



**UNIVERSIDAD TÉCNICA DE BABAHOYO
FACULTAD DE CIENCIAS DE LA EDUCACION**

COMPUTACION

TEMA

TECNICAS PARA EL DESARROLLO DE LA LOGICA EN LA ELABORACIÓN DE PROGRAMAS INFORMÁTICOS Y SU INFLUENCIA EN EL APRENDIZAJE DE LOS ALUMNOS DEL PRIMER AÑO PARALELO “A” DE BACHILLERATO, DEL COLEGIO NICOLÁS INFANTE DIAZ DEL CANTÓN QUEVEDO PROVINCIA DE LOS RIOS AÑO 2.012

TESIS

Previo a la obtención del Título de Licenciado en Ciencias Sociales y de la Educación, mención Informática.

AUTOR

Egdo. Juan Carlos Loor Robles

TUTOR

Ing. Alicia Corina Enríquez

Quevedo - Los Ríos – Ecuador

2012– 2013



**UNIVERSIDAD TÉCNICA DE BABAHOYO
FACULTAD DE CIENCIAS DE LA EDUCACION**

COMPUTACION

TEMA

TECNICAS PARA EL DESARROLLO DE LA LOGICA EN LA ELABORACIÓN DE PROGRAMAS INFORMÁTICOS Y SU INFLUENCIA EN EL APRENDIZAJE DE LOS ALUMNOS DEL PRIMER AÑO PARALELO “A” DE BACHILLERATO, DEL COLEGIO NICOLÁS INFANTE DIAZ DEL CANTÓN QUEVEDO PROVINCIA DE LOS RIOS AÑO 2.012

POSTULANTE: Egdo. Juan Carlos Loor Robles

ÁREA: Computación.

DIRECTOR DE TESIS: Ing. Alicia Corina Enríquez

FECHA DE ENTREGA: Lunes 25 de Febrero del 2.013



**UNIVERSIDAD TÉCNICA DE BABAHOYO
FACULTAD DE CIENCIAS DE LA EDUCACION**

COMPUTACION

CERTIFICADO DE AUTORÍA DE TESIS

Yo, Juan Carlos Loor Robles portador de la cédula No. 1204120917-7, respectivamente estudiante del seminario de tesis previo a la obtención del título de Licenciado en Ciencias Sociales y de la Educación mención Informática, declaro que soy el autor del presente trabajo de investigación, el mismo que es original, auténtico y personal.

Todos los efectos académicos y legales que se desprenden del presente trabajo son de responsabilidad exclusiva del autor.

Egdo. Juan Carlos Loor Robles.



**UNIVERSIDAD TÉCNICA DE BABAHOYO
FACULTAD DE CIENCIAS DE LA EDUCACION**

COMPUTACION

APROBACIÓN DEL TUTOR

En mi calidad de Tutora del Trabajo de Grado, presentado por el Sr. Egdo. Juan Carlos Loor Robles, para optar el título de Licenciado en Ciencias Sociales y de la Educación, cuyo título es: Técnicas para el desarrollo de la lógica en la elaboración de programas informáticos y su influencia en los alumnos del primer año de bachillerato, del colegio Nicolás Infante Díaz del Cantón Quevedo Provincia de Los Ríos en el año 2.012.

Considero que la Tesis reúne los requisitos y méritos suficientes para ser sometidos a la presentación pública y evaluación por parte del jurado examinador que se designe.

En la ciudad de Babahoyo a los 25 días del mes de febrero del 2013.

Ing. Alicia Corina Enríquez



**UNIVERSIDAD TÉCNICA DE BABAHOYO
FACULTAD DE CIENCIAS DE LA EDUCACION**

COMPUTACION

TRIBUNAL DE APROBACIÓN DE LA INVESTIGACIÓN

**Manuel Cárdenas Vivero
DECANO DE LA FACULTAD DE
CIENCIAS SOCIALES Y DE LA EDUCACIÓN**

**Msc. Patricia Camacho Abril
DELEGADO SUB DECANO DE LA FACULTAD DE
CIENCIAS SOCIALES Y DE LA EDUCACIÓN**

**Msc. Javier Camacho Abril
DELEGADO DEL CONSEJO DIRECTIVO**

**Abg. Narcisa Saldivia Gómez
SECRETARIO(A)**



**UNIVERSIDAD TÉCNICA DE BABAHOYO
FACULTAD DE CIENCIAS DE LA EDUCACION**

COMPUTACION

**SECRETARIA DE LA FACULTAD DE CIENCIAS SOCIALES Y DE LA
EDUCACIÓN**

TRIBUNAL DE SUSTENTACIÓN

**El Tribunal de Sustentación otorga al presente trabajo de
investigación la calificación de: _____**

Equivalente a: _____

**Dr. Manuel Cárdenas Vivero
DECANO**

**Msc. Patricia Camacho Abril
DELEGADO SUB DECANO**

**Msc. Javier Camacho Abril
DELEGADO DEL CONSEJO
DIRECTIVO**

**Abg. Narcisa Saldivia Gómez
SECRETARIO(A) QUE
CERTIFICA**

DEDICATORIA

Este trabajo de Tesis está dedicado a Jehová Dios sobre todas las cosas ya que me ha dado paciencia y la sabiduría necesaria para lograr objetivos para finalmente alcanzar una de mis metas, en especial a mi querida madre que ha sido el pilar fundamental ya que con sus consejos y perseverancia permitió que haya llegado a ser un profesional, a mi padre aunque él no esté entre nosotros pero estoy seguro que se sentiría orgulloso del logro alcanzado de su último hijo, a mis pequeños que con su inocencia me daban las fuerzas necesarias cuando estas me traicionaban, a mis hermanas Angelita y Fresia Loor juntas encendieron la chispa que estaba apagada para que continúe con mis estudios, a toda mi familia que de una manera directa o indirecta me apoyaron y estuvieron en los momentos más difíciles de mi vida a todos ellos muchas gracias de todo corazón esto va por ustedes. Familia.

Egdo. Juan Carlos Loor Robles

AGRADECIMIENTO

A todos los que de alguna manera participaron en el presente trabajo de investigación, a Dios creador de la vida y del Universo e inspirador de temor por mantenerme con buena salud para seguir adelante, a mis hermanos por el apoyo constante, mi madre que es la base principal y es lo más importante en mi vida, a mis hijos por darme las fuerzas necesarias cuando estas me traicionan, a los maestros de la Universidad por impartir sus conocimientos dentro y fuera de las aulas de clase, a todos quienes han hecho posible que haya alcanzado esta meta.

Juan Carlos Loor Robles

INDICE

PORTADA.....	i
TEMA.....	ii
CERTIFICACION DE AUTORIA DE TESIS.....	iii
APROBACION DEL TUTOR.....	iv
TRIBUNAL DE APROBACION DE LA INVESTIGACION.....	v
CALIFICACION.....	vi
DEDICATORIA.....	vii
AGRADECIMIENTO.....	viii
INDICE.....	ix

CAPITULO I

1. CAMPO CONTEXTUAL DE LA PROBLEMÁTICA

1.1. Teoría De La Programación

- Mecanismo:
- Explicación:
- Objetivo:
- Plan:
- Solución enlatada:
- Refinado paso a paso:
- Métodos de composición de planes:
 - Concatenar plan:
 - Anidar plan:
 - Fusionar plan:

- Adaptar plan:

• Simulación:

• Reglas de buen estilo:

1.2. Análisis de la situación actual del objeto de investigación.

1.3. Formulación del problema

- Problemas General
- Problemas Derivados

1.4. Delimitación De La Investigación

- Delimitación espacial
- Delimitación temporal
- Unidades de Información

1.5. Justificación

1.6. Objetivos

1.6.1. Objetivo General

1.6.2. Objetivos Específicos

CAPITULO II

2. MARCO TEORICO CONCEPTUAL

2.1. Alternativas Teóricas Asumidas

- La Lógica de Programación:
- Lenguaje de Programación.
- La programación Lógica junto a la Funciona
- Metodología para la Construcción de Programas.
- Formulación y Análisis del Problema
- Especificación Funcional
- Entrada Proceso Salida
- Especificación de los Argumentos o Parámetros
- Establecimiento de Restricciones y Atributos

- **Diseño**
- **Algoritmo**
- **Tipos de Algoritmos**
- **Lenguajes Algorítmicos**
- **Tipos de Lenguajes Algorítmicos**
- **Metodología para la solución de problemas por medio de la computadora**
- **Definición del Problema**
- **Análisis del Problema**
- **Diseño del Algoritmo**
- **Entidades primitivas para el desarrollo de algoritmos**
- **Propiedades de los algoritmos estructurales:**
- **Herramientas para diseñar algoritmos:**
- **Programación del algoritmo**
- **Representación de algoritmos**
- **Pseudocódigo**
- **Técnicas para la formulación de algoritmos, diagramas de flujo**
- **Diagramas de Flujo**
- **Organigramas**
- **Tipos de Datos**
- **Datos Numéricos**
- **Datos Lógicos**
- **Datos Alfanuméricos (String)**
- **Identificadores**
- **Reglas para formar un Identificador**
- **Constantes, Variables y Expresiones**
- **Constantes**
- **Variable**
- **Clasificación de las Variables**
- **Por su contenido**
- **Por su Uso**

- **Explosiones**
 - **Operadores**
 - **Operadores Aritméticos**
 - **Operadores Aritméticos**
 - **Prioridad de los Operadores Aritméticos**
 - **Operadores Relacionales**
 - **Tipos de Operadores Relacionales**
 - **Operadores Lógicos**
 - **Tipos de operadores lógicos**
 - **Operador Not Operador Not**
 - **Operador And Operador And**
 - **Operador Or Operador Or**
 - **Prioridad de los Operadores**
 - **Prioridad de los Operadores en General**
 - **Técnicas de diseño**
 - **Top Down**
 - **Bottom UP**
 - **Codificación**
 - **Compilación y ejecución.**
 - **Verificación y depuración.**
 - **Tipos de errores**
 - **Documentación y mantenimiento.**
 - **Mantenimiento**
 - **Ejercicio para ilustrar la metodología**
- 2.2. Categoría del análisis conceptual**
- 2.3. Antecedentes investigativos**
- 2.4. Fundamentación teórica – científica**
- 2.5. Planteamiento de la hipótesis: general, específica**
- 2.5.1. Hipótesis General**
- 2.5.2. Hipótesis Específica**

2.6. Operacionalización de Variables

2.6.1. Variable Independiente

2.6.2. Variable Dependiente

2.7. Definición de Términos

CAPITULO III

3. DISEÑO METODOLOGICO DE LA INVESTIGACION

3.1. Metodología Empleada

3.1.1. Método Inductivo

3.1.2. Método deductivo

3.1.3. Método descriptivo

3.1.4. Modalidad de la investigación.

3.1.5. Nivel o tipo de investigación.

3.2. Población y Muestra

3.2.1. Población

3.2.2. Muestra

3.3. Técnicas e Instrumentos de Investigación

3.3.1. Encuesta

3.4. Fuentes de Obtención de la Información

3.4.1. Fuentes primarias

3.4.2. Fuentes secundarias

3.5. Procesamiento de la Información

CAPITULO IV

4. ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

4.1. Verificación de Hipótesis

4.2. Conclusiones y Recomendaciones

4.3. Conclusiones

4.4. Recomendaciones

CAPITULO V

5. MARCO ADMINISTRATIVO

5.1. Recursos

- Humanos
- Materiales
- Presupuesto

CAPITULO VI

6. PROPUESTA ALTERNATIVA

6.1. Tema

6.2. Justificación

6.3. Fundamentación

6.4. Contenido

- Estructuras repetitivas
- Estructura Desde – Hasta
- Estructura Mientras
- Estructura Repetir – Hasta _Que
- Operadores y operandos
- Operadores Aritméticos
- Operadores Relacionales
- Mayor que (>)
- Menor que (<)
- Mayor o igual que (>=)
- Menor o igual que (<=)

- Diferente (<>)
- Igual (=)
- Operadores Lógicos
- And (Y)
- Or (O)
- Not (Negación)
- ENCUENTRO N° 1
- ENCUENTRO N° 2
- ENCUENTRO N° 3
- ENCUENTRO N° 4

6.5. Objetivos

6.5.1. Objetivo General

6.5.2. Objetivos Específicos

6.6. Descripción de los aspectos operativos de la propuesta

- Determinación del Lugar para la Capacitación
- Análisis Creativo antes de las Actividades
- Relación del Trabajo Organizacional
- Actividades
- Actividades Individuales
- Análisis Crítico: después de las Actividades
- Métodos y Técnicas a Utilizarse

6.7. Recursos

- Humanos
- Materiales
- Técnicos (Funciones de los Capacitadores)
- Responsables
- Financiamiento
- Cronograma de Ejecución de la Propuesta

6.8. Bibliografía
- **Bibliografía de Textos**

Anexos

INTRODUCCIÓN

Los acelerados cambios, muchos de ellos irreversibles, que se están produciendo no sólo en los modos de conocer, acceder a la información, pensar, actuar, enseñar y aprender, sino también en la manera de ver a la mente humana y sus productos, que aparecen como el resultado de la implantación de las tecnologías, constituyen un objeto de reflexión y de preocupación por parte de los poderes públicos y de los demás sujetos comprometidos con los resultados del proceso educativo.

Las respuestas a estos desafíos no pueden hacerse esperar demasiado, ni ser de forma fragmentarias o parciales, por el contrario, se debe partir de una convicción profunda acerca de la necesidad de encarar con decisión un futuro incierto y abierto, elaborando para ello respuestas novedosas, que afirmen en lo posible lo mejor del sistema formativo tradicional, pero que al mismo tiempo señalen una clara línea de ruptura con el pasado y con algunos pobres resultados hasta ahora obtenidos en el área de la matemática.

En este sentido, la fusión de la informática y la pedagogía en el campo de la educación da paso a lo que se denomina Enseñanza Asistida por Computador. Esta fusión proporciona una enseñanza individualizada a través de herramientas didácticas que facilitan el avance educativo de los estudiantes, así como también, representen un apoyo para los docentes al momento de impartir sus clases.

Sobre la base de estos argumentos, se presenta este estudio que tuvo como finalidad contribuir con el proceso de enseñanza - aprendizaje, planteando las bases teóricas (pedagógicas e informáticas), que permitan alcanzar un conjunto de objetivos en el área de matemática, a través

del desarrollo de un Software Educativo para la enseñanza del contenido transformaciones en el plano.

1. CAMPO CONTEXTUAL DE LA PROBLEMÁTICA

1.1. Teoría de la Programación

El primer objetivo es, especificar un plan detallado y realizable (un algoritmo) que solucione el problema.

El segundo es, implementar este plan en un lenguaje de programación.

El tercero es, depurar el programa resultante; este proceso puede ser tan complejo como los anteriores.

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

Programar es una disciplina de diseño y su resultado no es un programa, sino una construcción que resuelve un problema. Soloway parte de que los expertos no son conscientes del conocimiento que utilizan para resolver problemas. La hipótesis Whorfiana plantea que el lenguaje ayuda al pensamiento. Siguiendo esto, Soloway crea un vocabulario que considera

necesario enseñar explícitamente, aunque sea conocimiento tácito de los expertos. Este vocabulario incluye los siguientes conceptos:

- **Mecanismo:** Especifica una cadena de acciones que, al ser activadas, producen un efecto deseado.
- **Explicación:** El rastro de información que especifica cómo y por qué fue diseñado un artefacto.
- **Objetivo:** Un resultado que se desea alcanzar; la solución final de un problema.
- **Plan:** Una forma de alcanzar un objetivo. Para realizarse, puede requerir alcanzar uno o más subobjetivos.
- **Solución enlatada:** Plan que se conoce a la perfección y puede ser aplicado para alcanzar objetivos en distintas situaciones.
- **Refinado paso a paso:** Macro estrategia de diseño que sugiere descomponer un problema en subproblemas, pero de tal forma que se puedan usar soluciones enlatadas para resolverlos.
- **Métodos de composición de planes:** Micro estrategia de diseño que sugiere distintos métodos para unir soluciones enlatadas, de forma que se solucione un problema complejo:
 - **Concatenar plan:** Dos planes se juntan en secuencia, uno después de otro.
 - **Anidar plan:** Un plan se rodea completamente por otro.
 - **Fusionar plan:** Al menos dos planes se mezclan. Puede ser extremadamente complejo.
 - **Adaptar plan:** Modificar un plan para adecuarlo a una situación específica.
- **Simulación:** Estrategia para encontrar las propiedades dinámicas de un plan.
- **Reglas de buen estilo:** Conjunto de reglas, generalmente implícitas, que determinan un estilo de programación más fácil de entender.

