.

Indicadores	**Grupo de estudio**					
	Si		**No**		**Total**	
	Nº	**%**	**Nº**	**%**	**Nº**	**%**
Instala correctamente el software.	26	74	9	26	35	100
Usa correctamente los comandos de las estructuras.	32	91	3	9	35	100
Ejecuta correctamente el algoritmo.	30	86	5	14	35	100
Depura correctamente los errores encontrados.	25	71	13	29	35	100
Genera correctamente el diagrama de flujo.	33	94	2	6	35	100

Fuente: Guía de observación a los estudiantes del III ciclo de computación e informática del I.E.S.T. Público "Perú - Japón" **– Chachapoyas.**

Gráfico 3: Aplicación de los indicadores del dominio del software PseInt después de su enseñanza.



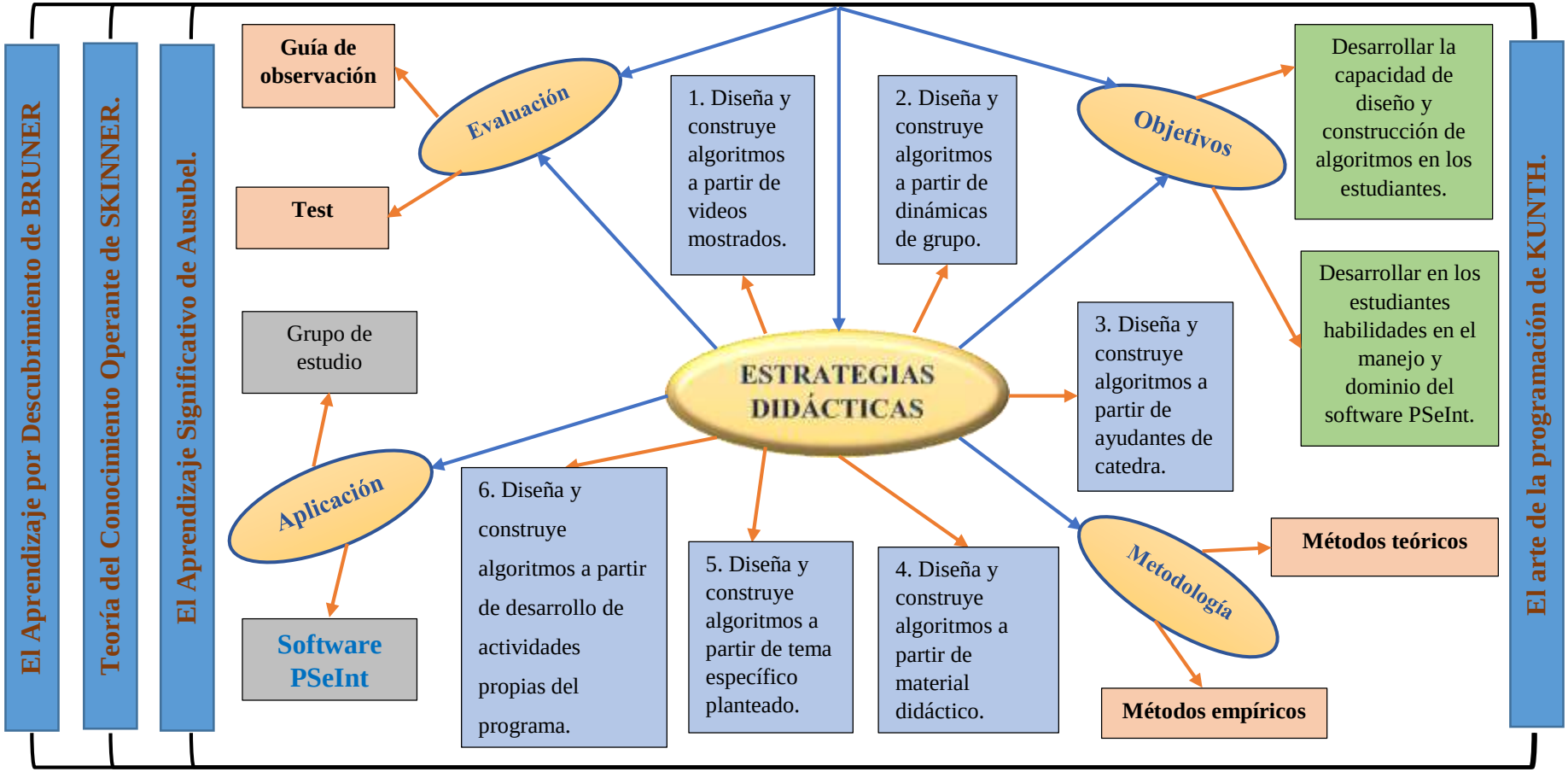
***Fuente: Guía de observación a los estudiantes del III ciclo de computación e informática del I.E.S.T. Público "Perú - Japón"* – Chachapoyas.**

Se observa en el Cuadro 6 y Gráfico 3, que los estudiantes del grupo de estudio evidencian un avance significativo en el conocimiento y manejo de dicho software, puesto que entre el 71% y el 94% dominan el software, y solo un pequeño porcentaje que oscila entre el 9% y el 29% no evidencia un dominio eficiente. El mayor avance se percibe en los indicadores "usa correctamente los comandos de las estructuras" y "genera correctamente el diagrama de flujo", donde alcanzan el 91% y 94% respectivamente.

A partir de estos resultados se concluye que el proceso de enseñanza y aprendizaje en relación al dominio del software PSeInt, en el que estuvieron inmersos los estudiantes del grupo de estudio, fue eficiente: este logro ha servido de base para alcanzar el objetivo general planteado en este trabajo de investigación; puesto que las estrategias didácticas diseñadas y aplicadas al grupo de estudio para desarrollar la capacidad de diseño y construcción de algoritmos ha tenido como base el uso del este software.

3.2. GRÁFICO DEL MODELO TEÓRICO DE LA PROPUESTA

PROGRAMA DE ESTRATEGIAS DIDÁCTICAS BASADAS EN EL USO DEL SOFTWARE PSEINT PARA DESARROLLAR LA CAPACIDAD DEL DISEÑO Y CONSTRUCCIÓN DE ALGORITMOS.



3.3. PROPUESTA TEÓRICA

3.3.1. PRESENTACIÓN

La presente propuesta teórica, titulado **"Programar es un arte muy sencillo",** adquiere significado, porque se orienta a desarrollar en los estudiantes del III ciclo de computación e informática del I.E.S.T. Público "Perú - Japón", Chachapoyas, una de las competencias básicas, como parte de las exigencias del trabajo curricular dentro de la unidad didáctica lógica de programación.

La problemática abordad en este estudio y que motivo el diseño y aplicación de la propuesta, tuvo que ver con deficiencias observadas en el aprendizaje de los estudiantes del III ciclo de computación e informática del I.E.S.T. Público "Perú - Japón", Chachapoyas, en la unidad didáctica lógica de programación.

En esta unidad didáctica se ha percibido que los estudiantes tienen un nivel muy bajo en el desarrollo de la capacidad de diseño y construcción de algoritmos; evidenciándose en la incapacidad para crear algoritmos sencillos teniendo en cuenta las características del software de calidad, serias dificultades para pensar lógicamente, frecuentes errores en la escritura del algoritmo, dificultades para solucionar los errores que se generan, así como dificultades para generar los diagramas de flujo.

Considerando que el desarrollo de la capacidad de diseño y construcción de algoritmos, es una tarea fundamental que se tiene que cumplir en el trabajo de la unidad didáctica, se ha planteado como alternativa de solución al problema, esta propuesta de estrategias, que tiene su base en el uso del software educativo PSeInt.

PSeInt es una herramienta valiosa al momento de diseñar y construir algoritmos, tanto por parte del estudiante como del profesor, es una aplicación gratuita que está dirigida fundamentalmente a la escritura de algoritmos utilizando pseudo lenguaje simple, intuitivo y en español. Permite escribir, ejecutar y generar los diagramas de flujo de los algoritmos.

La elección de PSeInt para este trabajo de investigación, se hizo intencionadamente como una manera de fomentar e integrar el uso de las nuevas

tecnologías en el aula; pero además las ventajas significativas que ofrece han tenido cierto peso a la hora de optar por esta propuesta.

Entre las ventajas tenemos:

- Permite la edición simultánea de múltiples algoritmos.
- Permite generar y editar el diagrama de flujo del algoritmo.
- El lenguaje seudocódigo utilizado es configurable
  Ofrece perfiles de configuración predeterminados para numerosas instituciones.
- Puede interpretar (ejecutar) los algoritmos escritos:
  Permite ejecutar el algoritmo paso a paso controlando la velocidad e inspeccionando variable y expresiones.
  Puede confeccionar automáticamente una tabla de prueba de escritorio.
- Es una herramienta de software libre, con una interfaz intuitiva y en español.
- Determina y marca claramente errores de sintaxis y en tiempo de ejecución.
- Permite convertir el algoritmo de pseudocódigo a código C++.

Así pues el hecho de ser un software de uso libre, de presentar una interfaz sencilla y fácil de aprender por un usuario con limitados conocimientos informáticos, lo que significa que los estudiantes del III ciclo de computación e informática no tendrían mayores inconvenientes para adecuarse a su uso y majeño; el hecho de que su interfaz permita trabajar al estudiante de manera rápida y cómoda; estas razones llevaron a optar por este software en este trabajo de investigación.

Pero además el poder trabajar individualmente y grupalmente sin necesidad de conexión a internet, bastando solo con guardar el archivo generado; y sobre todo, el hecho de que sea posible plantear estrategias didácticas a partir del uso de este software para desarrollar en los estudiantes la capacidad de diseñar y construir algoritmos, entre otras ventajas, evidencian las razones que han llevado a plantear la solución de la problemática encontrada en los estudiantes tomando como elemento de base el uso de este programa.

Cabe señalar que los planteamientos señalados en este programa, no deben ser tomados de ninguna manera como una receta o como una verdad acabada; la intención es proporcionar al docente de la unidad didáctica un material que le sirva de guía y oriente su trabajo pedagógico, de tal forma que se constituya en un pequeño foco de luz respecto a las estrategias didácticas que deben considerarse para desarrollar la competencia De diseño y construcción de algoritmos; considerando que no existen trabajos que aborden de modo especifico el asunto; sin embargo el docente, haciendo uso de su creatividad podrá, a partir de lo que aquí se propones, plantear y/o aplicar otras estrategias, ejercicios y procedimientos que vayan en esta misma línea de acción.

Finalmente cabe señalar que los directamente beneficiados con esta propuesta son los estudiantes, con el desarrollo de la competencia diseño y construcción de algoritmos, los docentes de la unidad didáctica, por utilizar las estrategias didácticas, y en general todos los miembros de la comunidad educativa donde se aplique el programa.

3.3.2. OBJETIVOS DEL PROGRAMA

- Diseñar estrategias didácticas del diseño y construcción de algoritmos, en los estudiantes del III ciclo de computación e informática del I.E.S.T. Público "Perú - Japón" de Chachapoyas, a partir del uso del software PSeInt.
- Desarrollar en los estudiantes habilidades en el manejo y dominio del software PSeInt, que sirvan de base para el desarrollo de la capacidad del diseño y construcción de algoritmos.