1.2. Análisis de la Situación Actual del Objeto de Investigación.

La programación en lógica se ha convertido en uno de los principales exponentes de la programación declarativa. Sin embargo, aunque se la ha utilizado como herramienta de representación de conocimiento, presenta limitaciones para adaptarse al razonamiento rebatible de los alumnos. Como los agentes inteligentes tienden a razonar en una forma rebatible, sería sumamente interesante disponer de un paradigma de programación donde conclusiones previas puedan ser refutadas ante la presencia de mayor información. De la misma forma que la programación en lógica significó una evolución de los lenguajes de programación, la utilización de la argumentación rebatible como paradigma, podría permitir un nuevo avance. En los últimos años se han desarrollado extensiones de la programación en lógica que incorporan algunos aspectos del razonamiento rebatible. Por ejemplo, Gelfond y Lifschitz en su trabajo *Logic Programming with Classical Negation* introdujeron los programas lógicos extendidos que permiten la representación de la negación clásica en los programas, y la derivación de literales negados. Presentan ventajas importantes para representar conocimiento, pero lamentablemente cuando dos literales complementarios (por ej. P y $\neg P$), el programa se torna "contradictorio", y entonces pueden derivarse todos los literales del programa. Un argumento para una conclusión constituye una pieza de razonamiento, tentativa que un agente inteligente está dispuesto a aceptar para explicar. Si el agente adquiriese luego nueva información, la conclusión junto con el razonamiento que la produjo podría quedar invalidada. En un sistema de argumentación rebatible, una conclusión será aceptada como una nueva creencia, cuando exista un argumento que sea una justificación. El proceso de obtención de una justificación, involucra la construcción de un argumento. Para verificar si un argumento está derrotado, se construyen contraargumentos que son posibles derrotadores. Como los derrotadores son argumentos, se debe verificar que no estén a su vez derrotados, y así siguiendo.

1.3. Formulación Del Problema

13.1. Problemas General

¿COMO INFLUYE LAS TECNICAS PARA EL DESARROLLO DE LA LÓGICA EN LA ELABORACIÓN DE PROGRAMAS INFORMÁTICOS, EN EL APRENDIZAJE DE LOS ALUMNOS DEL PRIMER AÑO PARALELO “A” DE BACHILLERATO, DEL COLEGIO “NICOLAS INFANTE DIAZ”, DEL CANTON QUEVEDO PROVINCIA DE LOS RIOS AÑO 2.012?

1.3.2. Problemas Derivados

- ¿De qué manera la lógica en la elaboración de programas informáticos influye en el desarrollo de las tareas escolares de los estudiantes del primer año bachillerato?.
- ¿Por qué la lógica en la elaboración de programas informáticos influye en la participación en las actividades prácticas en los estudiantes del primer año bachillerato?

1.4. DELIMITACIÓN DE LA INVESTIGACIÓN

Delimitación espacial.- La investigación se la realiza en el colegio fiscal mixto “Nicolás Infante Díaz” de la parroquia San Camilo del cantón Quevedo provincia Los Ríos.

Delimitación temporal.- En el periodo 2012

Unidades de Información.- Recogiendo la información de padres de familia, docentes y autoridades de la institución

1.5. JUSTIFICACION

Con la llegada de la informática, muchos cambios se han producido en los diversos sectores del país, quizás el más relevante ha sido en el campo educativo, debido a que la informática, según Pérez (2000) tiene el potencial de contribuir al mejoramiento de la calidad de la docencia, el aprendizaje y la enseñanza, especialmente en esta época caracterizada por la revolución del conocimiento y la información.

Si se toma en cuenta que la sociedad actual está invadida por computadoras, producto de las más sofisticadas tecnologías y se consideran las enormes posibilidades en el desarrollo de las capacidades humanas, así como los efectos que a nivel de los individuos tiene y tendrá la informática y sus aplicaciones en el proceso de aprendizaje, se puede concluir que su aplicación es necesaria incorporarla en el sistema educativo, con la finalidad de iniciar la formación de una mente analítica y se desarrolle en el estudiante un espíritu reflexivo y crítico (Galindo, 2000).

Por otro lado el rendimiento del sistema educativo en la última década y los vertiginosos adelantos que se registran en el campo de la ciencia y la tecnología descartan las posibilidades de seguir empleando con exclusividad los métodos tradicionales de enseñanza. Por el contrario, se hace evidente la urgente necesidad de introducir innovaciones metodológicas, técnicas, empleo de medios y recursos para llegar con mayor eficiencia y eficacia a toda la población estudiantil (Castro, 2000).

Por estas razones, existe la necesidad de crear nuevos recursos y herramientas que faciliten la eficiente administración del proceso de enseñanza – aprendizaje específicamente en el área de la Informática.

Bajo estas perspectivas, la presente investigación plantea la utilización de las tecnologías de la información y comunicación en educación, como una herramienta novedosa de enseñanza-aprendizaje, que permita introducir en el aula de clases los desarrollos tecnológicos como estrategias, que conllevan a mejorar el bajo rendimiento que se está obteniendo en Informática.

Este estudio proporcionará un recurso interactivo para la enseñanza del contenido transformaciones en el Plano, mejorando la enseñanza de este tópico, en el cual los alumnos presentan dificultades, esto se logra a través de la incorporación de la multimedia e hipermedia, elementos que permiten captar visual y auditivamente la información.

Cabe destacar que esta herramienta garantizará en los alumnos una enseñanza sólida, un mejor manejo de la geometría y control de su aprendizaje, por otra parte le permitirá al docente desarrollar al máximo sus roles de facilitador, mediador y orientador del proceso de enseñanza – aprendizaje, impartiendo una enseñanza más individualizada y sirviendo como modelador de las cualidades intelectuales del estudiante del colegio fiscal mixto “Nicolás Infante Díaz”, deben ser lo suficientemente reflexivos para alcanzar el éxito.

1.6. OBJETIVOS

1.6.1. Objetivo General

Investigar como influyen las técnicas para el desarrollo de la lógica en la elaboración de programas informáticos, en el aprendizaje de los alumnos del

primer año paralelo “a” de bachillerato, del colegio “Nicolás Infante Díaz”, del Cantón Quevedo provincia de los Ríos año 2.012?

1.6.2. Objetivos Específicos

- Aplicando las herramientas metodológicas de programas informáticos al desarrollo de la enseñanza – aprendizaje.
- Utilizando las diferentes estrategias para resolver los problemas informáticos que se plantean dentro del aula de clases para el análisis de información sistemática.

2. MARCO TEORICO CONCEPTUAL

2.1. Alternativas Teóricas Asumidas

Estudiar cada uno de los conceptos y las herramientas que se utilizan en la elaboración de programas informáticos o sistemas computarizados.

La Lógica de Programación: La lógica de programación son los pasos en secuencia que se debe seguir al momento de realizar un análisis del problema planteado.

Lenguaje de Programación: Es un Software de aplicación que permite al usuario crear programas mediante instrucciones y comandos. Los lenguajes de programación. Constituyen el *software* empleado para desarrollar los sistemas operativos o las aplicaciones de carácter general. En resumen, el conjunto de órdenes lógicas que permiten el funcionamiento del ordenador es el *software* del mismo.

La programación Lógica junto a la Funcional: La programación lógica junto con la funcional, forman parte de lo que conocemos como ***programación declarativa***. En los lenguajes tradicionales, la programación consiste en indicar como resolver un problema mediante sentencias: en la programación lógica, se trabaja de una forma descriptiva, estableciendo relaciones entre entidades, indicando no cómo, sino que hacer. La ecuación Robert Kowalski (Universidad de Edimburgo) establece la idea esencial de la programación lógica: ***algoritmo = lógica + control***. Es decir, un algoritmo se construye especificando conocimientos en un lenguaje formal (lógica de primer orden), y el problema se resuelve mediante un mecanismo de inferencia (control) que actúa sobre aquel.

La programación lógica constituye una herramienta que tradicionalmente ha provisto de una sólida estructura conceptual para representar conocimiento. Puede ser usada para describir y recuperar conocimientos sobre un dominio de aplicación, para describir nuestro conocimiento sobre tareas de cálculo tradicional y para describir tareas menos triviales como la capacidad de encontrar solución a problemas que normalmente se considera dentro del campo de la inteligencia artificial. Esto es porque encontrar soluciones a problemas es todavía una capacidad muy dependiente de la intuición humana, las cuales son muchas veces difíciles de desarrollar y formalizar. No hay una fórmula general para un buen uso de la programación lógica en el desarrollo de aplicaciones de este tipo, pero se intentará ilustrar en este trabajo una aproximación para abordar esta tarea. En esta aproximación se plantea un enfoque que parte desde una descripción del dominio de aplicación en términos de entidades y relaciones satisfechas por estas entidades, que a su vez permiten representar al mundo en términos de estados. Veamos unos ejemplos de paradigmas.

Paradigma de Programación: *Alonso y Segovia definen un paradigma de programación como un modelo básico de diseño e implementación de programas, Ambler, Brunett y Zimmerman lo definen como “una colección de modelos conceptuales que juntos modelan el proceso de diseño y determinan la estructura de un programa. Ambler propone que el tipo de solución que aporta un paradigma para resolver un problema particular, define tres categorías de paradigmas. Solución procedimental u Operacional, Solución Demostrativa y Solución Declarativa.*

Paradigma de Programación Lógica: Genesereth y Ginsberg señalaron que *“la programación lógica es una programación por descripción. El programa se construye describiendo el área de aplicación, esto es, se señala el que se desea (mediante hechos que son verdaderos) pero el como obtenerlo”. Asimismo, Ambler señaló que “el paradigma lógico asume la definición de un conjunto de hechos....y un conjunto de reglas que permitan la deducción de otros hechos. Así la programación lógica, desde la perspectiva del*

programador, es una técnica que consiste en expresar apropiadamente todos los hechos y reglas necesarias que definen el problema”.

Programa Lógico: un programa lógico se configura como un conjunto de hechos (asertos ó proposiciones), y de reglas lógicas previamente establecidas, que obtienen conclusiones en base a una serie de preguntas lógicas. El control es inherente al sistema, el que permite investigar las preguntas lógicas. Esta capacidad es el concepto clave que subyace en la Programación Lógica (descriptiva). Al separar el control y a la lógica, el programa lógico se transforma en un conjunto de declaraciones formales de especificaciones que deben ser correctas por definición.

Sistema de Programación Lógica: El conjunto de hechos y reglas constituyen lo que se denomina una descripción, que llega a ser un programa donde combinado con un procedimiento de inferencia independiente de la aplicación hace posible a la computadora sacar conclusiones acerca del área de aplicación y responder preguntas aún cuando estas respuestas no estén explícitamente registradas en la descripción. Esta capacidad es la base de la tecnología de la programación lógica.

Metodología para la Construcción de Programas: De una forma general les presentamos los principales pasos que se deben seguir para resolver problemas aplicando técnicas de programación. Esta metodología será desarrollada a lo largo de todo el proceso de enseñanza académica, en la medida que se estudien unidades del mismo.

Formulación y Análisis del Problema: Consiste en entender de qué se trata el problema planteado y esbozar su posible solución, concluyendo con una clara definición de tres aspectos:

- a) Qué es lo que nos piden, es decir, definición del resultado o solución deseada (para qué).
- b) Cómo obtener lo que nos piden (qué hacer).
- c) Qué necesitamos para obtener los resultados pedidos (con qué). Esto último nos facilitará la construcción de lo que denominaremos Especificación Funcional.

Especificación Funcional: Consiste en determinar las funciones que se van a realizar (qué hacer) y sus respectivas entradas (con qué) y salidas (para qué).

Entrada Proceso Salida: Dónde: Entrada son los argumentos (variables o constantes) que se requieren para resolver un problema, salida son los resultados (argumentos) que se desean obtener una vez resuelto el problema y proceso es el procedimiento(s) u operación(es) que deben efectuarse sobre las entradas para obtener las salidas deseadas.

Especificación de los Argumentos o Parámetros: Consiste en la documentación de los argumentos o parámetros (sean estos de entrada, salida o intermedios) requeridos en la solución del problema, mediante la elaboración de una tabla que contemple los siguientes aspectos: descripción del argumento, identificador, tipo longitud, condición restricción.

Establecimiento de Restricciones y Atributos: Consiste en determinar bajo qué restricciones se ha de operar y cuáles son las medidas de rendimiento y calidad que debe tener el sistema (programa). Este aspecto no se realizará, dada la simplicidad de los problemas tratados.

Diseño: Consiste en diseñar cómo hace el programa la tarea solicitada.. En forma general consiste en dividir el programa en subprogramas y cada subprograma en módulos. Este tópico se tratará con mayor detalle. Inicialmente no dividiremos el programa en subprogramas. El criterio de descomposición más utilizado es el de tipo funcional, el cual produce una estructura jerárquica en la que cada módulo ejecuta una o más funciones y para cada módulo se produce una especificación de programa o módulo, la cual contiene lo siguiente:

- Nombre del Programa o Módulo.
- Función que desarrolla.
- Parámetros o Argumentos.
- Parámetros o Argumentos de Entrada.
- Parámetros o Argumentos de Salida.
- Estructura de Datos Requerida.

- Lenguaje de Programación.
- Estructura de datos no se utilizará.

Algoritmo: La palabra algoritmo se deriva de la traducción al latín de la palabra árabe alkhwarizmi, nombre de un matemático y astrónomo árabe que escribió un tratado sobre manipulación de números y ecuaciones en el siglo IX. Un algoritmo es una serie de pasos organizados que describe el proceso que se debe seguir, para dar solución a un problema específico. También se puede decir que ALGORITMO es un conjunto finito de pasos en secuencia que indican cómo se resuelve un determinado problema.

Tipos de Algoritmos: Existen dos tipos y son llamados así por su naturaleza:

- **Cualitativos:** Son aquellos en los que se describen los pasos utilizando palabras.
- **Cuantitativos:** Son aquellos en los que se utilizan cálculos numéricos para definir los pasos del proceso.

Lenguajes Algorítmicos: Un Lenguaje algorítmico es una serie de símbolos y reglas que se utilizan para describir de manera explícita un proceso.

Tipos de Lenguajes Algorítmicos:

Gráficos: Es la representación gráfica de las operaciones que realiza un algoritmo (diagrama de flujo).

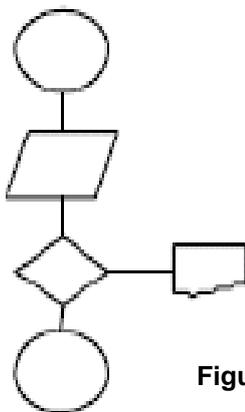


Figura 1: Tipos de Lenguajes

No Gráficos: Representa en forma descriptiva las operaciones que debe realizar un algoritmo (pseudocódigo).

INICIO

Edad: Entero

ESCRIBA “cuál es tu edad?”

Lea Edad

SI Edad \geq 18 entonces

ESCRIBA “Eres mayor de Edad”

FIN SI

ESCRIBA “fin del algoritmo”

FIN

Metodología Para La Solución De Problemas Por Medio De La Computadora

Definición del Problema: Esta fase está dada por el enunciado del problema, el cual requiere una definición clara y precisa. Es importante que se conozca lo que se desea que realice la computadora; mientras esto no se conozca del todo no tiene mucho caso continuar con la siguiente etapa.

Análisis del Problema: Una vez que se ha comprendido lo que se desea de la computadora es necesario definir:

Los datos de entrada

Cuál es la información que se desea producir (salida)

Los métodos y fórmulas que se necesitan para procesar los datos.

Una recomendación muy práctica es el de colocarse en el lugar de la computadora y analizar qué es lo que se necesita que se ordene y en qué secuencia para producir los resultados esperados.

Diseño del Algoritmo: Las características de un buen algoritmo son:

- Debe tener un punto particular de inicio
- Debe ser definido, no debe permitir dobles interpretaciones
- Debe ser general, es decir, soportar la mayoría de las variantes que se puedan presentar en la definición del problema

- Debe ser finito en tamaño y tiempo de ejecución
- Diseño del algoritmo
- Prueba de escritorio o Depuración.

Se denomina prueba de escritorio a la comprobación que se hace de un algoritmo para saber si está bien hecho. Esta prueba consiste en tomar datos específicos como entrada y seguir la secuencia indicada en el algoritmo hasta obtener un resultado, el análisis de estos resultados indicará si el algoritmo está correcto o si por el contrario hay necesidad de corregirlo o hacerle ajustes.

Entidades primitivas para el desarrollo de algoritmos: Todos estos elementos con los cuales se construyen dichos algoritmos se basan en una disciplina llamada: Programación Estructurada.

Empecemos por conocer las reglas para cambiar fórmulas matemáticas expresiones válidas para la computadora, además de diferenciar constantes e identificadores y tipos de datos simples.

Propiedades de los algoritmos estructurales:

- a) Número finito de pasos sin ambigüedades.
- b) Numeración de cada paso en orden secuencial
- c) La acción a realizar se indica con un verbo o con un gráfico.
- d) Condicionado a las estructuras básicas de la programación estructurada.
- e) Eficiente (menor número de pasos pero lo más claro posible)

Herramientas para diseñar algoritmos:

- a) Diagramas de Flujo: representación gráfica de un algoritmo.
- b) Pseudocódigo: lenguaje de especificación de algoritmos (el algoritmo se representa mediante palabras similares al inglés o al español, para facilitar tanto la lectura como la escritura de programas. Los símbolos o la notación para elaborar, respectivamente, los diagramas de flujo o los

pseudocódigos, serán indicados en los siguientes apuntes, en la medida que se estudie las correspondientes estructuras de programación.

Programación del algoritmo: Una vez que el algoritmo está diseñado y representado, se debe pasar a la fase de resolución práctica del problema con el ordenador. Esa fase se descompone a su vez en las siguientes subfases.

1. Codificación del Algoritmo en un programa
2. Ejecución del programa
3. Comprobación del programa



Figura 2: Programación del algoritmo

La fase de conversión de un algoritmo en instrucción de un lenguaje de programación, como sabemos se denomina codificación. El código deberá estar escrito de acuerdo con la sintaxis del lenguaje de programación ya que solamente las instrucciones sintácticamente correctas pueden ser interpretadas por el computador.

Nótese que durante el proceso de programación, se debe separar el diseño del algoritmo de su posterior implementación en un lenguaje de programación específico. Por ello se distingue entre el concepto más general

de programación y el más particular de codificación, que depende del lenguaje de programación utilizado. Tras la codificación del programa, éste deberá ejecutarse en un computador. El resultado de esta primera ejecución es incierto, ya que existe una alta posibilidad de que aparezcan errores bien en la codificación o en el propio algoritmo. Por tanto, el paso siguiente consiste en comprobar el correcto funcionamiento del programa y en asegurarse.

Representación de algoritmos: Un algoritmo es algo puramente conceptual que necesita una forma de representación, bien para comunicarlo a otra persona bien para ayudar a convertirlo en un programa. De hecho, la codificación en un lenguaje de programación, es una representación muy utilizada de un algoritmo, sin embargo tiene el inconveniente de que no todas las personas conocen el lenguaje que se haya elegido. Por ello, existen diferentes métodos que permiten que se pueda independizar el algoritmo de su correspondiente codificación.

Pseudocódigo: El pseudocódigo es un lenguaje de especificación de algoritmos (no de programación) basado en un sistema notacional, con estructuras sintácticas y semánticas, similares a los lenguajes procedurales, aunque menos formales que las de éstos, por lo que no puede ser ejecutado directamente por un computador. El pseudocódigo se utiliza para representar las sucesivas acciones, palabras reservadas similares a sus homónimas en los lenguajes de programación, tales como *start*, *end*, *stop*, *if-then-else*, *repeat-until* (*inicio*, *fin*, *parar*, *si-entonces-sino*, *mientras-hacer*, *repetir-hasta*), etc. La ventaja del uso del pseudocódigo reside en:

Su uso en la planificación de un programa; permitiendo que el programador se pueda concentrar en la lógica y en las estructuras de control y no tenga que preocuparse, por ahora de detalles acerca de las reglas sintácticas y semánticas de un lenguaje específico. Consiguientemente es más fácil de modificar, en el caso de que descubran errores o anomalías en la lógica del algoritmo.

Aunque el pseudocódigo es independiente del lenguaje de alto nivel que vaya a utilizarse, un algoritmo expresado en pseudocódigo puede ser traducido más fácilmente a mucho de ellos

Ejemplo

Supongamos que tenemos un algoritmo para averiguar si un número es par, que puede ser descrito narrativamente de la siguiente forma. “Si restado consecutivamente doses del número se obtiene el número 2, es par, si se obtiene otro valor (el 1), entonces es impar”. Este algoritmo escrito en pseudocódigo sería.

Leer N

Mientras N>2 hacer

N←N-2

Si N=2 entonces

Escribe “es par”

Sino

Escribe “es impar”

Fin

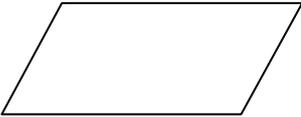
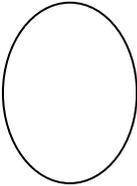
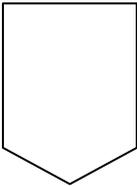
Nótese que en este ejemplo se han utilizado dos estructuras que son muy utilizadas en la programación: Mientras-hacer y si-entonces-si_no; y que la escritura del pseudocódigo usa normalmente la identificación (sangría en el margen izquierdo) de diferentes líneas para ayudar a delimitar visualmente cada una de las estructuras utilizadas.

Técnicas para la formulación de algoritmos. Diagramas de flujo

Diagramas de Flujo: Un diagrama de Flujo es la representación gráfica de un algoritmo. También se puede decir que es la representación detallada en

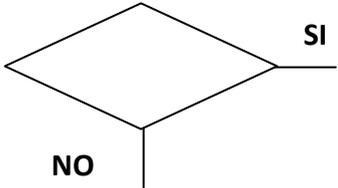
forma gráfica de cómo deben realizarse los pasos en la computadora para producir resultados.

Esta representación gráfica se da cuando varios símbolos (que indican diferentes procesos en la computadora), se relacionan entre si mediante líneas que indican el orden en que se deben ejecutar los procesos. Los símbolos utilizados han sido normalizados por el instituto norteamericano de normalización (ANSI):

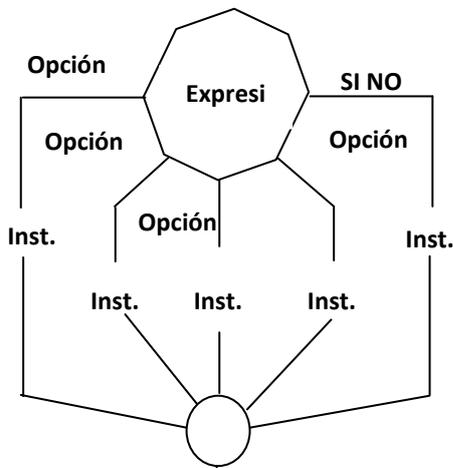
SIMBOLOS	DESCRIPCION
	Indica el inicio y el final de nuestro diagrama de flujo.
	Indica la entrada y salida de datos
	Símbolo de proceso y nos indica la asignación de un valor en la memoria y/o la ejecución de una operación aritmética
	Indica la salida de información por Impresora
	Conector dentro de página. Representa la continuidad del diagrama dentro de la misma página
	Conector fuera de página. Representa la continuidad del diagrama en otra página



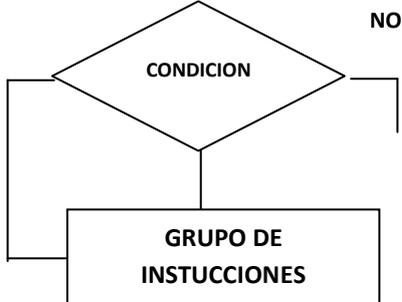
Indica la salida de información en la pantalla o monitor



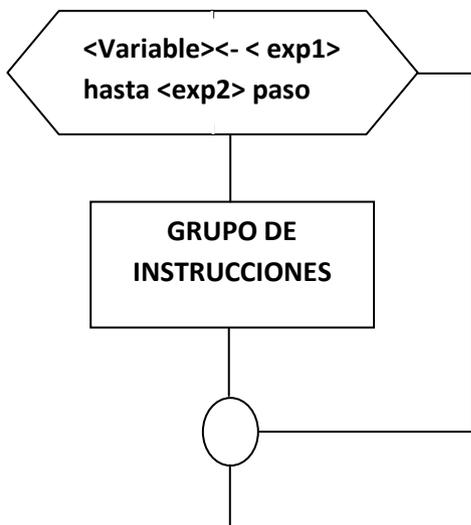
Símbolo de decisión. Indica la realización de una comparación de valores



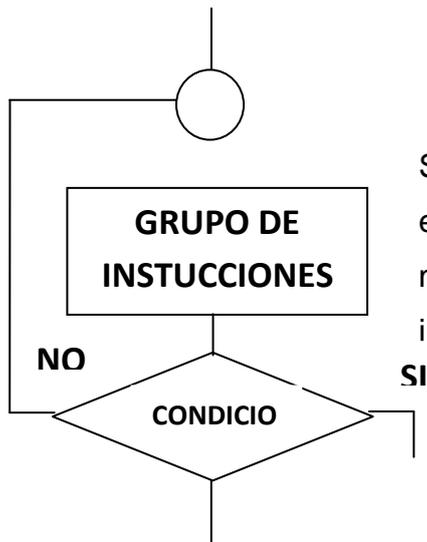
Símbolo de selección. Dada una expresión permite escoger una opción de muchas



Símbolo del mientras. Dada una expresión al principio de la interacción esta es evaluada; si la condición es verdadera realizará el ciclo, si es falsa la repetición cesará.



Símbolo del (para). Esta estructura de control repetitiva se usa generalmente cuando se conoce de antemano el número de iteraciones.



Símbolo Repita Hasta, funciona igual que la estructura Mientras, con la diferencia que al menos una vez hará un grupo de instrucciones y luego evaluará una condición

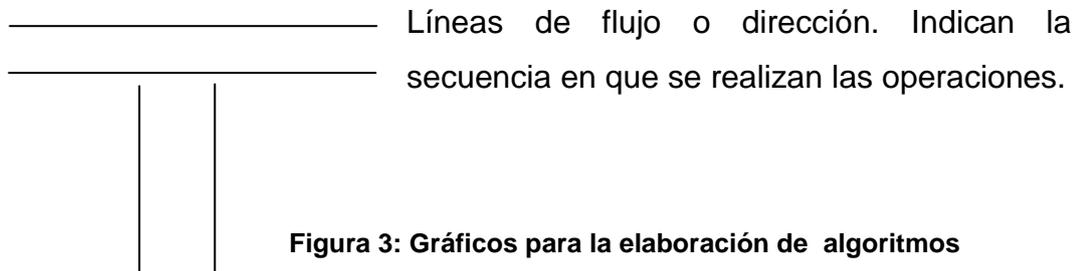


Figura 3: Gráficos para la elaboración de algoritmos

Organigramas: Para ganar claridad expositiva se han desarrollado una serie de símbolos gráficos que permiten presentar los algoritmos que son reconocidos universalmente. Veamos algunos ejemplos.

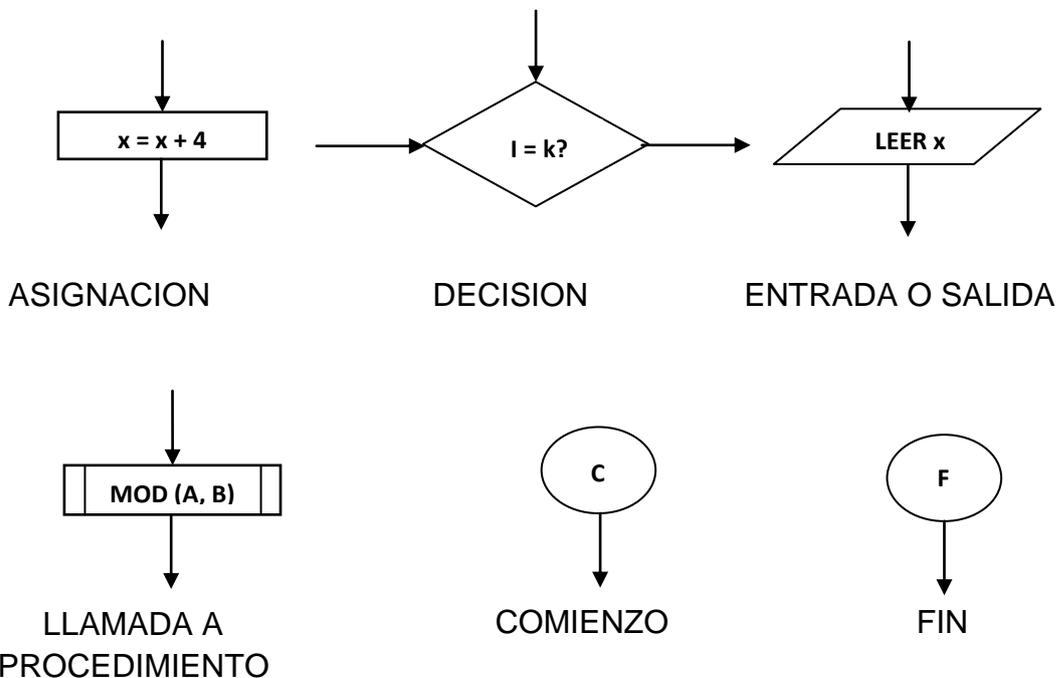


Figura 4: Gráficos que presentan a los algoritmos

Los organigramas o diagramas de flujo son herramientas gráficas utilizadas tanto para representar algoritmos, como en la ayuda en el diseño de programas. Están compuestos por una serie de símbolos, unidos con flechas, donde cada símbolo representa una acción distinta y las flechas el orden de realización de las acciones. Cada símbolo, por tanto, tendrá al menos una flecha que conduzca él y una flecha que parte de él, exceptuando el comienzo y final del algoritmo en la figura se muestran los símbolos utilizados habitualmente en la confección de organigramas, cuyo significado se completará más adelante.

En la siguiente figura se representa, en forma de organigrama, el algoritmo del ejemplo anterior ha sido expresado en pseudocódigo en la sección anterior.

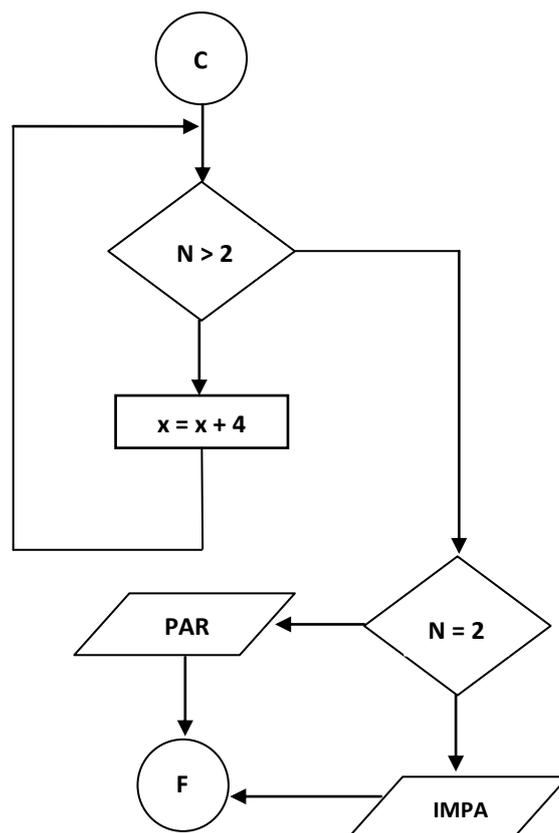
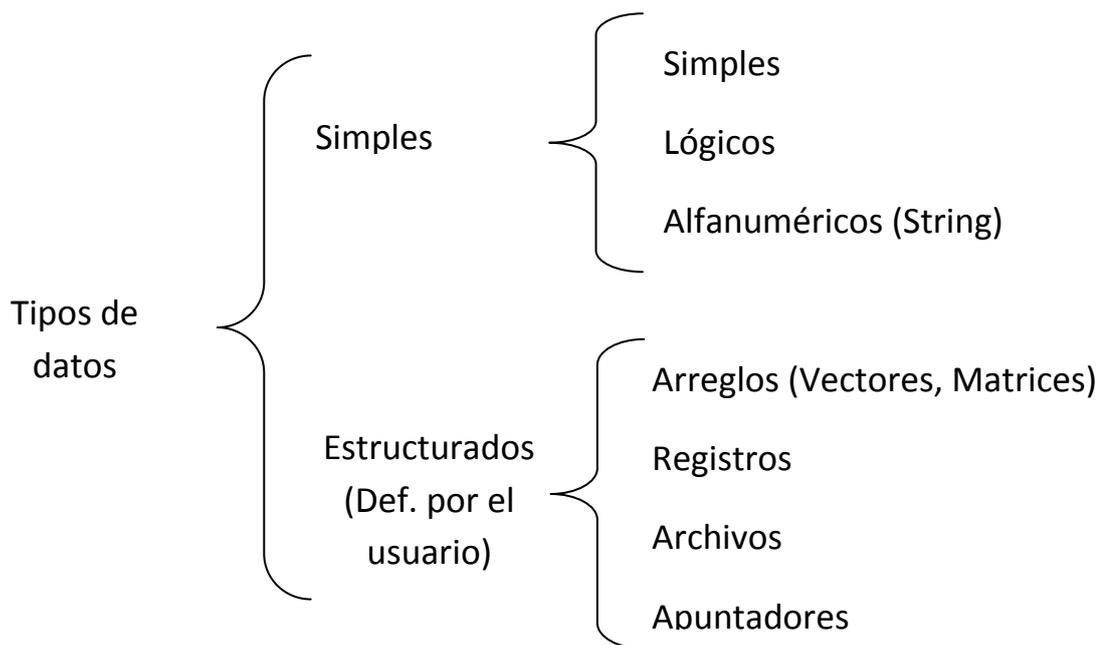


Figura 5: Organigrama del ejercicio

Codificación: Es la escritura en un lenguaje de programación de la representación del algoritmo desarrollado en la etapa de diseño. El resultado de la codificación es un programa fuente.

Tipos de Datos: Todos los datos tienen un tipo asociado con ellos. Un dato puede ser un simple carácter, tal como 'b', un valor entero tal 35. El tipo de dato determina la naturaleza del conjunto de valores que puede tomar una variable.



Datos Numéricos: Permiten representar valores escalares de forma numérica, esto incluye a los número enteros y a los reales. Este tipo de datos permiten realizar operaciones aritméticas comunes.

Datos Lógicos: Son aquellos que sólo pueden tener dos valores (cierto o falso) y que representan el resultado de una comparación entre otros datos (numéricos o alfanuméricos).

Datos Alfanuméricos (String): Es una secuencia de caracteres alfanuméricos que permiten representar valores identificables de forma descriptiva, esto incluye nombres de persona, direcciones, etc. Es posible representar números como alfanuméricos, pero estos pierden su propiedad

matemática, es decir no es posible hacer operaciones con ellos. Este tipo de datos se representan encerrados entre comillas.

Identificadores Los identificadores representan los datos de un programa (constantes, variables, tipos de datos). Un identificador es una secuencia de caracteres que sirve para identificar una posición en la memoria de la computadora, que permite acceder a su contenido.