3.3.3. FUNDAMENTACIÓN

Desde el punto de vista filosófico, epistemológico y psicopedagógico, este programa se sustenta en los principios del constructivismo, según este paradigma, la realidad es una construcción de nuestros pensamientos por el que organizamos nuestro mundo experimental y conforme a ello percibimos la realidad y actuamos sobre ella, no es un registro fiel de lo existente.

En psicopedagogía, la idea del constructivismo está referida a la concepción de los procesos cognitivos como construcciones eminentemente activas del sujeto que conoce, en interacción con su ambiente físico y social, vale decir que es el

estudiante el que construye su propio aprendizaje; en este marco el docente es un mediador cultural o facilitador del aprendizaje, por tanto no es el dueño del conocimiento ni puede trasmitirlo mecánicamente.

En consecuencia con los principios básicos del constructivismo, en este trabajo se enfatiza la participación activa del estudiante como responsable último de su propio aprendizaje, él es quien debe asimilar el conocimiento, para lo cual es necesario que el docente como facilitador aplique estrategias didácticas que motiven su participación, que es precisamente lo que se busca con este programa.

Sobre la base de la posición constructivista son dos teorías psicopedagógicas que sustentan este programa son: La Teoría sociocultural de Vygotsky y la Teoría de las Inteligencias múltiples de Gardner.

Para los intereses de este trabajo de investigación, el enfoque sociocultural de Vygotsky es clave y fundamental, puesto que el aprendizaje bajo esta concepción se entiende como social, es decir, un aprendizaje en interacción con otras personas y mediadas por instrumentos culturales que son las herramientas y los signos. En este trabajo, las herramientas fundamentales son la computadora y el software PSeInt, mientras que el signo, es el lenguaje pseudocódigo que se da en este medio, como un conjunto de elementos interconectados como son el lenguaje en sí, los diagramas, etc.

Los elementos medulares de la propuesta de Vygotsky como la conciencia, los instrumentos que son las herramientas y los signos, además de la zona de desarrollo próximo se relacionan con el desarrollo y aprendizaje de los estudiantes a través de las TIC, por lo que se asumen como una teoría de singular importancia.

En la teoría de inteligencias múltiples vemos que desde la escuela se deben procurar desarrollar las múltiples capacidades, habilidades o potencialidades del estudiante; para ello es importante no limitarse a la utilización de una única metodología de enseñanza – aprendizaje, ni al uso de ciertos recursos didácticos como únicos elementos generadores de aprendizaje.

Visto este trabajo de investigación desde esta óptica, cobra sentido hablar de esta teoría sustento de este trabajo, puesto que se busca a través de las estrategias

didácticas diseñadas con bases en el uso del software PSeInt, proponer una forma creativa de desarrollar el proceso enseñanza aprendizaje de la unidad didáctica lógica de programación, proporcionando a los estudiantes espacios donde pueden trabajar con comandos, gráficos, videos, etc. En este contexto los estudiantes pueden trabajar de manera individual y en equipos.

Se busca por tanto superar el modelo didáctico tradicional, limitado a la pizarra, la tiza y el verbalismo, donde la creatividad no tiene lugar, y en consecuencia muchos estudiantes están realmente limitados para desarrollar sus potencialidades; al romper con estos esquemas tradicionales se abre las puertas al estudiante para que aprenda de manera divertida y creativa, y en consecuencia logre desarrollar sus inteligencias.

3.3.4. METODOLOGÍA

Para la elaboración de este programa se ha utilizado tanto los métodos (analítico, sintético, inductiva y deductivo), así como lo métodos empíricos (experimento, observación y medición). Los métodos teóricos han servido para sistematizar las estrategias didácticas, y los métodos empíricos para aplicar dichas estrategias.

Los talleres se realizaron en cada sesión de aprendizaje en el laboratorio de computación de los estudiantes del III ciclo de computación e informática.

La secuencia de acciones metodológicas serán las siguientes:

- Selección, lectura y análisis de fuentes teóricas relacionadas a la lógica de programación.
- Uso de la pizarra digital.
- Guía de programación, conteniendo ejercicios desarrollados y propuestos.
- Selección de enunciados de algoritmos a trabajar en el software PSeInt.
- Ejecutar la ficha de sesión de aprendizaje.
- Desarrollo de clases orientadas al manejo y dominio del software PSeInt.
- Aplicación de las estrategias didácticas en el desarrollo del taller.
- Medición de los avances a través de pos test.

3.3.5. PRESENTACIÓN DE LAS ESTRATEGIAS

En esta sección se presentan las estrategias didácticas que se han diseñado y aplicado con el fin de desarrollar en los estudiantes la capacidad de diseño y construcción de algoritmos. Para cada estrategia se hace una descripción, indicando en que consiste y señalando sus principales características; también se señalan de manera precisa los pasos a seguir en cada una, es decir los procesos correspondientes.

ESTRATEGIA 1:

DENOMINACIÓN: DISEÑA Y CONSTRUYE ALGORITMOS A PARTIR DE VIDEOS MOSTRADOS.

DESCRIPCIÓN:

Consiste en presentar videos que guarden relación con algún contenido o tema a trabajar, luego se pide a los estudiantes que escriban en un papel el algoritmo presentado y creen su diagrama de flujo para luego ser expuesto.

PROCEDIMIENTO:

- Se determina el tema o contenido a trabajar.
- Se seleccionan los videos que guarden relación con el tema.
- Se presenta los videos a los estudiantes usando la data o haciendo entrega a través del uso de un USB.
- Los estudiantes trasladan los videos presentadas a sus respectivas computadoras y lo escriben en el software PSeInt.
- Los estudiantes ejecutan sus algoritmos.
- Los estudiantes desarrollan actividades específicas sobre diseño y construcción de algoritmos y realizan intercambios de archivos generados.

TIEMPO ESTIMADO:

Dos clases de 90 minutos cada una.

La actividad concluida se visualizara así:

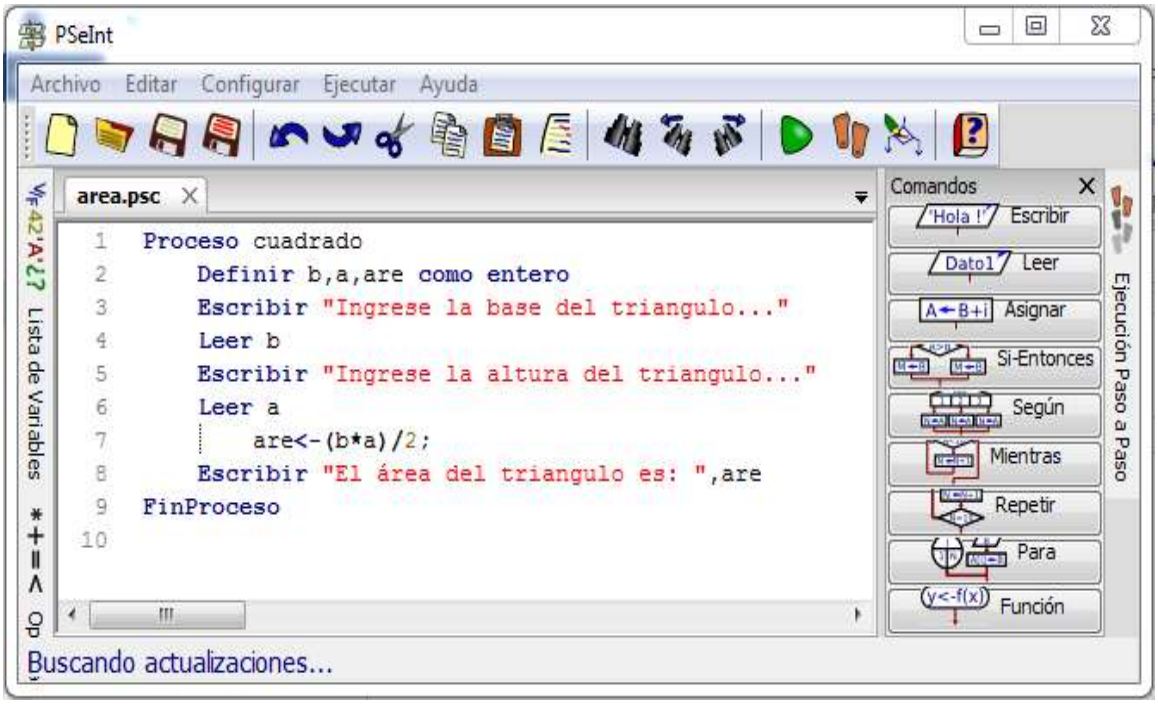


Gráfico 7: Interfaz gráfica del programa PSeInt.

SESIÓN DE APRENDIZAJE 1

ESPECIALIDAD : Computación e Informática
MÓDULO ESPECÍFICO : Desarrollo de software y gestión de base de datos
UNIDAD DIDÁCTICA : **LÓGICA DE PROGRAMACIÓN**
CAPACIDAD TERMINAL: Realizar la lógica de la programación considerando procesos y secuencias.
ACTIVIDAD DE APRENDIZAJE N° 1: **Elabora algoritmos secuenciales y condicionales.**

ELEMENTO DE LA CAPACIDAD TERMINAL: Utiliza el pseudocódigo para representar algoritmos procedimentales.	**Lugar**	**Horas Pedagógicas**
	LABORATORIO (X) TALLER (X)	4
	AULA ( )	

CONTENIDOS

Procedimientos	**Conceptos**	**Actitudes**
Conceptualiza las herramientas necesarias para la resolución de problemas por computadora	Introducción a la programación. Datos y expresiones, definición de variables y constantes. Herramientas de programación. Seudocódigo. Ejemplos de algoritmos secuenciales y condicionales. Operadores, algoritmos, concepto, características.	○ Muestra interés en conocer Los pasos secuenciales de un algoritmo.

SECUENCIA METODOLÓGICA

Momentos	**Estrategias**	**Métodos/ Técnicas**	**Recursos**	**Duración**
Motivación	○ Recoger saberes previos sobre programación: ¿Qué son los datos? ¿Qué herramientas usar?	Expositivo Relación D-A Lluvia de ideas	PC USB Data	10 min
Proporcionar Información	○ Los estudiantes leen en forma individual la separata sobre cómo crear el pseudocódigo.	Estudio dirigido	Separatas USB, Laptop Multimedia PDF	30 min
Desarrollar Práctica dirigida	○ Práctica guiada de creación de pseudocódigo.	Práctica individual Relación D-A	Guía de práctica. PC	45 min
Resolución de Problemas y transferencia	○ Se da respuesta a las interrogantes propuestas por los alumnos con respecto al tema.	Inductivo Relación D-A	Revistas Libros	45 min
Evaluación	○ Desarrollan la práctica de creación de algoritmos usando las herramientas adecuadas.	Observación Examen escrito	Ficha de observación Prueba escrita	50 min

EVALUACIÓN DE LOS APRENDIZAJES

CRITERIO (S) DE EVALUACIÓN: Realizar algoritmos de procesos y subprocesos

Indicadores	**Técnicas**	**Instrumento**
○ Elabora algoritmos secuenciales y condicionales con precisión-. ○ Muestra interés en conocer los pasos secuenciales de un algoritmo.	○ Exposición oral ○ Examen	○ Examen escrito ○ Ficha de observación

ESTRATEGIA 2:

DENOMINACIÓN: DISEÑA Y CONSTRUYE ALGORITMOS A PARTIR DE SOFTWARE LIBRE.