Ejemplo:

- Nombre
- Num_hrs
- Calif2

Reglas para formar un Identificador

- Debe comenzar con una letra (A a Z, mayúsculas o minúsculas) y no deben contener espacios en blanco.
- Letras, dígitos y caracteres como la subraya (_) están permitidos después del primer carácter.
- La longitud de identificadores puede ser de varios caracteres. Pero es recomendable una longitud promedio de 8 caracteres.
- El nombre del identificador debe dar una idea del valor que contiene.

Constantes, Variables y Expresiones

Constantes: Una constante es un dato numérico o alfanumérico que no cambia durante la ejecución del programa.

Ejemplo:

pi = 3.1416

Variable: Es un espacio en la memoria de la computadora que permite almacenar temporalmente un dato durante la ejecución de un proceso, su contenido puede cambiar durante la ejecución del programa.

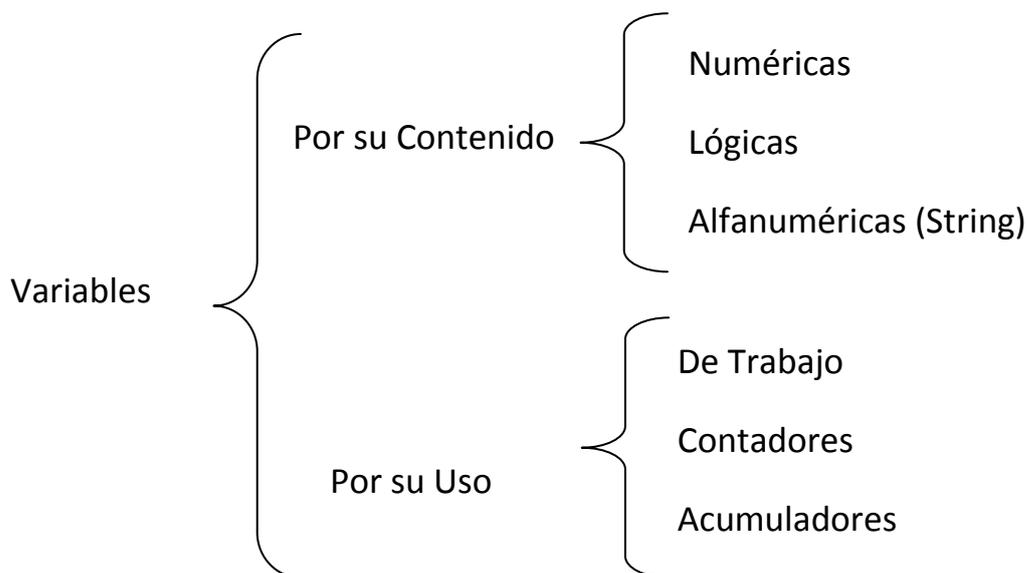
Para poder reconocer una variable en la memoria de la computadora, es necesario darle un nombre con el cual podamos identificarla de un algoritmo.

Ejemplo:

area = pi * radio

Las variables son: el radio, el área y la constante es pi.

Clasificación de las Variables



Por su contenido

Variables Numéricas: Son aquellas en las cuales se almacenan valores numéricos, positivos o negativos, es decir almacenan número del 0 al 9, signos (+ y -) y el punto decimal.

Ejemplo:

Iva = 0.15, pi = 3.1416, costo = 2500

Variables Lógicas: Son aquellas que solo pueden tener dos valores (cierto o falso) estos representan el resultado de una comparación entre otros datos.

Variabes Alfanuméricas: Está formada por caracteres alfanuméricos (letras, número y caracteres especiales).

Ejemplo:

Letra = 'a', apellido = 'López', dirección = 'Av. Guayaquil # 190'

Por su Uso

Variabes de Trabajo: Variables que reciben el resultado de una operación matemática completa y que se usan normalmente dentro de un programa.

Ejemplo:

Suma = $a + b / c$

Contadores: Se utilizan para llevar el control del número de ocasiones en que se realiza una operación o se cumple una condición. Con los incrementos generalmente de uno en uno.

Acumuladores: Forma que toma una variable y que sirve para llevar la suma acumulativa de una serie de valores que se van leyendo o calculando progresivamente.

Explosiones: Las explosiones son combinaciones de constantes, variables, símbolos de operación, paréntesis y nombres de funciones especiales.

Por ejemplo:

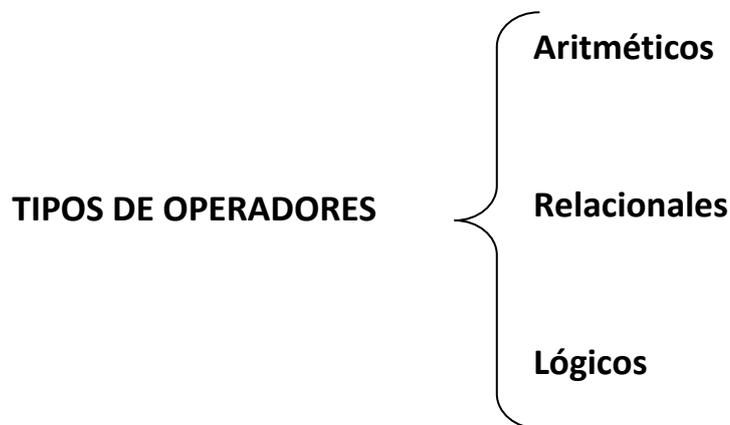
$a + (b + 3) / c$

Cada expresión toma un valor que se determina tomando los valores de las variables y constantes implicadas y la ejecución de las operaciones indicadas.

Una expresión consta de operadores y operandos. Según sea el tipo de datos que manipulan, se clasifican las siguientes expresiones en:

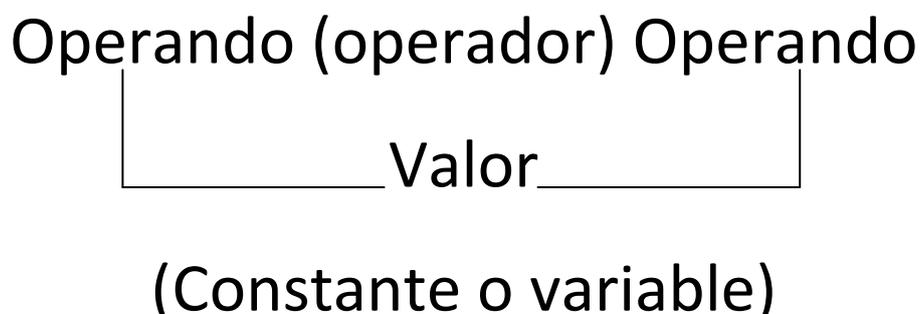
- Aritmética
- Relacionales
- Lógicas

Operadores: Son elementos que se relacionan de forma diferente, los valores de una o más variables y/o constantes. Es decir, los operadores nos permiten manipular valores.



Operadores Aritméticos: Los operadores aritméticos permiten la realización de operaciones matemáticas con los valores (variables y constantes).

Los operadores pueden ser utilizados con tipos de datos enteros o reales. Si ambos son enteros, el resultado es entero; si alguno de ellos es real, el resultado es real.



Operadores Aritméticos

+ Suma

- Resta

* Multiplicación

/ División

Mod Modulo (residuo de la división entera)

Ejemplos:

Expresión	Resultado
$7 / 2$	3.5
$12 \text{ mod } 7$	5
$4 + 2 * 5$	14

Prioridad de los Operadores Aritméticos: Todas las expresiones entre paréntesis se evalúan primero. Las expresiones con paréntesis anidados se evalúan de dentro a fuera, el paréntesis más interno se evalúa primero. Dentro de una misma expresión los operadores se evalúan en el siguiente orden

1. ^ Exponenciación.
2. *, /, mod, Multiplicación, División, Modulo.
3. +, - Suma y Resta

Los operadores en una misma expresión con igual nivel de prioridad se evalúan de izquierda a derecha.

Ejemplos:

$$4 + 2 * 5 = 14$$

$$3 + 5 * (10 - (2 + 4)) = 23$$

$$23 * 2 / 5 = 9.2$$

$$2.1 * (1.5 + 12.3) = 2.1 * 13.8 = 28.98$$

Operadores Relacionales: Se utilizan para establecer una relación entre dos valores. Luego compara estos valores entre si y esta comparación produce un resultado de certeza o falsedad (verdadero ó falso).

Los relacionales comparan valores del mismo tipo (numéricos o Cadenas). Estos tienen el mismo nivel de prioridad en su evaluación.

Los operadores relacionales tienen menor prioridad que los aritméticos.

Tipos de Operadores Relacionales

> Mayor que

< Menor que

>= Mayor o igual que

<= Menor o igual que

<> Diferente

= Igual

Ejemplos:

Ejemplos lógicos

Si $a = 10$, $b = 20$, $c = 30$

$a + b > c$ Falso

$a - b < c$ Verdadero

$a - b = c$ Falso

$a * b <> c$ Verdadero

Ejemplos no Lógicos:

$a < b < c$

$10 < 20 < 30$

$t > 5 < 30$

(No es lógico porque tiene diferentes operadores)

Operadores Lógicos: Estos Operadores se utilizan para establecer relaciones entre valores lógicos. Estos valores pueden ser resultado de una expresión relacional.

Tipos de operadores lógicos

And (Y)

Or (O)

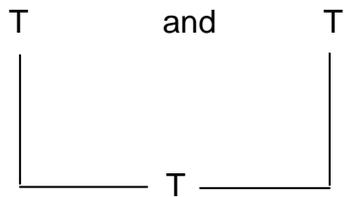
Not (Negación)

Ejemplos:

Para los siguientes Ejemplos T significa verdadero y F falso.

$(a < b) \text{ and } (b < c)$

$(10 < 20) \text{ and } (20 < 30)$



Operador Not

Operando	Resultado
T	F
F	T

Operador And

Operando 1	Operador	Operando 2	Resultado
T	AND	T	T
T		F	F
F		T	F
F		F	F

Operador Or

Operador 1	Operador	Operador 2	Resultado
T	OR	T	T
T		F	T
F		T	T
F		F	F

Prioridad de los Operadores

1. Not
2. And
3. Or

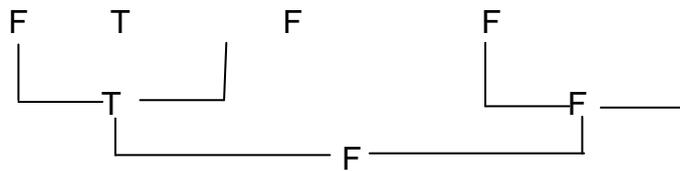
Prioridad de los Operadores en General

1. ()
2. ^
3. *, /, Mod, Not
4. +, -, And
5. >, <, >=, <=, <>, =, Or

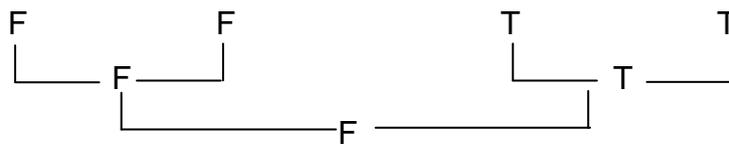
Ejemplos:

Sea: $a = 10$, $b = 12$, $c = 13$, $d = 10$

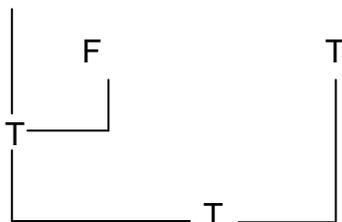
- 1) $((a > b) \text{ or } (a < c)) \text{ and } ((a = c) \text{ or } (a >= b))$



- 2) $((a >= b) \text{ or } (a < d)) \text{ and } ((a >= d) \text{ and } (a > d))$



- 3) $\text{not } (a = c) \text{ and } (c > b)$



Técnicas de diseño

Top Down: También conocida como de arriba – abajo y consiste en establecer una serie de niveles de mayor a menor complejidad (arriba – abajo) que den solución al problema. Consiste en efectuar una relación entre las etapas de la estructuración de forma que una etapa jerárquica y su inmediato inferior se relacionan mediante entradas y salidas de información. Este diseño es una serie de descomposiciones sucesivas del problema inicial, que recibe el refinamiento progresivo del repertorio de instrucciones que van a formar parte del programa.

La utilización de la técnica de diseño Top – Down tiene los siguientes objetivos básicos.

- Simplificación del problema y de los subprogramas de cada descomposición.
- Las diferentes partes del problema pueden ser programadas de modo independiente e incluso por diferentes personas.
- El programa final queda estructurado en forma de bloqueo o módulos lo que hace más sencilla su lectura y mantenimiento.

Bottom UP: El diseño ascendente se refiere a la identificación de aquellos procesos que necesitan computarizarse con forme vayan apareciendo, su análisis como sistema y su codificación, o bien, la adquisición de paquetes de software para satisfacer el problema inmediato.

Cuando la programación se realiza internamente y haciendo un enfoque ascendente, es difícil llegar a integrar los subsistemas al grado tal de que el desempeño global, sea fluido. Los problemas de integración entre los subsistemas son sumamente costosos y muchos de ellos no se solucionan hasta que la programación alcanza la fecha límite para la integración total del sistema. En esta fecha, ya se cuenta con muy poco tiempo, presupuesto o paciencia de los usuarios, como para corregir aquellas delicadas interfaces, que en un principio, se ignoran. Aunque cada subsistema parece ofrecer lo que

se requiere, cuando se contempla al sistema como entidad global, adolece de ciertas limitaciones por haber tomado un enfoque ascendente.

Uno de ellos es la duplicación de esfuerzos para acceder el software y más aún al introducir los datos.

Otro es, que se introducen al sistema muchos datos carentes de valor.

Un tercero y tal vez el más serio inconveniente del enfoque ascendente, es que los objetivos globales de la organización no fueron considerados y en consecuencia no se satisfacen.

Entonces: La diferencia entre estas dos técnicas de programación se fundamenta en el resultado que representan frente a un problema dado.

Imagine una empresa, la cual se compone de varios departamentos (contabilidad, mercadeo...), en cada uno de ellos se fueron presentando problemas a los cuales se le dieron una solución basados en un enfoque ascendente (Bottom Up): creando programas que satisfacían sólo el problema que se presentaba.

Cuando la empresa decidió integrar un sistema global para suplir todas las necesidades de todos los departamentos se dio cuenta que cada una de las soluciones presentadas no era compatible la una con la otra, no representaba una globalidad, característica principal de los sistemas.

Como no hubo un previo análisis, diseño de una solución a nivel global en todos sus departamentos, centralización de información, que son características propias de un diseño Descendente (Top Down) y características fundamentales de los sistemas; la empresa no pudo satisfacer su necesidad a nivel global.

La creación de algoritmos es basado sobre la técnica descendente, la cual brinda el diseño ideal para la solución de un problema.

Compilación y ejecución: Es el proceso de traducción del programa fuente al lenguaje de máquina. Este proceso se realiza con el compilador y el Sistema Operativo. El resultado, si no hay errores, es la obtención del programa objeto que todavía no es ejecutable directamente. Luego, mediante el Sistema Operativo se realiza la carga del programa objeto con las librerías

del programa compilador, el resultado es un programa ejecutable. Cuando el programa ejecutable se ha creado, se puede ejecutar el programa desde el Sistema Operativo generalmente con sólo su nombre. Si no hay errores se obtiene como salida los resultados del programa.

NOTA: El Turbo Pascal compila y ejecuta con una sola orden.

Verificación y depuración.: Es el proceso de probar que el programa trabaje correctamente y cumpla con los requerimientos del usuario.

- a) **VERIFICAR:** es el proceso de ejecución del programa con una amplia variedad de datos de entrada (test o pruebas) que determinarán si el programa tiene errores.
- b) **DEPURAR:** es el proceso de encontrar los errores del programa y corregir o eliminar dichos errores.

Tipos de errores:

- a) **Errores de compilación:** suelen ser errores de sintaxis.
- b) **Errores de ejecución:** suelen ser instrucciones que la computadora comprende pero que no puede ejecutar. Ejemplo: divisiones por cero, raíces negativas, etc.
- c) **Errores de lógica:** se producen en la lógica del programa, en el diseño del algoritmo. Se detectan porque los resultados son incorrectos.

Documentación y mantenimiento: Consta de la descripción de los pasos a dar en el proceso de resolución de un problema. La documentación de un programa puede ser:

Interna: contenida en las líneas de comentarios del propio programa.

Externa: Incluye el análisis, la especificación del programa, el algoritmo, los manuales de los usuarios, etc. El mantenimiento consiste en la actualización de los programas con los cambios requeridos por el usuario o corrección de posibles errores futuros.

Gráficamente la metodología puede ilustrarse de la siguiente manera:

Especificación funcional

Especificación de argumentos

Mantenimiento

DOCUMENTACIÓN

VERIFICACIÓN Y DEPURACIÓN

EJECUCIÓN

CARGA

¿error?

COMPILACIÓN

EDICIÓN

CODIFICACIÓN

ANALISIS

DISEÑO Algoritmo

Programa Fuente

Programa Fuente tipado en el editor (Turbo Pascal) u otro lenguaje de programación

Programa Objeto

Programa Ejecutable

Resultados

Manuales de documentación

Programas actualizados

NOTA: Esta metodología es una simplificación de la utilizada para el desarrollo de sistemas más complejos. Será aplicada aún más esquemáticamente tal como se plantea en el siguiente ejemplo.

Ejercicio para ilustrar la metodología

Planteamiento del problema:

Calcular la resistencia combinada cuando tres resistencias (R1, R2 y R3) están conectadas en paralelo.

La fórmula de la resistencia combinada es:

$$\frac{1}{1/R1 + 1/R2 + 1/R3}$$

El programa debe producir la siguiente salida: Resistencia Combinada en Ohmios es: xxxxxx

Análisis:

a) Especificación funcional: suponiendo que llamo RC a la Resistencia Combinada, del análisis deduzco que requiero de R1, R2 y R3 (entradas) para poder realizar el cálculo (proceso) de la RC (salida).

ENTRADA

RC

R1

R2

R3

1

$1/R1 + 1/R2 + 1/R3$

RC=

PROCESO SALIDA

b) Especificación de los argumentos: se trata de la documentación de los argumentos utilizados.

- Argumentos identificador tipo longitud condición restricción
- Resistencia 1 R1 real 5.1 variable >0
- Resistencia 2 R2 real 5.1 variable >0
- Resistencia 3 R3 real 5.1 variable >0
- Resistencia Combinada RC real 7.1 variable >0

Diseño.

* Especificación del Programa:

- Nombre del Programa: resistencia
- Función: Calcular la Resistencia Combinada de 3 resistencias conectadas en paralelo
- Argumentos: R1, R2, R3, RC
- Argumentos de Entrada: R1, R2, R3
- Argumentos de Salida: RC
- Lenguaje de Programación: Turbo Pascal
- Algoritmo: (nota: puede utilizarse pseudocódigo o diagrama de flujo).

Pseudocódigo o algoritmo escrito:

- Comienzo (resistencia)
- Leer (R1, R2, R3)
- RC $1/((1/R1) + (1/R2) + (1/R3))$
- Escribir (Resistencia Combinada en Ohmios es:, RC)
- Fin (resistencia) Diagrama de flujo o algoritmo gráfico:

RC

1

$1/R1 + 1/R2 + 1/R3$

fin

RC

resistencia

R1, R2, R3

Edición: tipiar el programa anterior (programa fuente) en el computador.

Compilación y Ejecución del programa editado: para la obtención de los resultados previa corrección de posibles errores de sintaxis o de ejecución.

Verificación y Depuración de los resultados: para corregir los posibles errores de lógica que puedan existir.

2.2. Categorías de Análisis Teórico Conceptual

Colegio Nicolás Infante Díaz, Primer Establecimiento Secundario del Cantón Quevedo.

Quevedo era un altivo Cantón, que en 1.956 tenía 12 años de catonizado, en lo que respecta a la educación, existían varias escuelas de instrucción primaria entre las que citamos: José Isaac Montes, Manuel de Jesús Calle, Nuestra Señora de Fátima, Quito y otros establecimientos de los cuales egresaban numerosos alumnos que no contaban con un colegio secundario.

Quevedo era rico en producción agrícola por lo que había trabajo, lo cual significó mucho para sus habitantes que podían mandar a sus hijos a educarse en la Capital de la República, en Guayaquil, Cuenca, Ambato e incluso a otros países.

Esta situación preocupó a los padres de familia, quienes comprendieron la necesidad de crear un colegio secundario para la juventud anhelante de grandes ideales como era su preparación cultural. Con este propósito, un grupo de entusiastas padres de familia se organizaron en comisiones para realizar las gestiones pertinentes para la consecución de dicho establecimiento.

Las aspiraciones se vieron cristalizadas el 11 de junio de 1.956 como resultado de hombres y mujeres, heraldos de civilización, cuya luz ha brillado

en la abnegada tarea educativa, entre los que citamos a: Dr. Manuel Quintana Miranda, Gilberto Montes Veliz, Sergio Peña, Elisa Marín de Jalón, Dora de Peña, Lady Esteves, Luís Orejuela y otras nobles personas, gracias al apoyo del filántropo Don Camilo Arévalo Govea que era presidente del Ilustre Consejo Cantonal de Quevedo.

Las clases se iniciaron con la asistencia de 12 señoritas y 7 varones, recibiendo las clases en unos chalets de caña ubicados en lo que hoy la calle novena (actualmente edificio del IESS) entre Siete de Octubre y Dra. June Guzmán de Cortés y que pertenecían a la Cruz Roja Cantonal de Quevedo.

El decreto de creación por Resolución Ministerial N° 464 del numeral 8 de la Ley Orgánica de Educación del Normal Rural Municipal, fue firmado por el Ministro de Educación, Ing. José Pons, Ministro de Educación con fecha 28 de Junio de 1.956.

Creado el Normal Rural Municipal, se dignó como Rector al Ilustre historiador Dr. Manuel Quintana Miranda, siendo sus catedráticos: Agro. Carlos Bolaños, Rafael Andino, Aída Quintana, Lady Esteves, Mauro Iturralde, Lidia Gangotena, Irma Coello Pimentel y el Lcdo. Manuel Gradín que es el único profesor que continúa en la cátedra cumpliendo su labor ininterrumpidamente.

El Dr. Carlos Julio Emanuel, expuso en el congreso que el plantel debía llevar el nombre un ilustre hombre de la Provincia de los Ríos, seleccionando el nombre del pequeño luchador que levantó las huestes reivindicadoras de la opresión en que sumía a nuestra patria, este hombre fue el Coronel NICOLAS INFANTE DIAZ de ideas liberales que luchó junto al General Eloy Alfaro.

En 1.959 el Dr. Manuel Quintana consiguió que el concejo presidido por el Sr. Walter Andrade Fajardo, compre al Sr. Gustavo Chong-Qui un local ubicado en las calles Av. Guayaquil y Manabí de la Parroquia Rural San Camilo, albergando numerosos alumnos.

Numerosos profesores normalistas dejaron huellas de sus enseñanzas, especialmente en la preparación de futuros profesores, normalistas entre las que recordamos a la Sra. Luz María de Feijoó, Directora de Prácticas Docentes, Isabel de Navas, María Rosa de Boero, Mario Zambonino, etc.

Después de 3 años de su creación como Normal Rural en el que se preparaba exclusivamente a futuros maestros y que actualmente son profesores en establecimientos primarios y secundarios así como también ejercen la docencia en el propio colegio Nicolás Infante Díaz, adoptó la categoría de colegio de Bachillerato en Humanidades Modernas por Resolución Ministerial N°. 379; a partir de entonces, funcionó el primer curso de Bachillerato, siendo la última promoción de Normalistas en 1.962.

En 1.964, el Sr. Camilo Arévalo Govea, en un noble gesto de filantropía donó un extenso terreno en la parroquia San Camilo, en el que hoy se levanta majestuosamente un funcional edificio con canchas deportivas y un salón de usos múltiples: este coliseo es escenario de diferentes eventos sociales, culturales y deportivos.

En 1.974, cuando era Rector el Lcdo. Simón Avilés Naranjo, se inauguró el Centro Artesanal Nocturno “Gilberto Montes Veliz” nombre del distinguido periodista e historiador, uno de los gestores de la cantonización de Quevedo. En dicho centro se representaban a los jóvenes en carreras técnicas de mecánica y electricidad en aulas debidamente dotadas para su objetivo, tal es así que existe una promoción de técnicos que se graduaron en ese centro que lamentablemente dejó de funcionar con el cambio de Rector.

A través de 45 años de fundación, nuestro colegio ha obtenido un sitio privilegiado, ubicándolo entre los mejores centros de Educación de la Provincia de Los Ríos y de la Patria. En el plano intelectual, social, cultural y deportivo, gracias a la destacada actuación de sus alumnos que honran al Colegio y lo han llenado de gloria Internacional como son: Ivón Calderón y Marcia Quiñónez, testimonio de triunfos estudiantiles son los trofeos que se exhiben en la galería especial del Plantel.

Con mucho orgullo Quevedo ve a exitosos profesionales infantinos que culminaron su carrera universitaria y hoy ponen sus servicios al beneficio de la comunidad.

Destacadas autoridades que han trabajado arduamente en el rectorado para lograr la superación del Nicolás Infante Díaz, han visto realizadas sus

metas ellos son: Dr. Manuel Quintana Miranda, Dr. Eduardo Laborde, Eco. Aída Quintana de Barrezueta, Dr. Néstor Miranda Macas, Dr. Edmundo Betancourt, Lcdo. Simón Avilés Naranjo, Ing. Milton Saltos Muñoz, Lcdo. Eduardo Gaibor, Dr. Orlando Coello, Lcda. Silvia López, Lcdo. Antonio Moreira Bravo, Lcdo. Fausto Hualpa, Lcdo. Edison Martínez, Lcda. Ruth Corrales de Muñoz, Lcdo. Pedro Morales Pincay y Lcdo. Marco Alfaro Álvarez, Rector en el 2.001, Lcdo. Juan Agustín Kure (Rector encargado) Octubre 24/2.002, Lcdo. Fausto Hualpa (Rector Encargado) 06/05/2.006 y la Lcda. Ivón Calderón como Rectora hasta la presente.

El personal Docente del plantel, está conformado por catedráticos que con su noble misión de impartir conocimientos en las áreas: Sociales, Físico – Matemático, Químico – Biólogo e Informática, han hecho posible que la juventud altiva y estudiosa del Nicolás Infante Díaz sean una promesa positiva de la Patria.

El personal Administrativo y de Servicio, muy bien organizado, cumple una intensa labor dentro de la familia infantina, contribuyendo a la buena marcha del plantel.

El colegio Nicolás Infante Díaz tiene un coliseo de deportes, salón de usos múltiples, una cancha de fútbol y más de 10 pabellones de hormigón armado, grandes canchas de básquet, biblioteca, sala audiovisuales, laboratorios de computación, sala de profesores, edificio de la aso de profesores, cabe señalar que la asociación de profesores fundó una Caja de Ahorro que cumple servicios sociales.

2.3. Antecedentes Investigativos

La programación en la lógica se ha convertido en uno de los principales exponentes de la programación declarativa. Sin embargo, aunque se la ha utilizado como herramienta de representación de conocimiento, presenta limitaciones para adaptarse al razonamiento del sentido común en los estudiantes del Colegio Nicolás Infante Díaz. Paralelamente al desarrollo de la

programación en la lógica, los investigadores en inteligencia artificial han continuado produciendo nuevos formalismos con el fin de obtener mejores métodos de razonamiento y de representación de conocimiento de los estudiantes. Uno de estos formalismos es la argumentación rebatible, donde se pone especial énfasis en la noción de argumento. Un argumento para una conclusión constituye una pieza de razonamiento tentativa que un agente inteligente está dispuesto a aceptar para explicar. Si el agente adquiriese luego nueva información, la conclusión junto con el razonamiento que la produjo podría quedar inválida.

Como los agentes inteligentes tienden a razonar en una forma rebatible, sería sumamente interesante disponer de un paradigma de programación donde conclusiones previas lleven a los estudiantes a refutar ante la presencia de mayor o menor información.

Por lo tanto uno de los objetivos de esta Tesis es definir un lenguaje de programación en lógica como una extensión de la programación. Este nuevo lenguaje captura aspectos del razonamiento del sentido común que son difíciles de expresar en la programación y la lógica tradicional.

Desde su concepción, la lógica ha estado ligada al razonamiento científico. La lógica provee un lenguaje preciso para la representación de los conocimientos de los alumnos (as) y para la formulación explícita de nuestras hipótesis y objetivos, además introduce fundamentos necesarios para obtener conclusiones a partir de nuestras premisas, establecer la consistencia de nuestras ciencias y verificar la validez de nuestros argumentos.

Sin embargo, la formación del mismo mediante el uso de la lógica, permite analizar los problemas desde una mejor posición, lo mismo debería ocurrir con los lenguajes de programación.

Aunque la lógica ha sido usada como una herramienta para diseñar tanto computadoras como lenguajes de programación, el uso de lógica directamente como un lenguaje de programación es muy reciente.

2.4. Fundamentación Teórica – Científica

El punto de vista de la lógica de programación como ingeniería es muy diferente. Los programadores, como seres humanos, están ligados a fechas de entrega, cometen errores e introducen fallos en la programación. Por otro lado los requisitos del software no siempre están especificados y pueden ser imprecisos o verse modificados durante el desarrollo, requiriendo cambios en los programas.

Más aún, el proceso de desarrollo no es lineal en la práctica, lo que significa que muchas partes del código de un proyecto pueden ser esbozadas como prototipos y refinadas más adelante. Incluso si estos problemas no fueran suficiente existen restricciones fuertes de seguridad, donde hay ciertos fallos que no son admisibles en absoluto: por ejemplo, que una aplicación externa accede directamente a los recursos de la máquina.

La lógica formal detrás de la programación declarativa a menudo simplifica el razonamiento automático sobre los programas informáticos y los programas informáticos escritos en un lenguaje de programación que está más cerca a la especificación, son más fáciles de escribir y verificar. Esto tiene algunas implicaciones profundas

- ✓ Debido a la amplia expresividad del lenguaje, los mecanismos de ejecución tienen un carácter muy general, lo que se traduce en ocasiones especialmente en los sistemas menos maduros, en un rendimiento pobre, comparado con aquel que ofrecen los lenguajes de bajo nivel.
- ✓ Las transformaciones y los análisis de programas (como evaluación e interpretación abstracta) son en principio más fáciles de aplicar.
- ✓ No describir un problema a un nivel detallado y específico a la máquina implica que existe más libertad por parte de los mecanismos de ejecución para encontrar soluciones.

2.5. Planteamiento de la hipótesis:

2.5.1. Hipótesis General

Mediante la aplicación de las técnicas para el desarrollo de la lógica en la elaboración de programas informáticos, en el aprendizaje de los alumnos del primer año paralelo “A” de bachillerato, del colegio “Nicolás infante Díaz”, del Cantón Quevedo provincia de los Ríos año 2.012?

2.5.2. Hipótesis Particulares

- Aplicando la lógica en la elaboración de programas informáticos influirá en la participación de las actividades prácticas en los estudiantes del primer año de bachillerato paralelo “A” del colegio Nicolás Infantes Díaz en el año 2012.
- Mediante la aplicación correcta de la lógica en la elaboración de programas informáticos para las actividades escolares facilitaremos el desarrollo de las tareas de los estudiantes del primer año bachillerato. Del colegio Nicolás Infantes Díaz.

2.6. Operacionalización De Las Variables

2.6.1. Variable Independiente

Técnicas para el desarrollo de la lógica en la elaboración de programas informáticos.

Cuadro 1. Operacionalización de la primera hipótesis Independiente

CONCEPTOS	CATEGORIA	INDICADORES	ÍTEMS BÁSICOS	TECNICAS E INSTRUMENTOS
<p>La elaboración de programas informáticos es el Conjunto de conocimientos científicos y técnicos que se ocupa del tratamiento de la información por medios automáticos, principalmente mediante equipos electrónicos de procesos de datos que Ayudan en su coeficiente intelectual y en el comportamiento de una sociedad que se encuentra en constante progreso, en los diferentes campos socioeconómicos.</p>	<p>Si aplicamos la elaboración de programas informáticos en las practicas, talleres para el fortalecimiento de las tareas</p>	<p>-Aplicación -Talleres -Conocimientos -Intelectos</p>	<p>-¿Los docentes realizan talleres prácticos durante las clases? -¿Los programas que se aplican son previamente escogidos para actividades concretas? -¿Les brindan la atención necesaria a los programas informáticos en clases?</p>	<p>Entrevista a docentes y encuestas a los padres de familia y alumnos.</p>

2.6.2. Variable Dependiente

Como influye en el aprendizaje de los alumnos del primer año paralelo “A” de Bachillerato del Colegio “Nicolás Infante Díaz”.

Cuadro 2. Operacionalización de la segunda hipótesis Dependiente

CONCEPTOS	CATEGORIA	INDICADORES	ÍTEMS BÁSICOS	TECNICAS E INSTRUMENTOS
<p>Aprendizaje.- Se denomina aprendizaje al proceso de adquisición de conocimientos, habilidades, valores y actitudes, posibilitado mediante el estudio, la enseñanza o la experiencia. Dicho proceso puede ser entendido a partir de diversas posturas, lo que implica que existen diferentes teorías vinculadas al hecho de aprender. La psicología conductista, por ejemplo, describe el aprendizaje de acuerdo a los cambios que pueden observarse en la conducta de un sujeto.</p>	<p>Como influye en el aprendizaje de los alumnos del primer año paralelo "A" de Bachillerato del Colegio "Nicolás Infante Díaz"</p>	<p>Participación -Dedicación -Realización de tareas -Presentación de tareas</p>	<p>¿La utilización de los programas informáticos en clases facilita su aprendizaje? -¿Con la utilización de los programas en clases ayudan a su coeficiente intelectual?</p>	<p>Entrevista a docentes y encuestas a los padres de familia y alumnos.</p>

2.7. Definición de Términos

Actitud: "Disposición de ánimo que hace reaccionar o actuar de una forma determinada delante de una idea, una persona o un hecho concreto.

Implica la tendencia a la acción directa, a favor o en contra del objeto. Las actitudes, junto con los valores y las normas, constituyen uno de los tres tipos de contenidos de enseñanza establecidos en el currículum".

Acto Educativo: Acto sistemático e intencional que realiza el hombre y cuyo objetivo es la consecución del fin de la Educación, es decir, la perfección humana.

Adaptaciones curriculares: Las adaptaciones curriculares son decisiones relativas a la organización de los recursos dirigidos al análisis de los diferentes alumnos. Se busca la adaptación y organización de los elementos espaciales para conseguir así facilitar el aprendizaje y favorecer la autonomía y movilidad en el centro.