DESCRIPCIÓN:

Consiste en presentar a los estudiantes el uso de software libre, porque se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software. En este caso sería el software PSeInt que se usara en el diseño y construcción de algoritmos.

PROCEDIMIENTO:

- Buscar en internet software libre para desarrollar algoritmos.
- Seleccionar el software que se ajuste mejor a las necesidades del estudiante.
- Descargar el software.
- Guardarlo en su carpeta específica.

Ejemplo

TIEMPO ESTIMADO:

Dos clases de 90 minutos cada una.

SESIÓN DE APRENDIZAJE 2

ESPECIALIDAD : Computación e Informática
MÓDULO ESPECÍFICO : Desarrollo de software y gestión de base de datos
UNIDAD DIDÁCTICA : **LÓGICA DE PROGRAMACIÓN**
CAPACIDAD TERMINAL: Realizar la lógica de la programación considerando procesos y secuencias
ACTIVIDAD DE APRENDIZAJE N° 2: **Elabora algoritmos repetitivos finito e infinitos.**

ELEMENTO DE LA CAPACIDAD TERMINAL: Utiliza el pseudocódigo para representar algoritmos procedimentales.	**Lugar**	**Horas Pedagógicas**
	LABORATORIO (X) TALLER ( )	4
	AULA (X)	

CONTENIDOS

Procedimientos	**Conceptos**	**Actitudes**
Desarrolla la estructura para la solución de un problema computacional.	Análisis del problema Especificaciones de entradas y salidas.	○ Participa de manera activa en el desarrollo de las prácticas y valora la importancia de los algoritmos.

SECUENCIA METODOLÓGICA

Momentos	**Estrategias**	**Métodos/ Técnicas**	**Recursos**	**Duración**
Motivación	○ Recoger saberes previos sobre la elaboración de: ¿Qué son algoritmos finitos? ¿Qué son algoritmos infinitos?	Expositivo Relación D-A Lluvia de ideas	PC USB Data	10 min
Proporcionar Información	○ Los estudiantes leen en forma individual la separata. ○ Presentación de diapositivas con los temas y se entrega el archivo.	Estudio dirigido	Separatas USB, Laptop Multimedia PPT	30 min
Desarrollar Práctica dirigida	○ Práctica guiada de creación de los tipos de algoritmos usando un lenguaje de programación para desarrollar el pseudocódigo.	Práctica individual Relación D-A	Guía de práctica. PC	45 min
Resolución de Problemas y transferencia	○ Se da respuesta a las interrogantes propuestas por los alumnos con respecto al tema.	Inductivo Relación D-A Tutoría	Revistas Libros	45 min
Evaluación	○ Desarrollan los ejercicios haciendo uso de la computadora y en lenguaje Devc ++.	Observación Ejercicios prácticos	Ficha de observación	50 min

EVALUACIÓN DE LOS APRENDIZAJES

CRITERIO (S) DE EVALUACIÓN: Realizar algoritmos de procesos y subprocesos

Indicadores	**Técnicas**	**Instrumento**
○ Describe en forma ordenada los procesos y subprocesos de un algoritmo. ○ Participa de manera activa en el desarrollo de las prácticas y valora la importancia de los algoritmos.	○ Examen práctico	○ Práctica calificada ○ Ficha de observación

ESTRATEGIA 3:

DENOMINACIÓN: DISEÑA Y CONSTRUYE ALGORITMOS A PARTIR DE DINÁMICAS DE GRUPO.

DESCRIPCIÓN:

Consiste en presentar a los estudiantes diversas dinámicas de grupo porque estas técnicas favorecen la cohesión grupal y el trabajo en equipo. Estas técnicas tienen como fin favorecer el conocimiento entre los miembros del grupo, mejorar la comunicación, reforzar su confianza o fomentar la colaboración entre ellos, mientras que otras van más encaminadas a generar nuevas ideas, mejorar la toma de decisiones o potenciar la creatividad grupal o individual.

PROCEDIMIENTO:

- Agrupar a los estudiantes en grupos de trabajo.
- Aplicar una dinámica de las que ya hemos descrito anteriormente.
- Trabajar el diseño de algoritmos después de la aplicación de la dinámica y comparar los resultados obtenidos.

Ejemplo

- Dinámicas para la reflexión grupal e individual: el dado, supervivencia en los andes.
- Dinámicas de confianza: la estatua, los sonidos de los animales.
- Dinámicas de trabajo en equipo: técnica 6.3.5, el globo aerostático, cosas en común.

TIEMPO ESTIMADO:

Dos clases de 90 minutos cada una.

SESIÓN DE APRENDIZAJE 3

ESPECIALIDAD : Computación e Informática
MÓDULO ESPECÍFICO : Desarrollo de software y gestión de base de datos
UNIDAD DIDÁCTICA : **LÓGICA DE PROGRAMACIÓN**
CAPACIDAD TERMINAL: Realizar la lógica de la programación considerando procesos y secuencias.
ACTIVIDAD DE APRENDIZAJE N° 3: **Elabora algoritmos con uso de arreglos.**

ELEMENTO DE LA CAPACIDAD TERMINAL:	**Lugar**	**Horas Pedagógicas**
Utiliza el pseudocódigo para representar algoritmos procedimentales.	LABORATORIO (X) TALLER ( )	2
	AULA (X)	

CONTENIDOS

Procedimientos	**Conceptos**	**Actitudes**
Reconoce los componentes para el análisis del problema. Conceptualiza las herramientas necesarias para el diseño de algoritmos.	○ Diseño de algoritmos, diseño descendente, refinamientos por pasos. ○ Arreglos, declaración, creación. ○ Clases, operaciones. Ejemplos.	○ Participa de manera activa en el desarrollo de las prácticas y valora la importancia de los algoritmos.

SECUENCIA METODOLÓGICA

Momentos	**Estrategias**	**Métodos/ Técnicas**	**Recursos**	**Duración**
Motivación	○ Recoger saberes previos sobre la elaboración de arreglos: ¿Qué son arreglos unidimensionales? ¿Qué son arreglos bidimensionales?	Expositivo Relación D-A Lluvia de ideas	PC USB Data	20 min
Proporcionar Información	○ Informar sobre la construcción de arreglos. ○ Los estudiantes leen en forma individual la separata. ○ Presentación de diapositivas con los temas y se entrega el archivo.	Estudio dirigido	Separatas USB, Laptop Multimedia PDF	30 min
Desarrollar Práctica dirigida	○ Práctica guiada de operaciones con arreglos.	Práctica individual Relación D-A	Guía de práctica. PC	45 min
Resolución de Problemas y transferencia	○ Se da respuesta a las interrogantes propuestas por los alumnos con respecto al tema.	Inductivo Relación D-A Tutoría	Revistas Libros	45 min
Evaluación	○ Desarrollan los ejercicios haciendo uso de la computadora y en lenguaje Devc ++	Observación Ejercicios prácticos	Ficha de observación Pc	40 min

EVALUACIÓN DE LOS APRENDIZAJES

CRITERIO (S) DE EVALUACIÓN: Realizar algoritmos de procesos y subprocesos

Indicadores	**Técnicas**	**Instrumento**
○ Crea arreglos y desarrolla los procesos usando datos reales con precisión. ○ Participa de manera activa en el desarrollo de las prácticas y valora la importancia de los algoritmos.	○ Observación ○ Examen	○ Práctica calificada ○ Ficha de observación

ESTRATEGIA 4:

DENOMINACIÓN: DISEÑA Y CONSTRUYE ALGORITMOS A PARTIR DE AYUDANTES DE CÁTEDRA.

DESCRIPCIÓN:

Consiste en que los estudiantes seleccionados como ayudantes de catedra brinden soporte al docente, a través de la enseñanza en las clases prácticas con metodología activa, para facilitar el aprendizaje de los estudiantes a través de la práctica; además de asistir al docente en la evaluación de prácticas calificadas, para facilitar la labor académica y el aprendizaje del estudiante.

PROCEDIMIENTO:

- El o los estudiantes ayudantes de catedra brindaran acompañamiento a sus compañeros.
- Ayudarán a reforzar y resolver las interrogantes que presenten los estudiantes.
- Debe tomar decisiones y solucionar problemas.

Ejemplos

- Dinámicas para la reflexión grupal e individual: el dado, supervivencia en los andes.
- Dinámicas de confianza: la estatua, los sonidos de los animales.
- Dinámicas de trabajo en equipo: técnica 6.3.5, el globo aerostático, cosas en común.

TIEMPO ESTIMADO:

Dos clases de 90 minutos cada una.

Gráfico 4: Estudiantes siendo asesorados.

SESIÓN DE APRENDIZAJE 4

ESPECIALIDAD : Computación e Informática
MÓDULO ESPECÍFICO : Desarrollo de software y gestión de base de datos
UNIDAD DIDÁCTICA : **LÓGICA DE PROGRAMACIÓN**
CAPACIDAD TERMINAL: Realizar la lógica de la programación considerando procesos y secuencias.
ACTIVIDAD DE APRENDIZAJE N° 4: **Diseña y ejecuta diagramas de flujo de algoritmos procedimentales.**

ELEMENTO DE LA CAPACIDAD TERMINAL: Utiliza el pseudocódigo para representar algoritmos procedimentales.	**Lugar**	**Horas Pedagógicas**
	LABORATORIO (X) TALLER ( )	2
	AULA (X)	

CONTENIDOS

Procedimientos	**Conceptos**	**Actitudes**
Desarrolla la estructura para la resolución de un problema computacional.	○ Herramientas de programación Diagramas de flujo Casos aplicativos.	○ Demuestra iniciativa y responsabilidad en el trabajo en equipo e individual.

SECUENCIA METODOLÓGICA

Momentos	**Estrategias**	**Métodos/ Técnicas**	**Recursos**	**Duración**
Motivación	○ Recoger saberes previos sobre el uso de las herramientas como: ¿Qué son los diagramas de flujo? ¿Cómo representar un algoritmo?	Expositivo Relación D-A Lluvia de ideas	PC USB Data	20 min
Proporcionar Información	○ Los estudiantes leen en forma individual la separata. ○ Presentación de diapositivas con los temas y se entrega el archivo.	Estudio dirigido	Separatas USB, Laptop Multimedia PPT PDF	30 min
Desarrollar Práctica dirigida	○ Desarrollo de ejercicios modelos, usando el programa del Visio.	Práctica individual Relación D-A	Guía de práctica. PC	45 min
Resolución de Problemas y transferencia	○ Se da respuesta a las interrogantes propuestas por los alumnos con respecto al tema.	Inductivo Relación D-A Tutoría	Revistas Libros	45 min
Evaluación	○ Desarrollan los ejercicios haciendo uso de la computadora y en lenguaje Devc ++ y del Visio 2010	Observación Ejercicios prácticos	Ficha de observación Pc	40 min

EVALUACIÓN DE LOS APRENDIZAJES

CRITERIO (S) DE EVALUACIÓN: Elabora y comprueba diagramas de flujo de los procedimientos requeridos

Indicadores	**Técnicas**	**Instrumento**
○ Diseña diagramas de flujo de los procesos de un algoritmo adecuadamente. ○ Demuestra iniciativa y responsabilidad en el trabajo en equipo e individual.	○ Examen ○ Observación	○ Examen escrito ○ Ficha de observación

ESTRATEGIA 5:

DENOMINACIÓN: DISEÑA Y CONSTRUYE ALGORITMOS A PARTIR DE MATERIAL DIDÁCTICO.

DESCRIPCIÓN:

Consiste en utilizar medios y recursos para facilitar la enseñanza y el aprendizaje. Aquí utilizamos libros, revistas, internet, diapositivas y separatas que se les entrega en cada sesión de clase incluyendo casos prácticos desarrollados y propuestos.

PROCEDIMIENTO:

- El docente al inicio de la sesión de clase hace la entrega del material didáctico a emplearse.
- Los estudiantes leen y analizan el tema a tratar.
- Los casos prácticos y propuestos se desarrollan en clase junto al docente y a los ayudantes de catedra.

Ejemplo

Gráfico 5: Estudiante usando material didáctico

TIEMPO ESTIMADO:

Dos clases de 90 minutos cada una.

SESIÓN DE APRENDIZAJE 5

ESPECIALIDAD : Computación e Informática
MÓDULO ESPECÍFICO : Desarrollo de software y gestión de base de datos
UNIDAD DIDÁCTICA : **LÓGICA DE PROGRAMACIÓN**
CAPACIDAD TERMINAL: Realizar la lógica de la programación considerando procesos y secuencias.
ACTIVIDAD DE APRENDIZAJE N° 5: Elabora algoritmos procedimentales y verifica su validez generando código C++ correspondiente.

ELEMENTO DE LA CAPACIDAD TERMINAL: Utiliza el pseudocódigo para representar algoritmos procedimentales.	**Lugar**	**Horas Pedagógicas**
	LABORATORIO (X) TALLER ( )	2
	AULA (X)	

CONTENIDOS

Procedimientos	**Conceptos**	**Actitudes**
Conceptualiza los métodos formales de verificación de programas	Resolución de algoritmos en la computadora, codificación del algoritmo, ejecución y comprobación del programa.	○ Demuestra iniciativa y responsabilidad en el trabajo en equipo e individual.

SECUENCIA METODOLÓGICA

Momentos	**Estrategias**	**Métodos/ Técnicas**	**Recursos**	**Duración**
Motivación	○ Recoger saberes previos sobre el análisis y resolución de algoritmos, generando el código fuente y ejecutándolo.	Expositivo Relación D-A Lluvia de ideas	PC USB Data	20 min
Proporcionar Información	○ Creación del código fuente; Depuración del código. ○ Los estudiantes leen en forma individual la separata. ○ Presentación de diapositivas con los temas y se entrega el archivo.	Estudio dirigido	Separatas USB, Laptop Multimedia PPT PDF	40 min
Desarrollar Práctica dirigida	○ Desarrollo de ejercicios modelos, usando el DEV C++ para generar el código fuente y comprobar el algoritmo.	Práctica individual Relación D-A	Guía de práctica. PC	45 min
Resolución de Problemas y transferencia	○ Se da respuesta a las interrogantes propuestas por los alumnos con respecto al tema.	Inductivo Relación D-A Tutoría	Revistas Libros	45 min
Evaluación	○ Resuelven los ejercicios propuestos haciendo uso de la computadora y el lenguaje Dev c ++.	Examen práctico	Ficha de observación	30 min

EVALUACIÓN DE LOS APRENDIZAJES

CRITERIO (S) DE EVALUACIÓN: Elabora y comprueba diagramas de flujo de los procedimientos requeridos

Indicadores	**Técnicas**	**Instrumento**
○ Genera el código fuente de un algoritmo con precisión. ○ Demuestra iniciativa y responsabilidad en el trabajo en equipo e individual.	○ Examen ○ Observación	○ Practica calificada ○ Lista de cotejo

ESTRATEGIA 6:

DENOMINACIÓN: DISEÑA Y CONSTRUYE ALGORITMOS A PARTIR DE TEMA ESPECÍFICO PLANTEADO.

DESCRIPCIÓN:

En esta estrategia se busca que los estudiantes, en grupos organicen información para resolver algoritmos de temas que ya conocen.

Utilizando el internet deben buscar enunciados de algoritmos propuestos, aplicar sus conocimientos sobre lógica de programación haciendo uso de los comandos del PSeInt. Cada tema debe ser expuesto por el grupo de trabajo. Cada grupo puede presentar su trabajo con diferentes estructuras.

PROCEDIMIENTO:

- Organizar con anterioridad los grupos de trabajo y asignarles un tema específico a cada uno.
- Pedir previamente a los estudiantes que busquen y organicen la información sobre el tema asignado a cada grupo. Esta información deberán llevarlo luego a la sesión de clase.
- Durante el desarrollo de la sesión de clase, los estudiantes por grupos organizan en el software PSeInt su información recopilada.
- Los estudiantes por grupos exponen su trabajo concluido.
- Los estudiantes reflexionan sobre la calidad de los trabajos elaborados y expuestos.
- El – la docente precisa las conclusiones y recomendaciones finales.

TIEMPO ESTIMADO:

Dos clases de 90 minutos cada una.

Gráfico 6: Estudiante desarrollando un tema específico.

SESIÓN DE APRENDIZAJE 6

ESPECIALIDAD : Computación e Informática
MÓDULO ESPECÍFICO : Desarrollo de software y gestión de base de datos
UNIDAD DIDÁCTICA : **LÓGICA DE PROGRAMACIÓN**
CAPACIDAD TERMINAL: Realizar la lógica de la programación considerando procesos y secuencias.
ACTIVIDAD DE APRENDIZAJE N° 6: Estructuras de control selectivas.

ELEMENTO DE LA CAPACIDAD TERMINAL: Comprobar los diagramas realizados para su validación.	**Lugar**	**Horas Pedagógicas**
	LABORATORIO (X) TALLER ( )	2
	AULA (X)	

CONTENIDOS

Procedimientos	**Conceptos**	**Actitudes**
Realizar una breve introducción sobre el tema.	Resuelve ejercicios sobre expresiones lógicas. Resuelve algoritmos de estructuras selectivas.	○ Usa los recursos digitales con responsabilidad.

SECUENCIA METODOLÓGICA

Momentos	**Estrategias**	**Métodos/ Técnicas**	**Recursos**	**Duración**
Motivación	○ Recoger saberes previos sobre cómo resolver algoritmos con expresiones lógicas.	Expositivo Relación D-A Lluvia de ideas	PC USB Data	20 min
Proporcionar Información	○ Entregar los ejercicios y asesor el desarrollo de los mismos a través del uso del software PSeInt.	Estudio dirigido	Separatas USB Multimedia PPT	40 min
Desarrollar Práctica dirigida	○ Usando el software PSeInt generar los diagramas de flujo.	Práctica individual Relación D-A	Guía de práctica. PSeInt	45 min
Resolución de Problemas y transferencia	○ Se da respuesta a las interrogantes propuestas por los alumnos con respecto al tema.	Inductivo Relación D-A Tutoría	Revistas Libros	45 min
Evaluación	○ Resuelven los ejercicios propuestos haciendo uso de la computadora y el software PSeInt.	Examen práctico	Ficha de observación	30 min

EVALUACIÓN DE LOS APRENDIZAJES

CRITERIO (S) DE EVALUACIÓN: Elabora y comprueba diagramas de flujo de los procedimientos requeridos

Indicadores	**Técnicas**	**Instrumento**
○ Genera el código fuente de un algoritmo usando el software PSeInt. ○ Usa los recursos digitales con responsabilidad.	○ Examen ○ Observación	○ Practica calificada ○ Lista de cotejo

ESTRATEGIA 7:

DENOMINACIÓN: DISEÑA Y CONSTRUYE ALGORITMOS A PARTIR DE DESARROLLO DE ACTIVIDADES PROPIAS DEL PROGRAMA.

DESCRIPCIÓN:

En esta estrategia introduciremos el algoritmo haciendo uso de los comandos del programa, luego lo guardaremos y lo ejecutaremos haciendo pruebas de escritorio; si hay errores debemos solucionarlos y volver a ejecutarlo. Luego de que el algoritmo ya está sin errores generamos el diagrama de flujo.

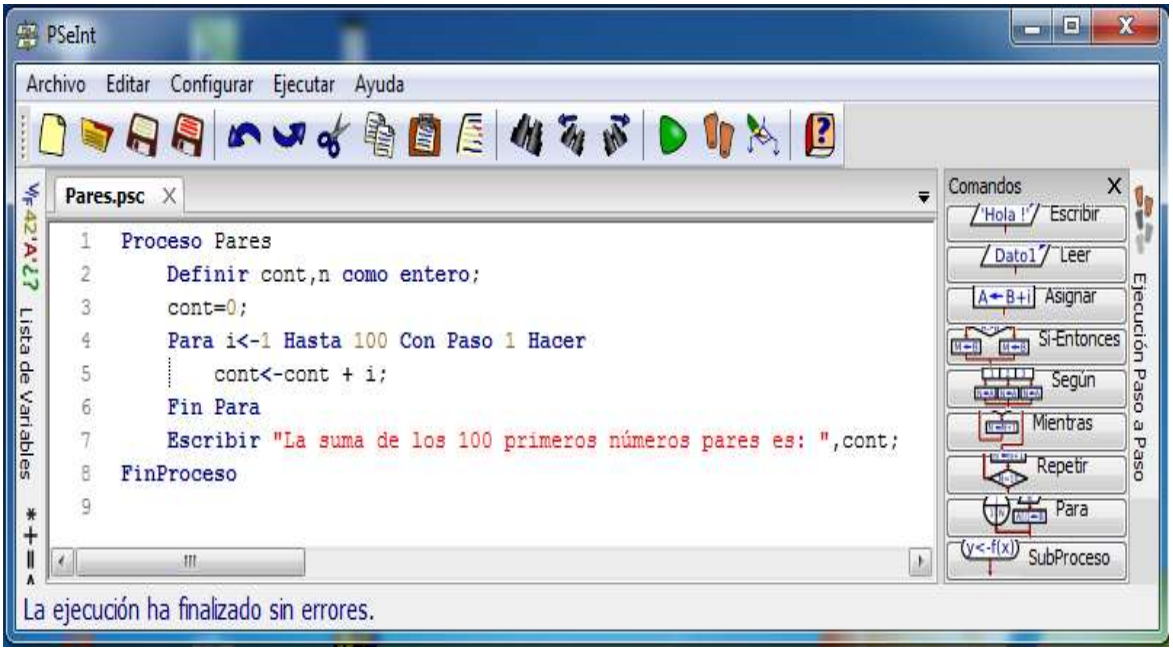
PROCEDIMIENTO:

- Leemos el enunciado, buscamos la mejor solución.
- Ejecutamos el PSeInt y escribimos el algoritmo usando los recursos propios del software PSeInt.
- Lo guardamos en un lugar del disco duro.
- Ejecutamos y hacemos pruebas de escritorio.
- Finalizamos generando el diagrama de flujo.