Algoritmo: Un algoritmo es una serie de pasos organizados que describe el proceso que se debe seguir, para dar solución a un problema específico. También se puede decir que ALGORITMO es un conjunto finito de pasos en secuencia que indican cómo se resuelve un determinado problema

Alonso: Hombre noble e idealista, sencillo y popular creía ser Don Quijote de la Mancha

Ambler: Define un paradigma de programación como un modelo básico de diseño e implementación de programas

Brunett: Define un paradigma de programación como un modelo básico de diseño e implementación de programas

Capacidades: Las capacidades son aquellas aptitudes que el alumno ha de alcanzar para conseguir un desarrollo integral como persona. En el currículum de una etapa educativa, los objetivos generales de etapa y de área vienen expresados en términos de capacidades.

Compilador: Un compilador es un programa informático que traduce un programa escrito en un lenguaje de programación a otro lenguaje de programación, generando un programa equivalente que la máquina será capaz de interpretar.

Concepción: Acción y efecto de concebir, formación de un concepto en la mente.

Consenso: Acuerdo producido por consentimiento entre todos los miembros de un grupo o entre varios grupos.

Contextual: Perteneciente o relativo al concepto

Cultura: "Conjunto de maneras de vivir y de pensar de un grupo humano en particular"

Currículum: Conjunto de objetivos, contenidos, metodologías y criterios de evaluación de un nivel o etapa educativo.

Educación: La educación consiste en un conjunto de prácticas o actividades ordenadas a través de las cuales un grupo social ayuda a sus miembros a asimilar la experiencia colectiva culturalmente organizada y a preparar su intervención activa en el proceso social.

Enseñar: "Acto por el cual el docente pone al alcance del discente el objeto de conocimiento para que este lo comprenda."

Formación profesional: La Formación Profesional es un conjunto de enseñanzas que, dentro del sistema Educativo regulado capacitan para el desempeño cualificado de las distintas profesiones.

Genesereth: Científico de la Universidad de Stamford creador del correo electrónico inteligente.

Gelfond: Matemático Ruso de Moscú, ocupó las cátedras de análisis matemática.

Ginsberg: Poeta estadounidense. Era hijo de un profesor de inglés y de una maestra de escuela rusa

Lifschitz: Ingeniero civil y político argentino.

Paradigmas: Ejemplo, modelo, esquemas formales en que se organizan las palabras nominales y verbales para sus respectivas flexiones.

Procedimientos: Es el conjunto de acciones ordenadas y orientadas a la realización de una tarea.

Proyecto curricular de centro (PCC): "El Proyecto Curricular de Centro es el proceso de tomas de decisiones por el cual el profesorado de una etapa educativa determinada establece, a partir del análisis del contexto de su centro, una serie de acuerdos acerca de las estrategias de intervención didáctica que va a utilizar, con el fin de asegurar la coherencia de su práctica docente".

Andrés Segovia: Fue un guitarrista clásico español, considerado como el padre del movimiento moderno de la guitarra clásica.

Semántica: La semántica estudia el significado de las palabras a través de los siglos hasta llegar a la actualidad

Sintaxis: Es la ciencia que estudia los componentes que forman sintagmas y las relaciones que entablan estos en si, y, además, también se ocupa del estudio de las oraciones.

Sistematización: Organizar según un sistema.

Sociedad: "Agrupación de individuos con el fin de cumplir mediante la mutua cooperación, todo o alguno de los fines de la vida".

Sociedad: Grupos de personas que se establecen e interactúan entre sí para que a partir de unas leyes puedan convivir y satisfacer las necesidades del colectivo.

Eddie Soloway: Fotógrafo de la National Geographic, docente de Santa Fe Center for Photography y un excelente artista visual, cuyas obras fotográficas se encuentran en numerosas colecciones particulares y corporativas.

Tecnología: Conjunto de conocimientos técnicos

Unidad didáctica: "Unidad de trabajo relativa a un proceso completo de enseñanza-aprendizaje, que no tiene duración temporal fija, y en la cual se precisan el conjunto de objetivos didácticos, bloques elementales de contenido y actividades de aprendizaje y de evaluación.

Whorfiana: La lengua de un hablante monolingüe determina completamente la forma en que éste conceptualiza, memoriza y clasifica la «realidad» que lo rodea (esto se da a nivel fundamentalmente semántico, aunque también influye en la manera de asumir los procesos de transformación y los estados de las cosas expresados por las acciones verbales). Es decir la lengua determina fuertemente el pensamiento del hablante.

Zimmerman: Es el creador de PGP (Pretty Good Privacy), un software de cifrado, firma y autenticación. Zimmerman fue uno de los primeros en realizar un software de cifrado asimétrico, o de clave pública, que fuera sencillo de utilizar por el público en general. Este software era PGP y su código fuente fue publicado por Zimmerman en 1991. El programa podía obtenerse de manera gratuita. En estas primeras versiones de PGP, uno de los algoritmos de cifrado tenía el humorístico nombre de Bass-O-Matic (nombre sacado de un sketch del programa televisivo Saturday Night Live) y el propio nombre de Pretty Good Privacy está inspirado en el del colmado Ralph's Pretty Good Grocery de Lake Wobegon, una ciudad ficticia inventada por el locutor de radio Garrison Keillor.

3. DISEÑO METODOLOGICO DE LA INVESTIGACION

3.1. Metodología Empleada

3.1.1. Método Inductivo.- Va de lo particular a lo general. Es decir, partir del conocimiento de cosas y hechos particulares que se investigaron, para luego, utilizando la generalización y se llega al establecimiento de reglas y leyes científicas.

3.1.2. Método deductivo.- Es el proceso que permite presentar conceptos, principios, reglas, definiciones a partir de los cuales, se analiza, se sintetiza compara, generaliza y demuestra.

3.1.3. Método descriptivo.- Lo utilice en la descripción de hechos y fenómenos actuales por lo que digo: que este método me situó en el presente. No se redujo a una simple recolección y tabulación de datos a los que se acompaña, me integro el análisis reflexión y a una interpretación imparcial de los datos obtenidos y que permiten concluir acertadamente el trabajo

3.1.4. Modalidad de la investigación.- En la presente investigación se empleará la modalidad de campo y documental: De campo porque se realizarán entrevistas a docentes y autoridades y encuestas a padres de familia.

3.1.5. Nivel o tipo de investigación.- Los tipos de investigación a emplearse son: descriptivas y explicativas.

Descriptivas, por cuanto a través de la información obtenida se va a clasificar elementos y estructuras para caracterizar una realidad y, Explicativa, porque permite un análisis del fenómeno para su rectificación

3.2. Población y Muestra

Población: Para esta investigación se considera como población a los docentes total 197 – en donde 10 maestros dan al primer año de bachillerato en informática paralelo “A” y un total de 4.417 estudiantes que conforman el alumnado del colegio – en donde 45 alumnos están cursando el primer año de bachillerato de Informática paralelo “A”.

Muestra: Se considera como muestra a una parte de la investigación que facilita el estudio y hace más eficaz a la investigación, en este caso contaremos con la totalidad de la población.

Las encuestas se aplicarán a la totalidad de la población distribuida de la siguiente manera: 10 docentes y 20 padres de familia de los 45 estudiantes. Los mismos que están distribuidos de la siguiente manera:

MUESTRA

Cuadro 3: Población del primer año de bachillerato

	Población	Muestra	Mujeres	Hombres
Maestros	10	10	6	4
Padre de Famil.	20	20	16	4
Estudiantes	45	45	26	19

Cuadro 4: Población, Frecuencia y Porcentaje del primer año de bachillerato

POBLACION	FRECUENCIA	PORCENTAJE
Docentes	10	13
Estudiantes	45	60
Padres de Familia	20	27
Total	75	100 %

3.3. Técnicas e Instrumentos de Investigación

Durante toda investigación se considerara para el proceso de recolección de datos especialmente el cumplimiento de los objetivos específicos, técnicas como: encuestas, la entrevista, la confiabilidad y valides de las mismas se constatará al momento que se lo aplique.

Entrevista.- Es una conversación que tiene como finalidad la obtención de información. Hay muy diversos tipos de entrevistas: laborales (para informarse y valorar al candidato a un puesto de trabajo), de investigación (realizar un determinado estudio), informativas (reproducir opiniones) y de personalidad (retratar o analizar psicológicamente a un individuo), entre otras.

En una entrevista intervienen el entrevistador y el entrevistado. El primero, además de tomar la iniciativa de la conversación, plantea mediante preguntas específicas cada tema de su interés y decide en qué momento el tema ha cumplido sus objetivos. El entrevistado facilita información sobre sí mismo, su experiencia o el tema en cuestión.

La entrevista se aplicara a los docentes del plantel.

Encuesta.- Una encuesta implica la recopilación y el análisis de las respuestas de grandes grupos de personas, a través de sondeos y cuestionarios diseñados para conocer sus opiniones, actitudes y sentimientos hacia un tema determinado, también tiene la ventaja de permitir obtener información sobre hechos pasados de los encuestadores, gran capacidad para estandarizar datos lo que permite su tratamiento informático y el análisis estadístico.

La misma que será aplicada a los alumnos mediante un cuestionario de preguntas

La fase de análisis e interpretación de resultado previo al planteamiento del proceso se los tabulara y ordenará para representarlo en gráficos que luego serán el material de apoyo para sustentar las variables de la investigación

3.4. Fuentes de Obtención de la Información

Este trabajo lo definimos como netamente investigativo, utilizare como fuentes de investigación y apoyo los siguientes elementos:

Fuentes primarias.

- Encuesta directa al recurso humano y docentes del plantel en relación con la situación del objeto de estudio.

Fuentes secundarias.

- La Institución, Análisis de documentos e Internet.

3.5. Procesamiento de la Información

Para el desarrollo de la investigación se cumplirá con el siguiente procedimiento:

- Mediante la aplicación del método analítico-descriptivo se analizarán y sintetizarán los principales planteamientos teóricos sobre conceptualizaciones básicas de categorías en torno al problema de investigación y que se constituyeron en los sustentos teóricos del proceso de investigación desarrollado.
- Apoyados en el marco teórico previamente construido se operacionalizaron las variables de cada una de las hipótesis planteadas hasta llegar a determinar los indicadores de la investigación, a partir de los cuales se elaboraron los instrumentos de investigación para la recopilación de información de campo de los diferentes segmentos de la población estudiada.
- Se aplicarán encuestas a los docentes, padres de familia y alumnos(as) que componen la muestra con preguntas previamente estructuradas.
- Una vez recolectada la información se procederá a tabular, utilizando tablas de frecuencias, cuadros ilustrativos y representaciones gráficas lo que permitirá el análisis e interpretación de los datos.
- Con toda la información recabada, una vez, analizada e interpretada, se procederá a verificar las hipótesis planteadas, para luego establecer conclusiones.

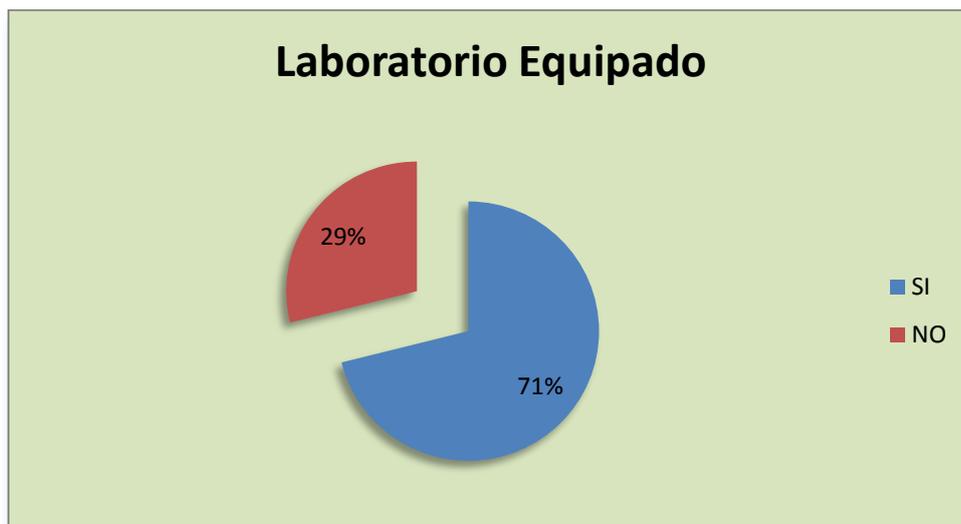
- Una vez conocida la realidad, se plantearán los lineamientos alternativos que permitieron solucionar o por lo menos, mejorar la problemática investigada.
- Concluido el trabajo de investigación, se procederá a preparar el informe final, en el cual se trasmite de forma clara, precisa y sistemática los resultados de la investigación; con la finalidad de contribuir al fortalecimiento institucional.

4. ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

Encuesta a Estudiantes.

1. ¿Se encuentra equipado el laboratorio con las necesidades del estudiante para realizar programas de aplicación?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
ALUMNOS	SI	32	71
	NO	13	29
TOTAL		45	100



Análisis: El resultado de los estudiantes del Primer año de Bachillerato paralelo "A" que se han encuestado el 71 % nos indican que, el laboratorio si se encuentra equipado con todas las necesidades del estudiante y el 29 % nos dice que no se encuentra equipado con todas las necesidades del estudiante.

Interpretación: Este resultado indica que más de la mitad de los encuestados se encuentran felices con el equipamiento del laboratorio.

2. ¿Le entiendes al Maestro cuando elabora programas informáticos?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
ALUMNOS	MUCHO	12	27
	POCO	15	33
	NADA	18	40
TOTAL		45	100



Análisis: El resultado de esta encuesta da como resultado que el 27 % de los alumnos encuestados le entienden al Maestro las clases impartidas al alumnado, el 33 % le entiende poco al Maestro y el 40 % no le entiende nada al profesor.

Interpretación: Se puede decir que de acuerdo al resultado obtenido en esta encuesta existen falencias por parte del maestro para llegar con sus clases al dicente.

3. ¿Realizan talleres de estudio con el Maestro para desarrollar la lógica en la programación?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
ALUMNOS	SI	35	78
	NO	10	22
TOTAL		45	100

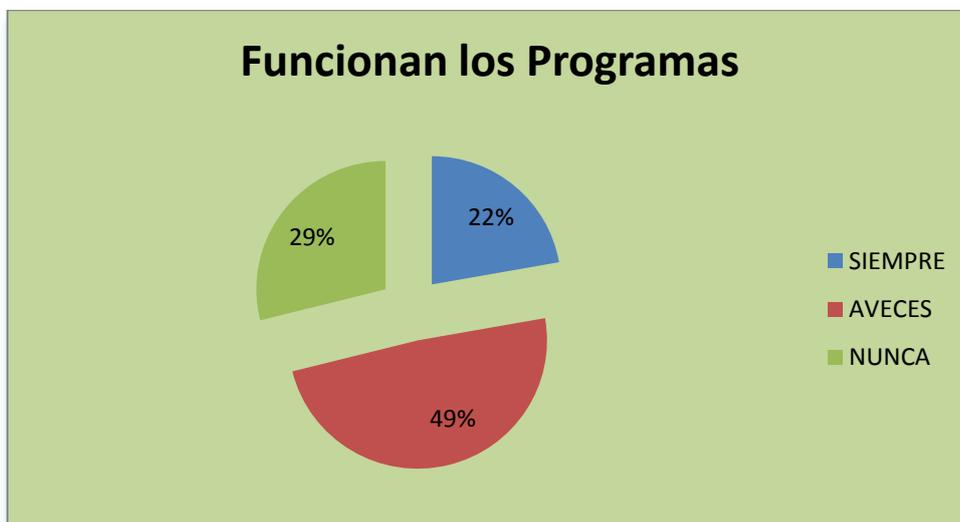


Análisis: El resultado de esta encuesta da como resultado que el 78 % de los alumnos encuestados dicen que el maestro si realiza talleres y el 22 % de los alumnos dicen que no realizan talleres de estudio con el Maestro.

Interpretación: Según el resultado se puede decir que el maestro debe seguir trabajando con los estudiantes en la realización de talleres, para que el alumno tenga una mejor captación de los diferentes problemas que se realizan en el aula de clases.

4. ¿Funcionan todos los programas informáticos dados en clase?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
ALUMNOS	SIEMPRE	10	22
	AVECES	22	49
	NUNCA	13	29
TOTAL		45	100

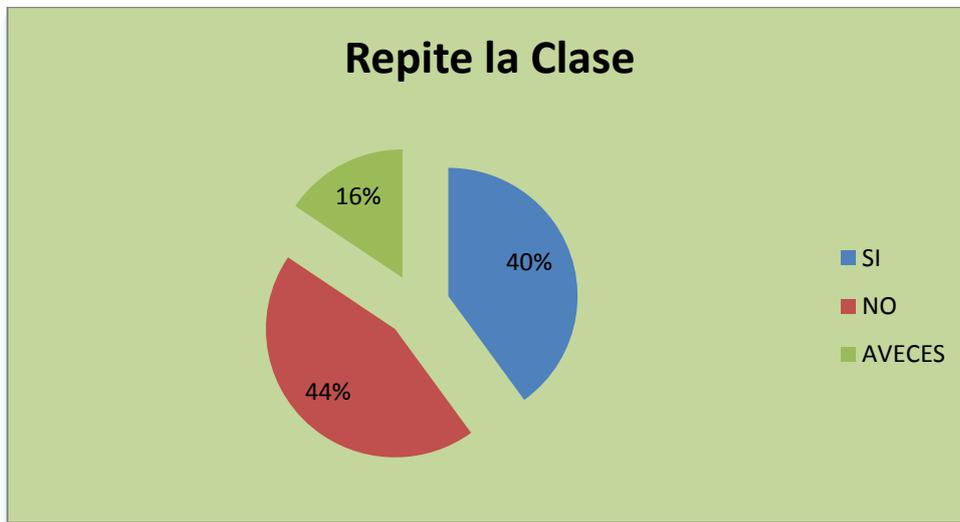


Análisis: El resultado de esta pregunta da como resultado que el 22 % de los alumnos encuestados dicen que los programas dados en clase funcionan siempre, el 49 % dicen que los programas funcionan a veces y el 29 % de los alumnos dicen que nunca funcionan los programas que se realizan en clase.