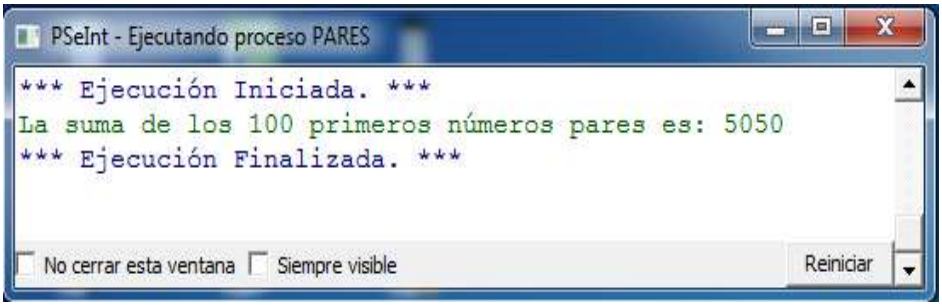
Ejemplos: *Desarrolle los siguientes algoritmos y genere sus diagramas de flujo:*

a. Contar los 100 primeros números pares.

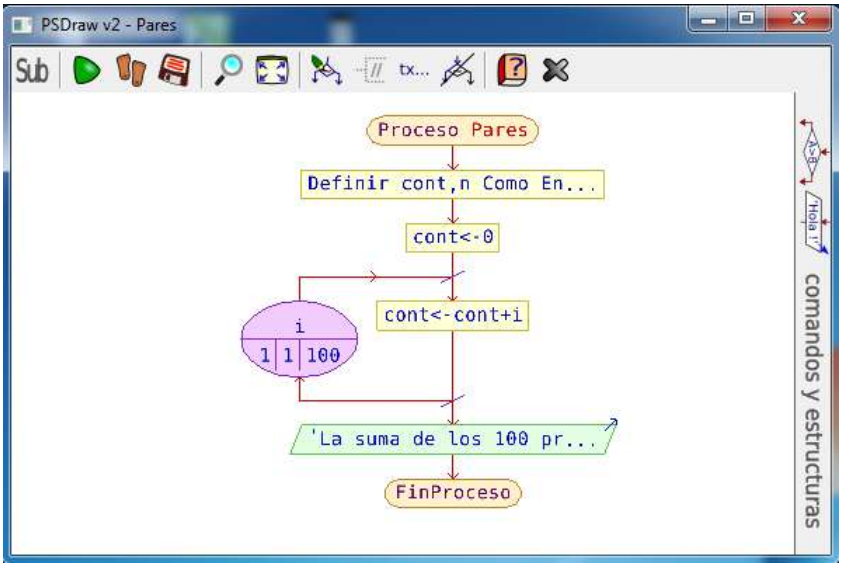
1. Diseñando el algoritmo.



2. *Ejecutando el algoritmo construido.*

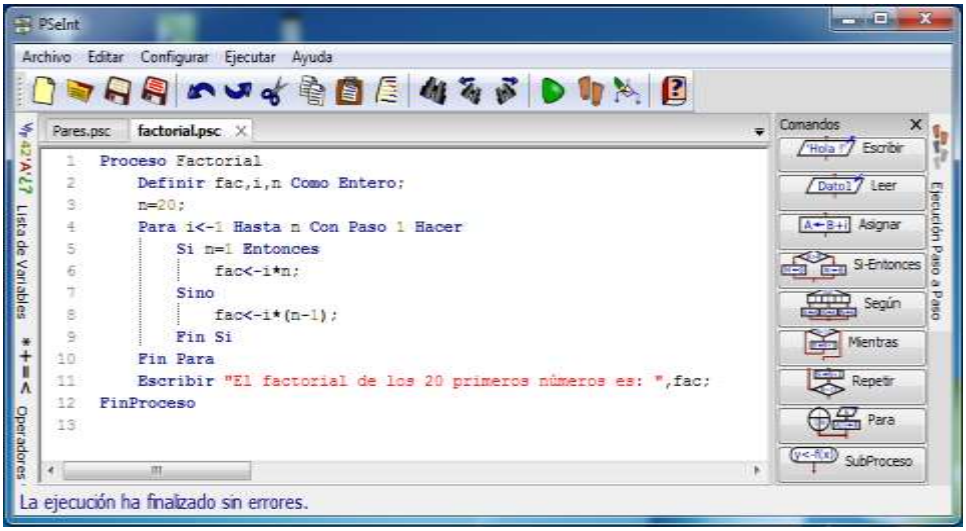


3. *Diagrama de flujo*

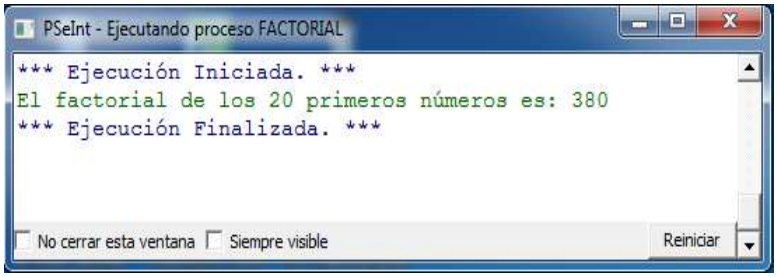


b. *Hallar la factorial de los 20 primeros números naturales.*

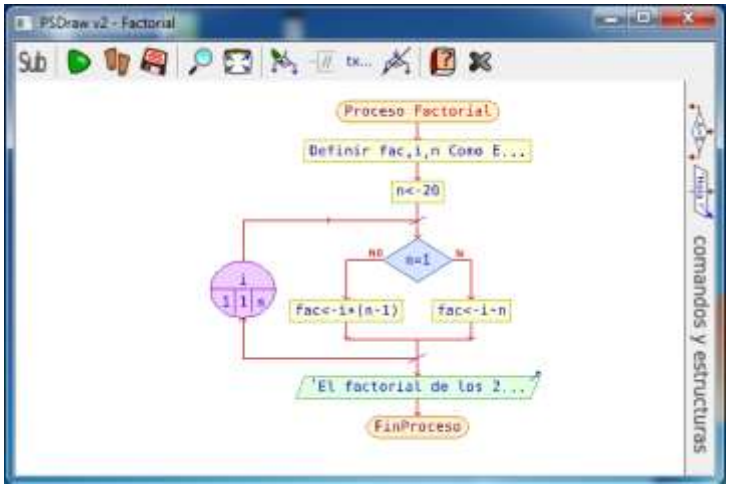
1. *Diseñando el algoritmo.*



2. *Ejecutando el algoritmo construido.*

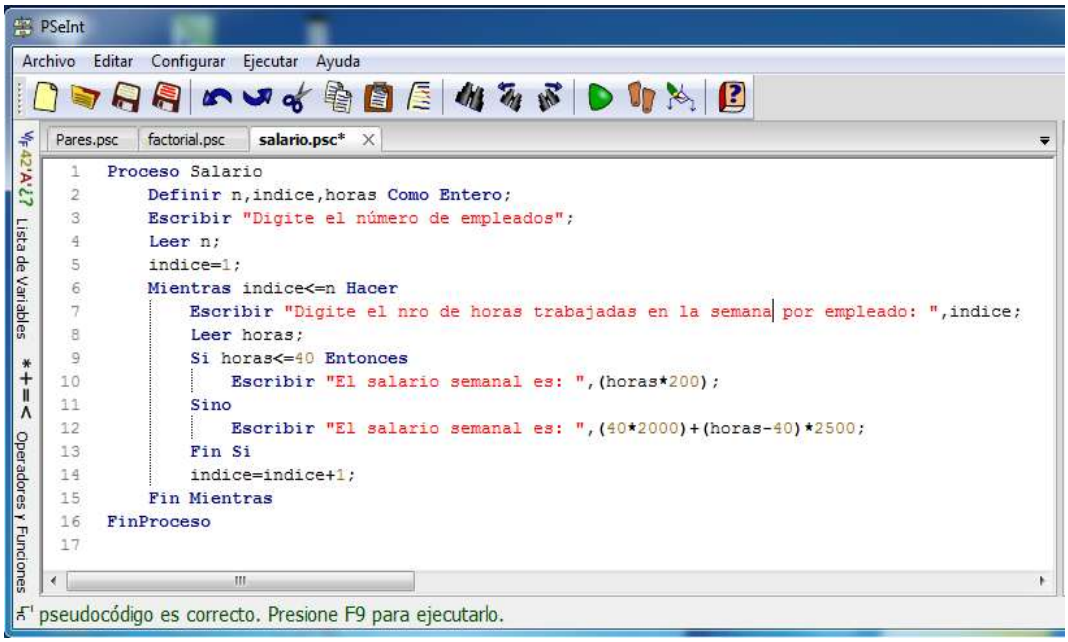


3. *Diagrama de flujo.*

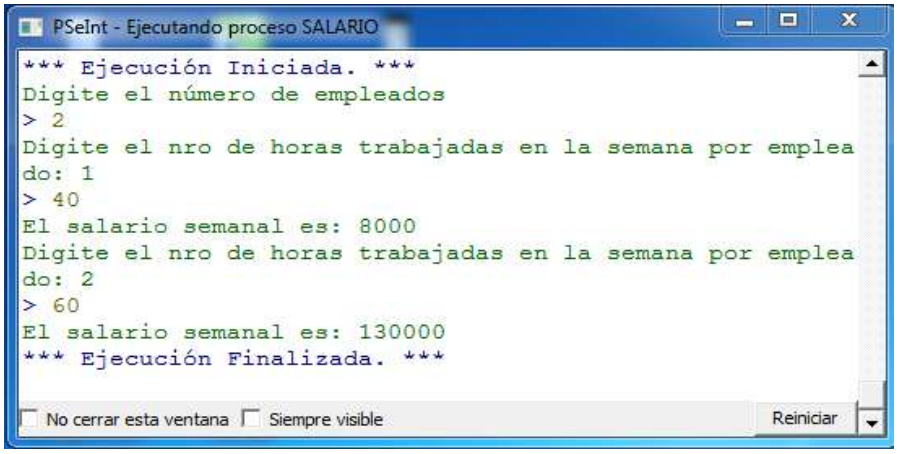


c. ***En una empresa se requiere calcular el salario semanal de cada uno de los "n" obreros que laboran en ella. El salario se obtiene de la siguiente manera: si el obrero trabaja 40 horas o menos se le paga 200 por hora. Si el obrero trabaja más de 40 horas, se le paga 2000 por cada una de las primeras horas y 2500 por cada hora extra.***

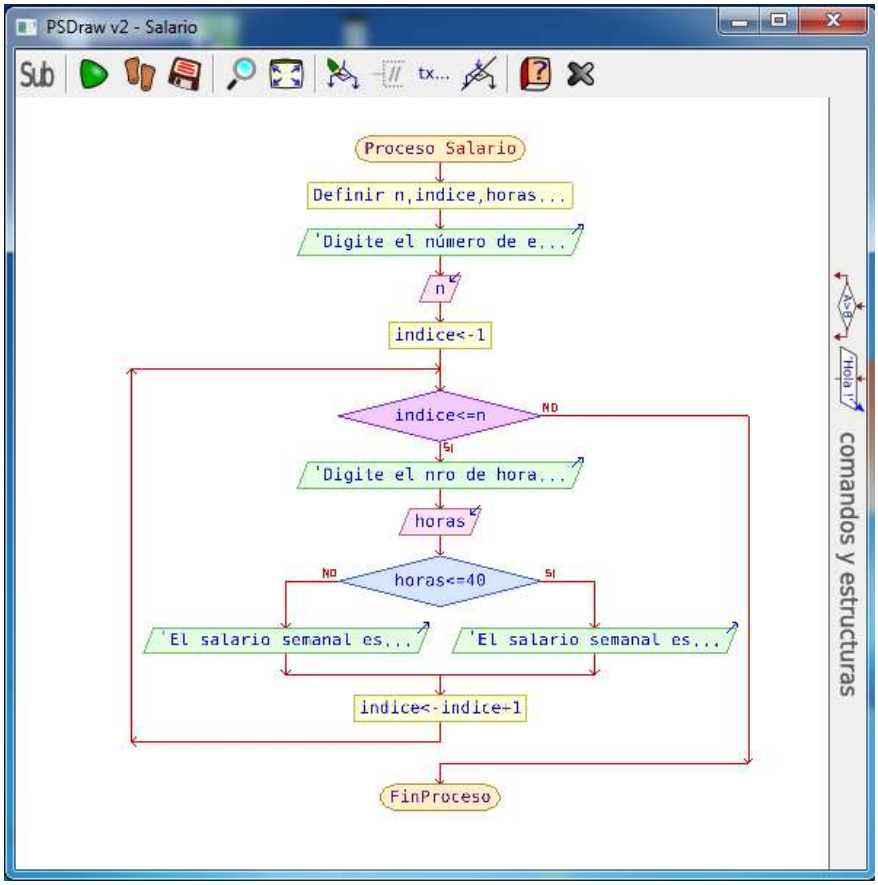
1. *Diseñando el algoritmo.*



2. *Ejecutando el algoritmo construido.*



3. *Diagrama de flujo.*



TIEMPO ESTIMADO:

Dos clases de 90 minutos.

SESIÓN DE APRENDIZAJE 7

ESPECIALIDAD : Computación e Informática
MÓDULO ESPECÍFICO : Desarrollo de software y gestión de base de datos
UNIDAD DIDÁCTICA : **LÓGICA DE PROGRAMACIÓN**
CAPACIDAD TERMINAL: Realizar la lógica de la programación considerando procesos y secuencias.
ACTIVIDAD DE APRENDIZAJE N° 7: Estructuras de control repetitivas.

ELEMENTO DE LA CAPACIDAD TERMINAL: Comprobar los diagramas realizados para su validación.	**Lugar**	**Horas Pedagógicas**
	LABORATORIO (X) TALLER ( )	2
	AULA (X)	

CONTENIDOS

Procedimientos	**Conceptos**	**Actitudes**
Realizar una breve introducción sobre el tema. Mostrar el producto de la actividad de aprendizaje.	Resuelve ejercicios sobre contadores y acumuladores. Resuelve algoritmos de estructuras repetitivas. El software PSeInt	○ Usa los recursos digitales con responsabilidad.

SECUENCIA METODOLÓGICA

Momentos	**Estrategias**	**Métodos/ Técnicas**	**Recursos**	**Duración**
Motivación	○ Recoger saberes previos sobre cómo resolver algoritmos con estructuras repetitivas.	Expositivo Relación D-A Lluvia de ideas	PC USB Data	20 min
Proporcionar Información	○ Entregar los ejercicios y asesor el desarrollo de los mismos a través del uso del software PSeInt.	Estudio dirigido	Separatas USB Multimedia PPT	40 min
Desarrollar Práctica dirigida	○ Usando el software PSeInt generar los diagramas de flujo. ○ Crea algoritmos en PSeInt.	Práctica individual Relación D-A	Guía de práctica. PSeInt	45 min
Resolución de Problemas y transferencia	○ Se da respuesta a las interrogantes propuestas por los alumnos con respecto al tema.	Inductivo Relación D-A Tutoría	Revistas Libros	45 min
Evaluación	○ Resuelven los ejercicios propuestos haciendo uso de la computadora y el software PSeInt.	Examen práctico	Ficha de observación	30 min

EVALUACIÓN DE LOS APRENDIZAJES

CRITERIO (S) DE EVALUACIÓN: Elabora y comprueba diagramas de flujo de los procedimientos requeridos

Indicadores	**Técnicas**	**Instrumento**
○ Genera el código fuente de un algoritmo usando estructuras repetitivas en el software PSeInt. ○ Usa los recursos digitales con responsabilidad.	○ Examen ○ Observación	○ Practica calificada ○ Lista de cotejo

3.3.6. CRONOGRAMA DE SESIONES

N° Sesión	Denominación	Fecha	Responsables
1	Pensando lógicamente.	29/05/2015	Ing. Danny Magallyt Chuquizuta Herrera. Ing. David Rodriguez Medina.
2	Pensando computacionalmente.	05/06/2015	
3	El arte de la programación.	12/06/2015	
4	Usando las estructuras secuencia, decisión y ciclos.	19/06/2015	
5	Mis videos de aprendizaje.	26/06/2015	
6	El entorno gráfico de PSeInt.	03/07/2015	
7	Escribiendo algoritmos en PSeInt.	10/07/2015	
8	Usando los comandos de PSeInt.	17/07/2015	
9	Ejecutando los algoritmos.	24/07/2015	
10	Generando diagramas de flujo con PSeInt.	31/07/2015	

3.3.7. MEDIOS Y MATERIALES EDUCATIVOS

Descripción			
Humanos	**Cantidad**	**Económicos**	**Cantidad**
Docentes	2	Proyector	1
		Pizarra digital	1
		Videos	10
		Papel Bond	1 ciento
Estudiantes	16	Lápices	2 cajas
		Plumones	2 cajas
		Computadoras	35
		Copias	1 millar

3.3.8. PRESUPUESTO DE LA PROPUESTA

ÍTEMS		SUB TOTAL
BIENES	**CANTIDAD**	
1. Bibliografía. (libros)	4	200.00
2. Material de escritorio: Papelotes, Plumones, Hojas bond, Lápices.	 500 20 1000 24	70.00
3. Computadoras/ laptops.	35	900.00
4. Anillados.	35	350.00
5. Fotocopias.	1000	50.00
6. Tipeos e impresiones	100	20.00
7. Internet.	35	90.00
8. Movilidad local y trabajo de campo.	96	50.00
9. Refrigerios.	37	100.00
TOTAL		**1830.00**

3.3.9. EVALUACIÓN DEL PROGRAMA

Para evaluar el programa de estrategias didácticas, se utilizó una guía de observación, orientada a medir el nivel de desarrollo de la competencia diseño y construcción de algoritmos a partir del uso del software PSeInt (anexo 1); también se utilizó un test, para medir el desarrollo de la competencia del diseño y construcción de algoritmos (anexo 2), ambos instrumentos se aplicaron a los estudiantes del grupo de estudio al concluir la fase de aplicación de las estrategias didácticas.

Tanto la guía de observación, así como el test, han sido elaborados teniendo como base los siguientes parámetros de evaluación:

Dimensiones	Indicadores	Índice	Técnicas e instrumentos
Condiciones para escribir un algoritmo. Aspectos básicos de la calidad del software.	Creatividad Confianza Motivación Precisión Funcionalidad Usabilidad Eficiencia Portabilidad	Excelente Muy bueno Bueno Regular Deficiente	Observación/ guía de observación Test

CAPÍTULO IV

Conclusiones, Recomendaciones, Referencias Bibliográficas y Anexos

CAPÍTULO IV
CONCLUSIONES, RECOMENDACIONES, REFERENCIAS BIBLIOGRÁFICAS Y ANEXOS

4.1. CONCLUSIONES

- ✓ Se determinó que los estudiantes antes de la aplicación del programa de estrategias didácticas, tenían un nivel deficiente en el desarrollo de la capacidad del diseño y construcción de algoritmos de la unidad didáctica de lógica de programación, posteriormente el grupo de estudio mejoro significativamente su aprendizaje.

- ✓ Los estudiantes inicialmente no tenían ningún conocimiento del software libre PSeInt, sin embargo después de aplicar las estrategias didácticas a los estudiantes del grupo de estudio, lograron un nivel de dominio significativo.

- ✓ Al aplicar las estrategias didácticas usando el software PSeInt, basado en los principios de las teorías del Aprendizaje por Descubrimiento de BRUNER, del Conocimiento Operante de SKINNER, del Aprendizaje Significativo de Ausubel, de PIAGET y el arte de la programación de KUNTH, el desarrollo de la capacidad del diseño y construcción de algoritmos en la unidad didáctica de lógica de programación mejoró en los estudiantes, validándose la hipótesis del trabajo de investigación.

4.2. RECOMENDACIONES

- ✓ A los docentes que dictan las unidades didácticas de programación, utilizar en el desarrollo de las sesiones de aprendizaje los recursos interactivos virtuales, a fin de motivar la participación activa de los estudiantes en su proceso de aprendizaje, en este ámbito son de gran ayuda las estrategias didácticas con base en el uso del software PSeInt diseñadas en esta investigación.

- ✓ Las investigaciones en el ámbito del diseño y construcción de algoritmos en lógica de programación son muy escasas, por lo que los investigadores deben orientar sus investigaciones en esta línea, a fin de contribuir a la mejora del desarrollo del proceso enseñanza aprendizaje en la unidad didáctica de lógica de programación.

- ✓ Al personal directivo del Instituto "Perú- Japón" de Chachapoyas, difundir los aportes de esta investigación y aplicarlo en todos los ciclos de computación e informática, la aplicación de estrategias didácticas, para mejorar los aprendizajes de los estudiantes de la unidad didáctica de lógica de programación.

- ✓ A todas las personas que lean este trabajo de investigación, hacer llegar a la autora sus observaciones y sugerencias de mejora, a fin de que se pueda enriquecer y profundizar en investigaciones afines en el futuro.