Interpretación: De acuerdo a este resultado se puede decir que el maestro debe seguir trabajando en el desarrollo de la lógica de elaboración de programas informáticos en los estudiantes, de esa manera el estudiante va a levantar su interés en aprender a desarrollar los problemas que se efectúan en clase.

5. ¿Cuando no entiendes un programa, ¿repite la clase el maestro?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
ALUMNOS	SI	18	40
	NO	20	44
	AVECES	7	16
TOTAL		45	100

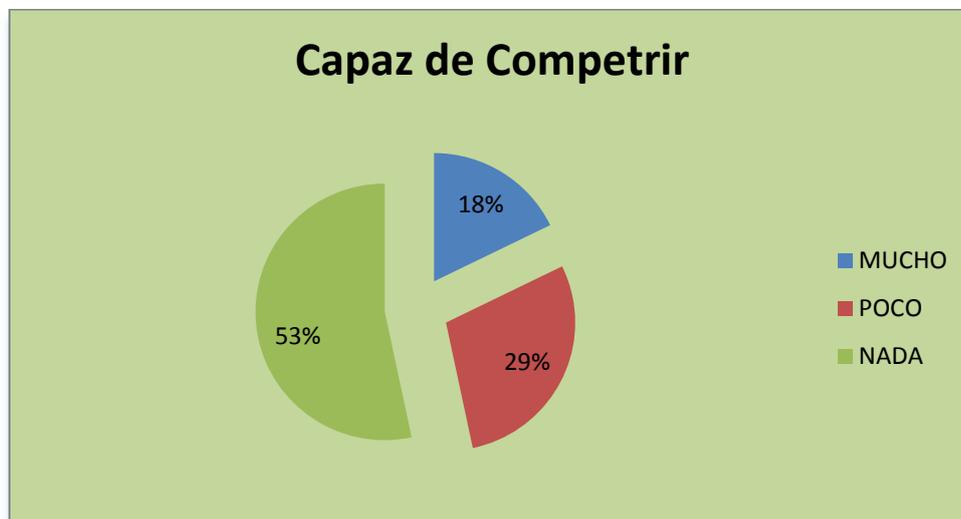


Análisis: El resultado de esta encuesta en la pregunta 5 da como resultado que el 40 % de los alumnos encuestados dicen que el maestro si repite la clase cuando no entienden, el 44 % de los alumnos dicen que el maestro no repite la clase cuando no entienden y el 16 % dicen que el maestro a veces repite la clase cuando no entienden.

Interpretación: Con el resultado se puede decir que el maestro debe repetir la clase ya que la mayoría de los estudiantes no entienden la clase que ha dictado o ha explicado el maestro en ese el día, es decir debe de cambiar la metodología.

6. ¿Te sientes capaz en competir con otro colegio realizando programas informáticos?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
ALUMNOS	MUCHO	8	18
	POCO	13	29
	NADA	24	53
TOTAL		45	100

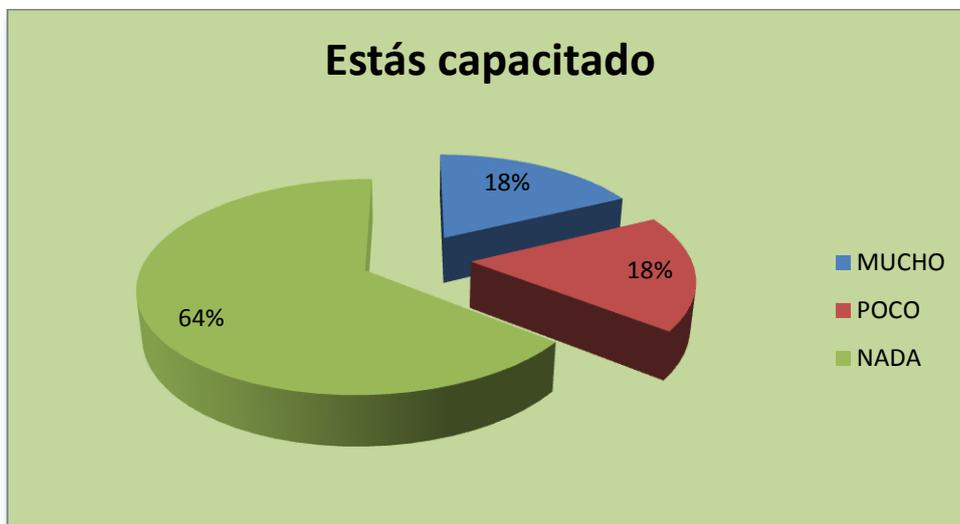


Análisis: El resultado de esta encuesta en la pregunta 6 da como resultado que el 18 % de los alumnos encuestados dicen que se sienten capaces de competir con otro colegio programando, el 28 % de los alumnos dicen que se sienten poco capaces de competir con otros colegios y el 53 % no están capacitados para realizar una competencia con otro colegio.

Interpretación: De acuerdo a este resultado podemos decir que hay que trabajar en la motivación de los alumnos para incentivarlos que las personas estamos preparadas y somos capaces de competir a nivel académico con cualquier otra institución.

7. ¿Estás capacitado para enseñar programa informáticos a otra persona?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
ALUMNOS	MUCHO	8	18
	POCO	8	18
	NADA	29	64
TOTAL		45	100



Análisis: El resultado de esta encuesta en la pregunta 7 da como resultado que el 18 % de los alumnos encuestados se encuentran muy capacitados para poder enseñar a otra persona a realizar programas informáticos, mientras que el 18 % de los alumnos dicen que se sienten poco capaces para enseñar a programar y el 64 % dicen que no están capacitados para enseñar a programar a otra persona.

Interpretación: De acuerdo a este resultado se puede decir que los alumnos no se encuentran completamente capacitados para enseñar a programar a otra persona ajena de la institución.

8. ¿Te sientes capaz de realizar un sistema Informático?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
ALUMNOS	SI	8	18
	NO	37	82
TOTAL		45	100



Análisis: El resultado de esta encuesta en la pregunta 8 da como resultado que el 18 % de los alumnos encuestados dicen que se sienten capaces de realizar un Sistema Informático, mientras que el 82 % de los alumnos encuestados dicen que no se sienten capaces de realizar un Sistema Informático.

Interpretación: Según el resultado del análisis se puede decir que los estudiantes no están capacitados al 100 % para poder realizar un Sistema Informático, es decir el Docente tiene que trabajar mucho en los alumnos para incentivarlos a realizar estos Sistemas Informáticos.

9. ¿Tienes computadora en casa para realizar los programas informáticos?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
ALUMNOS	SI	28	62
	NO	17	38
TOTAL		45	100



Análisis: El resultado de esta pregunta en la encuesta realizada da como resultado que el 62 % de los alumnos encuestados dicen que tienen computadoras en casa y pueden practicar los ejercicios que se dan en clase, mientras que el 38 % de los estudiantes no poseen computadoras en sus hogares y no pueden practicar los ejercicios.

Interpretación: De acuerdo a este resultado se puede decir que los estudiantes poseen computadoras personales y pueden realizar y practicar los ejercicios que se dicten en clase.

10. ¿Te gusta elaborar programas informáticos?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
ALUMNOS	MUCHO	12	27
	POCO	20	44
	NADA	13	29
TOTAL		45	100



Análisis: El resultado de esta encuesta en la pregunta 10 da como resultado que el 27 % de los alumnos encuestados dicen que les gusta mucho la programación, mientras que el 44 % de los alumnos encuestados les gusta poco la programación y el 29 % de los alumnos encuestados no les gusta para nada la programación.

Interpretación: Según este resultado se puede decir que los estudiantes se han equivocado al momento de elegir su carrera ya más de la mitad les gusta poco.

11. Cuándo tienes una inquietud en cuanto a un programa informático, le preguntas a tu Maestro ¿cómo te responde?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
ALUMNOS	CORTES	8	18
	PACIENTE	10	22
	AMABLE	10	22
	DESCORTES	12	27
	IMPACIENTE	3	7
	DESPOTA	2	4
TOTAL		45	100



Análisis: El resultado de esta encuesta en la pregunta 11 da como resultado que el 18 % de los alumnos encuestados dicen que el maestro es cortés cuando se encuentran con dudas o no entienden, el 22 % de los alumnos encuestados dicen que es paciente, el 22 % de los alumnos dicen que amable, el 27 % de los alumnos dicen que es descortés, el 7 % es impaciente y el 4% dicen que es déspota y no contesta las dudas en el estudiante.

Interpretación: De acuerdo a este resultado se puede decir que los estudiantes no están de acuerdo con el trato del maestro hacia ellos y esto puede una de las causales que haya problemas en el proceso de aprendizaje en los alumnos.

Encuesta a los Docentes y Autoridades.

1. ¿Los alumnos cumplen a cabalidad con las tareas?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
DOCENTES	SI	4	40
	NO	6	60
TOTAL		10	100



Análisis: El resultado de esta encuesta en la pregunta 1 da como resultado que el 40 % de los Maestros encuestados dicen que los estudiantes si cumplen con las tareas, mientras que el 60 % de los maestros encuestados dicen que los estudiantes no cumplen con las tareas.

Interpretación: De acuerdo a este resultado se puede decir que los estudiantes no están cumpliendo con las tareas encomendadas ya que más de la mitad de los maestros dicen que no cumplen.

2. ¿Cada que tiempo los padres de familia se acercan a preguntar por las notas de su representado?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
DOCENTES	DIARIO	3	30
	SEMANAL	4	40
	MENSUAL	1	10
	NUNCA	2	20
TOTAL		10	100

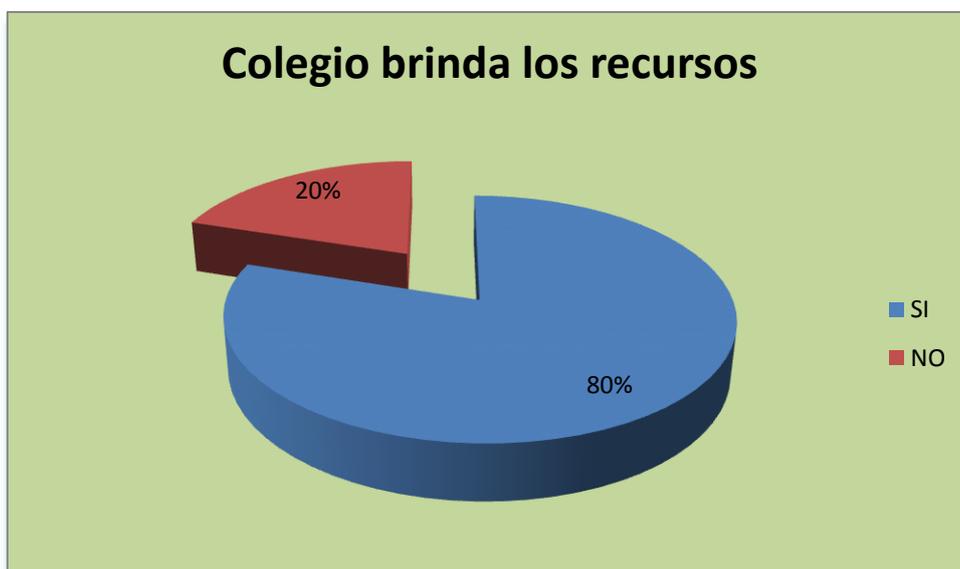


Análisis: El resultado de esta encuesta en la pregunta 2 da como resultado que el 30 % de los Maestros encuestados dicen que los padres de familia se acercan a preguntar por las notas de su representado diariamente, mientras que el 40 % de los maestros encuestados dicen que los padres de familia se acercan semanal, el 10 % mensualmente y el 20 % nunca se acercan.

Interpretación: De acuerdo a este resultado se puede decir que los padres de familia asisten a preguntar las notas de su representado a los maestros semanalmente.

3. ¿El Colegio brinda los recursos necesarios para desarrollar sus clases?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
DOCENTES	SI	8	80
	NO	2	20
TOTAL		10	100



Análisis: El resultado de esta encuesta en la pregunta 3 da como resultado que el 80 % de los Maestros encuestados dicen que el Colegio le brinda los recursos necesarios para desarrollar sus clases y el 20 % de los maestros encuestados dicen que el colegio no les brinda los recursos necesarios para desarrollar sus clases.

Interpretación: De acuerdo a este resultado podemos decir que el colegio le brinda al maestro los recursos necesarios para poder desarrollar sus clases.

4. ¿Le brinda la oportunidad de recuperación a los Alumnos (as)?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
DOCENTES	SI	9	90
	NO	1	10
TOTAL		10	100

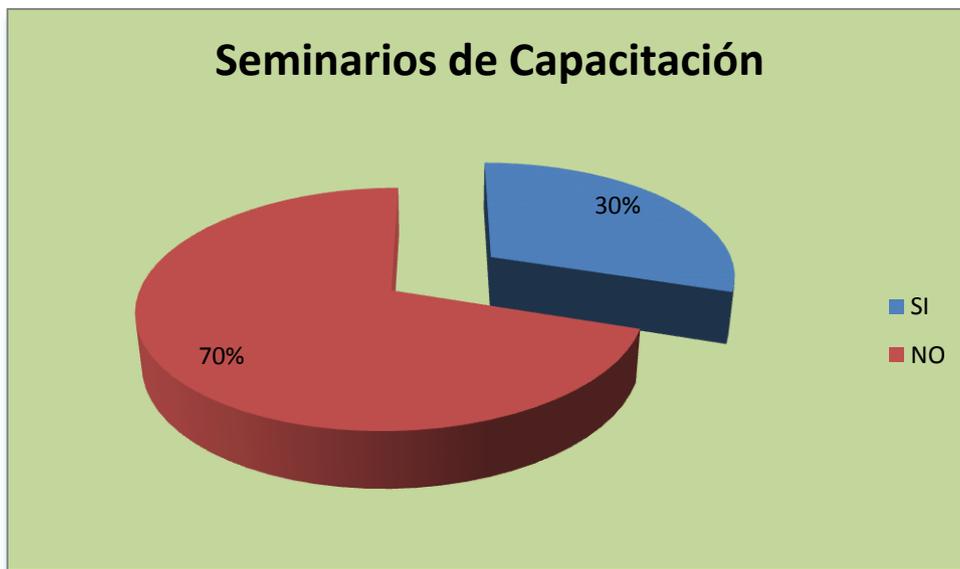


Análisis: El resultado de esta encuesta en la pregunta 4 nos da como resultado que el 90 % de los Maestros encuestados dicen que ellos le brindan la oportunidad al alumno(a) para que se recupere y el 10 % de los maestros encuestados dicen que no le brindan la oportunidad al alumno(a) para que se recupere.

Interpretación: De acuerdo a este resultado podemos decir que todos los maestros están de acuerdo en brindarle oportunidades de recuperación a los alumnos(as), para que no tengan ningún problema a futuro.

5. ¿Da seminarios de Capacitación a los Estudiantes?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
DOCENTES	SI	3	30
	NO	7	70
TOTAL		10	100



Análisis: El resultado de esta encuesta en la pregunta 5 nos da como resultado que el 30 % de los Maestros encuestados dicen que ellos si dan a los estudiantes seminarios de capacitación, mientras que el 70 % de los maestros encuestados dicen que no dan seminarios de capacitación a los Estudiantes.

Interpretación: De acuerdo a este resultado podemos decir que todos los maestros deben de realizar seminarios talleres a los alumnos para evitar cualquier inconveniente a futuro.

6. ¿Realiza Ejercicios de Algoritmo en Clase?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
DOCENTES	SI	8	80
	NO	2	20
TOTAL		10	100



Análisis: El resultado de esta encuesta en la pregunta 6 nos da como resultado que el 80 % de los Maestros encuestados dicen que si realizan ejercicios de algoritmo en clases, mientras que el 20 % de los maestros encuestados dicen que no, ellos no realizan ejercicios de algoritmo en clases.

Interpretación: De acuerdo a este resultado podemos decir que los maestros si están realizando ejercicios de algoritmo en clases, para que el alumno pueda desarrollar las destrezas necesarias en la elaboración de programas informáticos.

7. ¿Cada cuando tiempo evalúa a los Estudiantes para comprobar su aprendizaje en cuanto a la técnica de programación?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
DOCENTES	DIARIO	3	30
	SEMANTAL	3	30
	MENSUAL	3	30
	TRIMESTRAL	1	10
	NUNCA	0	0
TOTAL		10	100



Análisis: El resultado de esta encuesta en la pregunta 7 da como resultado que el 30 % de los Maestros encuestados dicen que evalúan a los estudiantes diariamente, mientras que el 30 % de los maestros encuestados dicen que los evalúan semanalmente, el 30 % los evalúan mensualmente, el 10 % lo realizan trimestralmente y el 0 % nunca.