- ✓ El papel de los adultos para facilitar el aprendizaje, del diseño y construcción de algoritmos, es proporcionar experiencias significativas y variadas donde se empleen estrategias didácticas para que el estudiante logre su aprendizaje significativo.

4.3. REFERENCIAS BIBLIOGRÁFICAS

4.3.1. BIBLIOGRAFÍA

- ✓ Ausubel, D.P. (1968) *Psicología de la educación*: Una visión cognitiva. New York: Holt, Rinehart & Winston.
- ✓ Bezanilla, M. y Martínez, J. (1996). *Bases técnico-pedagógicas para la elaboración de software educativo.* España: Universidad Nacional de Educación a Distancia.
- ✓ Balarin (2011) *Las Políticas TIC en los sistemas educativos de América Latina:* CASO PERÚ edición Fondo de las Naciones Unidas para la Infancia (UNICEF), junio de 2013.
- ✓ Barajas, M. (2003). *La tecnología educativa en la enseñanza superior. Entornos virtuales de aprendizaje.* Madrid. Editorial Mc Graw Hill.
- ✓ Barriga A., Frida y Hernández R. (1998) Gerardo. *Estrategias docentes para un aprendizaje significativo.* México: McGraw-Hill.
- ✓ Cabero, J. (1996). *Nuevas tecnologías, comunicación y educación EDUTEC.* Revista electrónica de tecnología educativa. recuperado de: http://www.uib.es/depart/gte/revelec1.html.
- ✓ Castaño, C; Maíz, I.; Palacio, G. y Villarroel, J. (2008). *Habilidades y competencias en la web 2.0. Prácticas educativas en entornos web 2.0.* Madrid: Síntesis.
- ✓ Da Rosa, F. y Heinz, F. (2007). *Guía práctica sobre software libre: su selección y aplicación local en américa LATINA y el Caribe.* Montevideo, Uruguay: UNESCO. Fandos.
- ✓ Díaz, A. & Hernández, G. (1999). *Estrategias de enseñanza para la promoción de aprendizajes significativos.* McGraw Hill. México.
- ✓ Ertmer, P. (1999). *Estrategias para la integración de la tecnología. Tecnología educativa de investigación y desarrollo 47(4): 47-61*. Recuperado en: http://mingaonline.uach.cl/scielo.php?pid=S0718070520020 00100011& script=sci_arttext.
- ✓ Fernández, J. (1975). E*nseñanza programada y tecnología educacional.* Didáctica (unidad IV) Madrid: UNED.
- ✓ Fernández, A. & Sarramona, J, (1987). *Didáctica y tecnología de la educación.* Diccionario de ciencias de la educación. Madrid: Anaya.

- García, J. (2004). *Ambientes con recursos tecnológicos.* Universidad Estatal a Distancia, San José, Costa Rica. Recuperado de http://books.google.com.co/books?id=VeEL_gDVX6AC&pg=PA106&lpg=PA106&dq=ambientes+con+recurso+tecnologicos&source=bl&ots=tBymjRPUYY&sig=LgBlwUEvbTfGS67y0MAZbYKOptk&hl=es&ei=nd_kTo3jG4KMgwfUm6jnBQ&sa=X.
- Hernández, R. Fernández, C & Bautista, P. (2010). *Metodología de la investigación.* Ed. Mc Graw Hill Quinta Edición. México.
- Iglesias, G. (2012). *Enfoques didácticos de enseñanza de la redacción y composición.* Universidad Metropolitana de Ciencias de la Educación.
- Knuth, D. E. (1986) *El arte de programar ordenadores, Vol. 1. Algoritmos fundamentales.* Ed. Reverté.
- Moreno, N. (2011). *Las TIC como herramientas para el desarrollo del aprendizaje autónomo del español como segunda lengua Universidad de Málaga*.
- Sancho, J. (2008). *Educación en la Era de la Información.* Revista Cuadernos de Pedagogía N° 253. Barcelona: Ediciones B.
- Siemens, J. (2004), *Conectivismo: Una teoría de aprendizaje para la era digital.* Creative Commons. Recuperado de: www.diegoleal.org/docs/2007/siemens(2004)-Conectivismo.doc.
- Urbina, S. (1999). *Informática y teorías del aprendizaje.* Universitat de les llles Ballears. Recuperado de http://tecnologiaedu.us.es/nweb/htm/pdf/gte41.
- Vygotsky, L. (1979). *Desarrollo de los procesos psicológicos superiores.* Barcelona: Grijalbo.
- Vygotsky, L. (1995). *La formación social de la gente.* Buenos Aires: Paidos.
- Wirth, N. (1986) *Algoritmos + estructuras de datos = programas*, Ediciones del Castillo, Madrid.

4.3.2. LINKOGRAFÍA

- http://www.ses.unam.mx/curso2014/pdf/LopezSegrera2008.pdf
- http://blogs.worldbank.org/voices/es/diez-tendencias-sobre-el-uso-de-la-tecnologia-en-la-educacion-en-los-paises-en-desarrollo
- http://sourceforge.net/projects/pseint/files/20150920/pseint-w32-20150920.exe/download?use_mirror=ufpr&download=

- ✓ http://www.monografias.com/trabajos16/sistemas-distribuidos/sistemas-distribuidos.shtml#ixzz2a79DadCr
- ✓ https://es.wikipedia.org/wiki/Heur%C3%ADstica_(inform%C3%A1tica)
- ✓ http://laresolucindeproblemasmatematicos.blogspot.pe/2009/09/223-teorias-heuristicas-que-sustentan.html.
- ✓ http://pseint.sourceforge.net/

4.4. ANEXOS

4.4.1. Anexo 1: GUÍA DE OBSERVACIÓN

I. DATOS GENERALES

1.1. **Institución Educativa:** I.E.S.T. PÚBLICO "PERÚ - JAPÓN".

1.2. **Ciclo:** III

1.3. **Fecha y Hora:** ……..……..

1.4. Nombre del Observador: Ing. Danny M. Chuquizuta Herrera

II. OBJETIVO: Recoger información acerca del nivel de conocimiento y manejo del software PSeInt en los estudiantes del III ciclo de computación e informática del I.E.S.T. PÚBLICO "PERÚ - JAPÓN".

Nº de Orden	APELLIDOS Y NOMBRES	INDICADORES REFERENTES A OBSERVAR							PUNTAJE TOTAL	NIVEL DE LOGRO
		A	B	C	D	E	F	G		
PROMEDIO GENERAL										

INDICADORES	
A	Diseña y construye algoritmos a partir de videos mostrados.
B	Diseña y construye algoritmos a partir de software libre.
C	Diseña y construye algoritmos a partir de dinámicas de grupo.
D	Diseña y construye algoritmos a partir de ayudantes de catedra.
E	Diseña y construye algoritmos a partir de material didáctico.
F	Diseña y construye algoritmos a partir de tema específico planteado.
G	Diseña y construye algoritmos a partir de desarrollo de actividades propias del programa.

PUNTAJE PARCIAL	SIGNIFICADO
1	No demuestra ningún signo de desarrollo.
2	Mínimo nivel de desarrollo.
3	Regular nivel de desarrollo.
4	Alto nivel de desarrollo.
5	Excelente desarrollo

PUNTAJE PARCIAL	SIGNIFICADO
1 – 20	Deficiente
21 – 30	Regular
31 – 40	Bueno
41 – 50	Muy bueno
51 – 60	Excelente

4.4.2. Anexo 2: TEST PARA MEDIR EL NIVEL DE DESARROLLO DE LA CAPACIDAD DEL "DISEÑO Y CONSTRUCCIÓN DE ALGORITMOS" EN LA UNIDAD DIDÁCTICA LÓGICA DE PROGRAMACIÓN.

I. DATOS INFORMATIVOS

Instituto Superior Tecnológico Público "Perú - Japón"

Unidad Didáctica: Lógica de Programación

Ciclo Académico: III

Fecha: …….....

II. INDICACIONES

Estimado (a) estudiante, a continuación se le presenta varios ítems, por favor responde de manera consciente y responsable, ya que servirá para obtener información que será utilizada con fines de mejorar el proceso de enseñanza aprendizaje dentro de la unidad didáctica.

1. **Desarrolle los siguientes algoritmos usando Pseudocódigo.**
   a. *Suponga que tiene una tienda y desea registrar sus ventas por medio de una computadora, diseñe el algoritmo que lea por cada cliente:*
      1. *El monto de la venta,*
      2. *Calcule e imprima el IGV,*
      3. *Calcule e imprima el total a pagar,*
      4. *Lea la cantidad con que paga el cliente,*
      5. *Calcule e imprima el cambio,*
      6. *Al final del día imprima la cantidad de dinero que debe haber en la caja.*

   b. *Hallar la factorial de los 20 primeros números naturales.*

   c. *Hallar al área de un cuadrado.*

2. **Generar los diagramas de flujo por cada algoritmo desarrollado.**

   …….....

   …….....

   …….....

   …….....

   …….....

   …….....

4.4.3. Anexo 3: Sílabo

INSTITUTO DE EDUCACIÓN SUPERIOR TECNOLÓGICO PÚBLICO

"PERÚ - JAPÓN"

SÍLABO DE LÓGICA DE PROGRAMACIÓN

I. INFORMACIÓN GENERAL

1. **Carrera Profesional** : Computación e Informática
2. **Tipo de Módulo** : Técnico Profesional
3. **Módulo Específico** : Desarrollo de software y gestión de base de datos
4. **Unidad Didáctica** : Lógica de Programación
5. **Semestre Académico** : 2016
6. **Semestre Curricular** : III Ciclo
7. **Duración del Semestre** : 5 Semanas
8. **Carga Horaria**
   - **8.1 N° Horas Semestral** : 10
   - **8.2 N° Horas Semanal** : 2
9. **Turno** : Tarde
10. **Horario** : lunes – viernes: 8:00 – 11:30 a.m.
11. **Periodo de Ejecución**
    - **11.1 Fecha de Inicio** : 08/03/2015
    - **11.2 Fecha de Término** : 01/08/2015
12. **Créditos** : 4
13. **Docente Responsable** : Danny Magallyt Chuquizuta Herrera
    Ing. en Computación e Informática
14. **Colegiatura N°** : 124370
15. **Celular** : 916620795
16. **Correo Electrónico** : chuhedama@gmail.com

II. COMPETENCIA DE LA CARRERA PROFESIONAL

Planificar, implementar y gestionar el uso de las tecnologías de la información y la comunicación de una organización, a partir del análisis de sus requerimientos, teniendo en cuenta los criterios de calidad, seguridad y ética profesional, propiciando el trabajo en equipo.

III. CAPACIDADES TERMINALES Y CRITERIOS DE EVALUACIÓN

Capacidad Terminal	Criterios de Evaluación
Realizar la lógica de la programación considerando procesos y secuencias.	○ Realizar algoritmos de procesos y subprocesos. ○ Elabora y comprueba diagramas de flujo de los procedimientos requeridos.

IV. ORGANIZACIÓN DE ACTIVIDADES Y CONTENIDOS BÁSICOS

SEM.	ELEMENTOS DE LA CAPACIDAD TERMINAL	ACTIVIDADES DE APRENDIZAJE	CONTENIDOS BÁSICOS	TAREAS PREVIAS
1 – 2	○ Utiliza el pseudocódigo para representar algoritmos procedimentales.	○ Elabora algoritmos secuenciales y condicionales	Introducción a la programación. ○ Datos y expresiones, definición de variables y constantes. Herramientas de programación. Seudocódigo. ○ Ejemplos de algoritmos secuenciales y condicionales. ○ Operadores, algoritmos, concepto, características. Ejemplos.	○ Conceptualiza las herramientas necesarias para la resolución de problemas por computadora.

3 - 9		○ Elabora algoritmos repetitivos finitos e infinitos ○ Elabora algoritmos con uso de arreglos. ○ Diseña y ejecuta diagramas de flujo de algoritmos procedimentales.	Análisis del problema Especificaciones de entradas y salidas. Ejemplos. ○ Diseño de algoritmos, diseño descendente, refinamientos por pasos. ○ Arreglos, declaración, creación. ○ Clases, operaciones. Ejemplos. ○ Herramientas de programación Diagramas de flujo Casos aplicativos.	○ Desarrolla la estructura para la resolución de un problema computacional ○ Reconoce los componentes para el análisis del problema. Conceptualiza las herramientas necesarias para el diseño de algoritmos.
10 - 12	○ Diseña diagramas de flujo y utiliza el lenguaje de programación C++ para implementar las acciones.	○ Elabora algoritmos procedimentales y verifica su validez generando código C++ correspondiente.	Resolución de algoritmos en la computadora, codificación del algoritmo, ejecución y comprobación del programa.	○ Conceptualiza los métodos formales de verificación de programas.
13 – 14		○ Elabora aplicaciones de tipo procedimental empleando FDU.	Ciclo de vida de los objetos: creación, uso, destrucción.	○ Conceptualiza los factores de la calidad del software.
15		○ Diseña clases para abstraer objetos de un sistema real.	Tipos de datos: tipos primitivos. Clases String, declaración de atributos, expresiones, asignación.	○ Capacidad para el desarrollo del trabajo de investigación.
16 – 17		○ Diseña aplicaciones con clases y subclases para	Métodos, definición, declaración de	○ Establece un criterio para la adecuada

		abstraer proceso de un sistema real	variables paso de parámetros.	utilización de estructuras en la resolución de un problema por computadora.
18	○ Evaluación de recuperación.			

V. METODOLOGÍA

Para el desarrollo de la Unidad Didáctica se aplicará los siguientes procedimientos didácticos:

- ✓ **Clases Teóricas:** Con exposición del Docente y la participación activa de los estudiantes.
- ✓ **Prácticas:** Se irán resolviendo aplicaciones mediante desarrollo de prácticas dirigidas y calificadas.
- ✓ **Intervenciones Orales:** Con presentaciones y exposiciones de los estudiantes en forma individual y grupal.
- ✓ **Exposiciones:** Sustentación de trabajos.
- ✓ **Exámenes escritos:** Resolución de preguntas.
- ✓ **Asesoría:** Se asesora la aplicación correcta de los conocimientos teóricos en el desarrollo de sus proyectos de base de datos.

VI. EVALUACIÓN

6.1. Indicadores e instrumentos de evaluación

○ La evaluación de los aprendizajes se centra en el logro de las capacidades terminales, a través de los criterios e indicadores de evaluación (expresados en la Ficha de Actividad), que son los estándares mínimos de calidad.
○ La evaluación utiliza el sistema vigesimal. El calificativo mínimo aprobatorio es 13 (trece). La fracción 0.5 o más se considera como una unidad a favor del estudiante.
○ El estudiante que en la evaluación de la Capacidad Terminal programada en la Unidad Didáctica, obtenga nota desaprobatoria igual o menor que 12 (doce), debe participar en un Programa de Actividades de Recuperación que será programado para la 18ª semana del 09 al 10 de Agosto del 2012, implementado y ejecutado por el docente, con carácter obligatorio y comprende acciones tales como: trabajos prácticos, actividades de autoaprendizaje y otras acciones formativas relacionadas con la capacidad terminal desaprobada.
○ Al término del semestre los estudiantes que obtuvieran notas entre 10 (diez) y 12 (doce) participarán del Programa de Recuperación, ejecutado por el docente y con carácter obligatorio, serán evaluados por un JURADO. El estudiante que obtuviera nota menor a 13 (trece) repite la Unidad Didáctica.
○ El estudiante que acumule inasistencias injustificadas en número igual o mayor al 30% del total de horas programadas en la Unidad Didáctica será desaprobado automática, anotándose en el registro y acta la nota 00 y en observaciones DPI (desaprobado por inasistencia).
○ El estudiante que deja de asistir sin solicitar licencia durante 20 (veinte) días hábiles consecutivos, se considerará como abandono de estudios profesionales, perdiendo su condición de estudiante.

6.2. Requisitos de aprobación

○ La nota aprobatoria mínima en cada evaluación parcial o final de una actividad, módulo o unidad didáctica es 13.
○ La nota final de cada actividad se determina con el promedio de las evaluaciones de lo cognitivo, procedimental y actitudinal.
○ Se considerará una asistencia mínima requerida del 70%
○ Presentar trabajos de investigación oportunamente, participación activa en las sesiones de clase.
○ Tener evaluación en todas las capacidades demostrando respeto, puntualidad y honradez

La unidad didáctica consta de una capacidad terminal. La nota final es el promedio de los tres criterios de evaluación.

Indicador 1: Contenidos conceptuales (conocimientos o conceptos; teoría)

Indicador 2: Contenidos procedimentales (ejercicios y prácticas)

Indicador 3: Contenidos actitudinales (comportamiento y valores)

$$\textbf{Criterio de Eval. (CE)} = \frac{\textbf{Indicador 1 + Indicador 2 + Indicador 3}}{\textbf{3}}$$

$$\textbf{Cap. Terminal (CT)} = \frac{\textbf{Criterio de Eval. 1 + Criterio de Eval. 2}}{\textbf{2}}$$

Promedio Final = Capacidad Terminal 1 = LOGRO (L)

VII. RECURSOS DIDÁCTICOS

○ Para las clases que se requiera el PROYECTOR MULTIMEDIA y UN PC, le solicitamos al encargado nos proporcione a tiempo el equipo.
○ Pizarra acrílica, plumones de colores

VIII. RECURSOS BIBLIOGRÁFICOS

IMPRESOS			
Título de la obra	**Autor**	**Editorial**	**Año**
○ **Metodología de la Programación**	Cairo B. Osvaldo	ALFAOMEGA	2003
○ **Programación en C++ Algoritmos, Estructuras de Datos y Objetos.**	Joyanes Aguilar Luis	Mc Graw Hill	2000
○ **Turbo C/C++ Manual de Referencia**	Herbert Schildt	Mc Graw Hill	1992
DIGITALES (URL)			
○ **http://www.lawebdelprogramador** ○ **http://www.microsoft.com**			

Chachapoyas, 08 de marzo del 2015

ING DANNY MAGALLYT CHUQUIZUTA HERRERA
CIP N° 124370
DOCENTE

V° B° JEFE DE ÁREA ACADÉMICA

V° B° JEFE DE UNIDAD ACADÉMICA

V° B° DIRECTORA GENERAL

4.4.4. Anexo 4: Tabla de Evaluación de Expertos

TABLA DE EVALUACIÓN DE EXPERTOS

Apellidos y Nombres del Experto:…….....……
Título y/o Grado. Ph. D.: ( ) Doctor: ( ) Magister: ( )
Universidad que obtuvo el post grado:……...……..……..
Fecha:……..

Título del Informe de Tesis

Estrategias didácticas para facilitar el aprendizaje del diseño y construcción de algoritmos de los estudiantes del III ciclo de computación e informática, del I.E.S.T. Público "Perú – Japón", Chachapoyas – 2015.

Mediante la tabla de evaluación de expertos, usted tiene la facultad de evaluar cada una de las preguntas marcando con "X" en las columnas de Si o No. Asimismo, le exhorto en la corrección de los ítems indicando sus observaciones y/o sugerencias, con la finalidad de mejorar la coherencia de las preguntas sobre el uso del software PSeInt en el diseño y construcción de algoritmos.

ÍTEMS	PREGUNTAS	APRECIA		OBSERVACIONES
		SI	NO	
1	¿El instrumento de medición cumple con el diseño adecuado?			
2	¿El instrumento de la recolección de datos tiene relación con el título de la investigación?			
3	¿El instrumento de la recolección de datos se menciona las variables de investigación?			
4	¿El instrumento de la recolección de datos facilitará el logro de los objetivos de la investigación?			
5	¿El instrumento de la recolección de datos se relaciona con las variables de estudio?			
6	¿La redacción de las preguntas es con sentido coherente?			
7	¿Cada una de las preguntas del instrumento de medición se relaciona con cada uno de los elementos de los indicadores?			
8	¿El diseño del instrumento de medición el análisis y procesamiento de datos?			
9	¿Del instrumento de medición son entendibles sus alternativas de respuesta?			
10	¿El instrumento medición será accesible a la población sujeto de estudio?			
11	¿El instrumento medición es clara, precisa y sencilla para que contesten y de esa manera obtener los datos requeridos?			
TOTAL				

SUGERENCIAS

_____

_____

_____

_____
FIRMA DEL EXPERTO

4.4.5. Anexo 5: Constancia.

"Año de la Consolidación del Mar de Grau"

EL DIRECTOR DEL I.E.S.T. PÚBLICO "PERÚ - JAPÓN" DEL DISTRITO DE CHACHAPOYAS, PROVINCIA DE CHACHAPOYAS Y REGIÓN AMAZONAS.

HACE CONSTAR:

CONSTANCIA

QUE LA DOCENTE ING. DANNY MAGALLYT CHUQUIZUTA HERRERA CON DNI Nº 40263111.

HA DESARROLLADO UNA PROPUESTA DENOMINADO

PROGRAMAR ES UN ARTE MUY SENCILLO EN LOS ESTUDIANTES DEL III CICLO DE COMPUTACIÓN E INFORMÁTICA DEL I.E.S.T. PÚBLICO "PERÚ - JAPÓN" DEL DISTRITO DE CHACHAPOYAS, PROVINCIA DE CHACHAPOYAS Y REGIÓN AMAZONAS.

DESDE EL MES DE ABRIL A AGOSTO DEL 2015 CONTRIBUYENDO DE ESTA MANERA AL MEJORAMIENTO DE CALIDAD EDUCATIVA.

SE EXPIDE LA PRESENTE CONSTANCIA A SOLICITUD DE LA PARTE INTERESADA PARA LOS FINES QUE ESTIME CONVENIENTE.

CHACHAPOYAS, SETIEMBRE DEL 2015.

DIRECCIÓN REGIONAL DE EDUCACIÓN AMAZONAS
I.E.S.T. PÚBLICO "PERÚ JAPÓN"
May. Ing. MARIO Y. TORREJÓN ARELLANOS
DIRECTOR GENERAL