Interpretación: De acuerdo a este resultado se puede decir que los maestros si evalúan a sus estudiantes normalmente para ganar los puntos necesarios y no tener problemas de nota al final del trimestre.

8. ¿Incentiva a los Estudiantes a participar en clases utilizando la lógica en la elaboración de programas informáticos?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
DOCENTES	SI	7	70
	NO	3	30
TOTAL		10	100

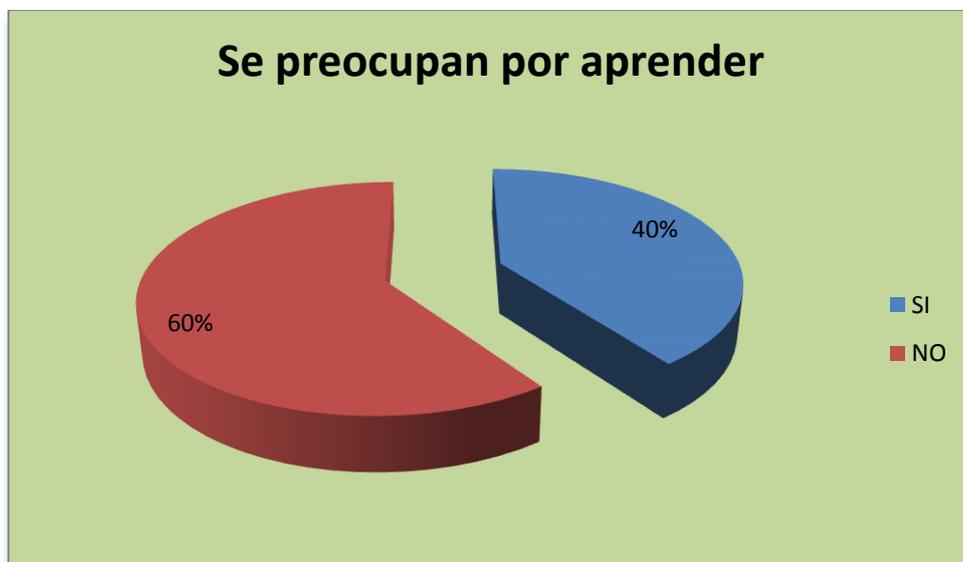


Análisis: El resultado de esta encuesta en la pregunta 8 nos da como resultado que el 70 % de los Maestros encuestados dicen que si incentivan a los estudiantes a participar en clases, mientras que el 30 % de los maestros encuestados dicen que ellos no incentivan a participar en clase a los estudiantes.

Interpretación: De acuerdo a este resultado podemos decir que más de la mitad de los maestros están incentivando a los estudiantes a participar en clases, ya que esto ayudará al estudiantado a elevar sus conocimientos y preocupación por sus estudios.

9. En el aula, ¿existen estudiantes que demuestran interés por aprender a desarrollar las técnicas de programación?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
DOCENTES	SI	4	40
	NO	6	60
TOTAL		10	100

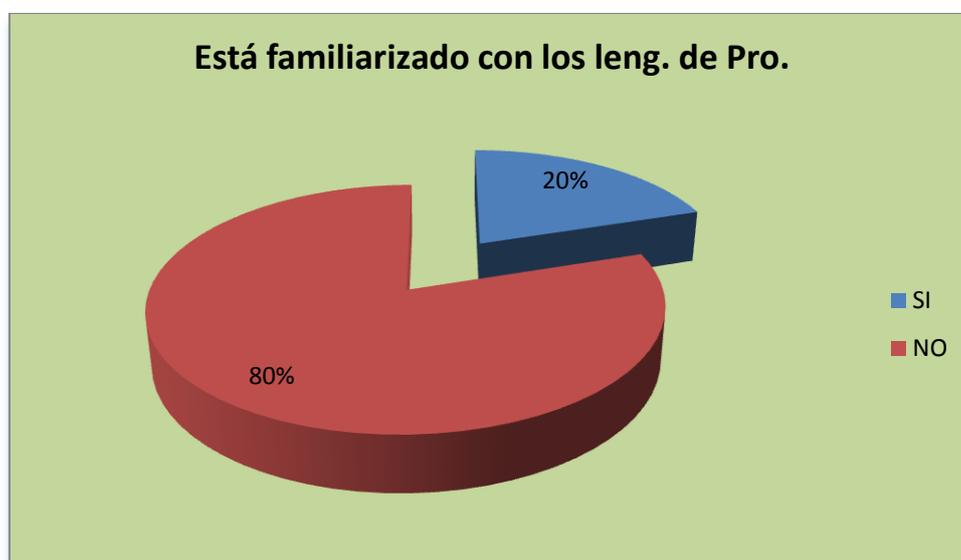


Análisis: El resultado de esta encuesta en la pregunta 9 nos da como resultado que el 40 % de los Maestros encuestados dicen si existen estudiantes que se preocupan por aprender a los conceptos y definiciones, mientras que el 60 % de los maestros encuestados dicen que en el aula de clases existen estudiantes que no se preocupan por aprender los conceptos y las definiciones.

Interpretación: De acuerdo a este resultado podemos decir que los estudiantes están fallando en el proceso del aprendizaje, ya que existe más de la mitad del curso que no se interesan en aprender.

10. ¿Está familiarizado con los diferentes lenguajes de Programación?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
DOCENTES	SI	2	20
	NO	8	80
TOTAL		10	100



Análisis: El resultado de la pregunta 10 de la encuesta vemos que el 20% de los maestros encuestados si están familiarizados con los diferentes lenguajes de programación, mientras que el 80 % de los Maestros encuestados dicen que no están familiarizados con los diferentes lenguajes de programación.

Interpretación: De acuerdo a este resultado podemos decir que más de la mitad de los maestros encuestados no están familiarizados con los diferentes lenguajes de programación, es decir hay que poner mayor énfasis en las investigaciones e interpretaciones de los lenguajes de programación.

Encuesta realizada a los padres de Familia

1. ¿Conoce Ud. si el Docente realiza ejercicios lógicos en su clase?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
PADRE DE FAMILIA	SI	10	50
	NO	6	30
	AVECES	4	20
TOTAL		20	100



Análisis: El resultado de la pregunta 1 de la encuesta vemos que el 50% de los padres de familia encuestados dicen que los docentes realizan los ejercicios lógicos en clase, mientras que el 30 % de los padres de familia encuestados dicen que no están realizando ejercicios lógicos en clase y el 20% de los padres de familia dicen que a veces realizan ejercicios lógicos en clase.

Interpretación: De acuerdo a este resultado podemos decir que más de la mitad de los padres de familia encuestados dicen que el docente realiza ejercicios lógicos en clase, lo cual indica que el Docente está trabajando de la mejor manera.

2. ¿Ud. cree que los ejercicios realizados en clase están acorde con el tema tratado?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
PADRE DE FAMILIA	SI	5	25
	NO	8	40
	AVECES	7	35
TOTAL		20	100



Análisis: El resultado de la pregunta 2 de la encuesta vemos que el 25% de los padres de familia encuestados dicen que los ejercicios que los docentes realizan en clase van de acorde con el tema tratado, mientras que el 40 % de los padres de familia encuestados dicen que los ejercicios realizados en clase no están acorde al tema tratado y el 35% de los padres de familia dicen que aveces van acorde a los temas tratados en clase.

Interpretación: De acuerdo a este resultado podemos decir que más de la mitad de los padres de familia encuestados dicen que los ejercicios realizados en clase no van acorde al tema tratado en clase.

3. ¿La elaboración de ejercicios lógicos en clase motiva la participación de su hijo?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
PADRE DE FAMILIA	SI	15	75
	NO	3	15
	AVECES	2	10
TOTAL		20	100



Análisis: El resultado de la pregunta 3 de la encuesta vemos que el 75% de los padres de familia encuestados dicen que la elaboración de ejercicios lógicos en clases motiva la participación del hijo, mientras que el 15 % de los padres de familia encuestados dicen que no motivan la participación de los hijos y el 10% de los padres de familia dicen que a veces se motivan y participan.

Interpretación: De acuerdo a este resultado podemos decir que más de la mitad de los padres de familia encuestados dicen que los ejercicios realizados en clase los motiva a participar en desarrollo de los mismos.

4. ¿La elaboración de ejercicios en clase facilita el aprendizaje de su hijo?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
PADRE DE FAMILIA	SI	10	50
	NO	3	15
	AVECES	7	35
TOTAL		20	100



Análisis: El resultado de la pregunta 4 de la encuesta vemos que el 50% de los padres de familia encuestados dicen que los ejercicios en clase facilita el aprendizaje de sus representados, mientras que el 15 % de los padres de familia encuestados dicen que los ejercicios en clase no facilitan el aprendizaje de sus representados y el 35 % de los padres de familia dicen que a veces los ejercicios realizados en clase facilita el aprendizaje de sus representados

Interpretación: De acuerdo a este resultado podemos decir que la mitad de los padres de familia encuestados dicen que los ejercicios realizados en clase ayudan en el aprendizaje de sus representados, lo cual está correcto.

5. Con la aplicación de ejercicios lógicos en clase, ¿su hijo participa activamente?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
PADRE DE FAMILIA	SI	8	40
	NO	4	20
	AVECES	8	40
TOTAL		20	100



Análisis: El resultado de la pregunta 5 de la encuesta vemos que el 40% de los padres de familia encuestados dicen que con la aplicación de los ejercicios lógicos realizados en clase ayuda a sus representados a participar en la clase activamente, mientras que el 20 % de los padres de familia encuestados dicen que la aplicación de los ejercicios lógicos no ayudan a participar en la clase activamente y el 40 % de los padres de familia dicen que a veces la aplicación de los ejercicios ayudan a participar a sus representados.

Interpretación: De acuerdo a este resultado podemos decir que hay una división de criterios, que los ejercicios ayuden a participar activamente a sus representados, lo cual indica que el Docente debe trabajar de la mejor manera.

6. Su hijo, ¿realiza los ejercicios lógicos en casa sin ningún problema?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
PADRE DE FAMILIA	SI	4	20
	NO	10	50
	AVECES	6	30
TOTAL		20	100



Análisis: El resultado de la pregunta 6 de la encuesta vemos que el 20% de los padres de familia encuestados dicen que sus representados realizan los ejercicios lógicos sin ningún problema, mientras que el 50 % de los padres de familia encuestados dicen que no realizan sus representados los ejercicios lógicos sin ningún problema y el 30% de los padres de familia dicen que a veces realizan ejercicios lógicos sin ningún problema.

Interpretación: De acuerdo a este resultado podemos decir que el porcentaje es bastante elevado, es decir los alumnos no pueden realizar los ejercicios lógicos sin ningún problema, el maestro tiene que trabajar de la mejor manera para cambiar ese porcentaje negativo.

7. Su hijo, ¿al elaborar un ejercicios lógico pide ayuda a otra persona?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
PADRE DE FAMILIA	SI	6	30
	NO	8	40
	AVECES	6	30
TOTAL		20	100

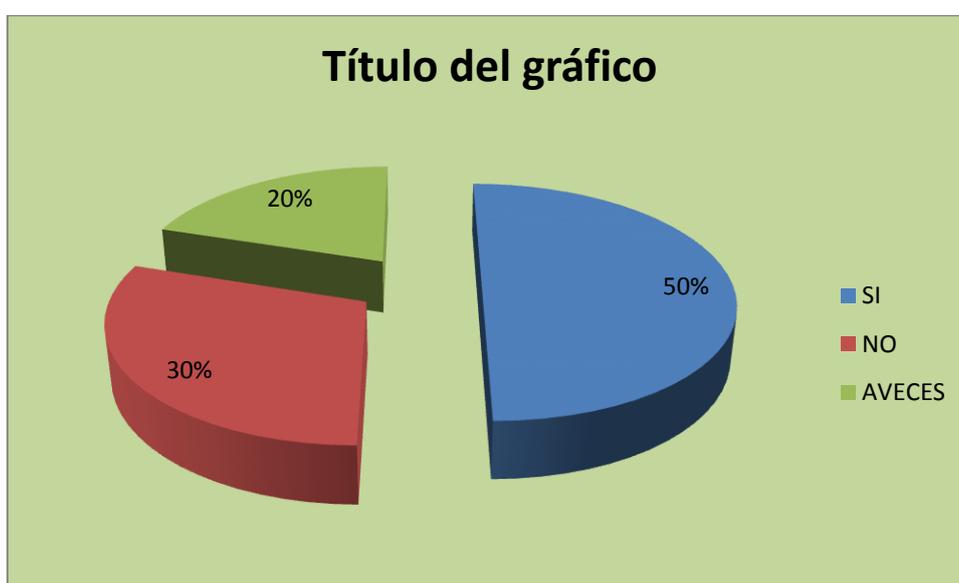


Análisis: El resultado de la pregunta 7 de la encuesta vemos que el 30% de los padres de familia encuestados dicen que sus representados si necesitan la ayuda de otra persona para realizar los ejercicios, mientras que el 40 % de los padres de familia encuestados dicen que no necesitan la ayuda de otra persona para realizar los ejercicios y el 30% de los padres de familia dicen que aveces necesitan la ayuda de otras personas para realizar los ejercicios.

Interpretación: De acuerdo a este resultado podemos decir que existen problemas de aprendizaje por parte del alumnado y no están cumpliendo con los diferentes ejercicios que se envían a casa es decir, los maestros tienen que trabajar mucho con los estudiantes.

8. Su hijo, ¿se adapta a las enseñanzas del docente cuando imparte sus clases?

POBLACION	VARIABLE	FRECUENCIA	PORCENTAJE %
PADRE DE FAMILIA	SI	10	50
	NO	6	30
	AVECES	4	20
TOTAL		20	100



Análisis: El resultado de la pregunta 8 de la encuesta vemos que el 50% de los padres de familia encuestados dicen que sus hijos se adaptan a las enseñanzas de los docentes cuando imparten la cátedra, mientras que el 30 % de los padres de familia encuestados dicen que sus hijos no se adaptan a las enseñanzas de los docentes y el 20% de los padres de familia dicen que a veces se adaptan a las enseñanzas de los maestros.

Interpretación: De acuerdo a este resultado podemos decir que hay una división de criterios ya que los docentes no comparten de la mejor manera sus conocimientos a los estudiantes.

4.1. Verificación De Hipótesis

De acuerdo a la investigación realizada se puede indicar que la hipótesis es positiva; ya que la carencia de la lógica en la elaboración de programas informáticos afecta el comportamiento intelectual de los estudiantes del primer año "A" de bachillerato del colegio "Nicolás Infante Díaz", del Cantón Quevedo Provincia de los Ríos en el año 2012. Esto hace que en la actualidad los docentes y padres de familia no logran comprender a los dicentes cuáles son sus carencias de conocimientos al momento de realizar sus tareas, esto ha generado que muchos de los alumnos(as) fracasen en sus estudios superiores.

4.2. Conclusiones y Recomendaciones

Conclusiones

Luego de haber establecido los resultados de la investigación, se establecen las siguientes conclusiones:

- Muchos de los Estudiantes no poseen un computador personal, lo cual indica que no pueden practicar los programas informáticos realizados en clases, esto indica que mandan a realizar sus trabajos a otras instituciones.
- Las autoridades del plantel deben trabajar con los Docentes dándole seminarios talleres para que estos cuenten con una herramienta que sirva para el desarrollo de las destrezas, habilidades e intelecto en los estudiantes para la elaboración de programas informáticos.
- La dificultad del aprendizaje y el desarrollo de las técnicas de programación, se debe a que la mayoría de los alumnos se han equivocado al momento de elegir su especialidad, los docentes del colegio están consientes que deben trabajar más, cuando están en la etapa de elegir su especialidad.
- El docente debe de prepararse de acuerdo a la planificación curricular, para que al momento de impartir su cátedra a los estudiantes tenga un excelente dominio y evite improvisar en el aula de clase.

Recomendaciones

De acuerdo a las conclusiones obtenidas del estudio de diagnóstico sobre la metodología empleada por los docentes de las asignaturas del área de informática, se establecen las siguientes recomendaciones:

- Es necesario que se den charlas para que los estudiantes realicen la elección de la carrera adecuada y no sucedan fracasos académicos.
- Se recomienda la planificación como una exigencia de nuestro tiempo, evita en lo posible el desajuste, mantiene una constante retroalimentación que nutre el proceso docente, una visión acertada abarata costos, por ello la planificación del área de informática debe hacerse de acuerdo a los avances tecnológicos y basarse en un trabajo sistemático que permita una práctica cada vez más lúcida, científica y profesional.
- La labor del docente es un trabajo de investigación, en el cual se descubren los elementos que determinarán la elaboración de procesos de planificación, por lo tanto esto ayudara a una continua reformulación mediante mecanismos de retroalimentación de conocimientos.

5. MARCO ADMINISTRATIVO

5.1. Recursos

Humanos

- Una persona para realizar las encuestas a cada una de las personas implicadas en el trabajo de investigación.
- Una persona para realizar las tabulaciones de las encuestas realizadas.
- Un director de tesis
- Un lector de tesis

Materiales

- Un computador
- Una Impresora
- Internet
- Hojas A-4
- Una oficina
- Material Logístico

Presupuesto

Cuadro 5: Gastos Realizados

CONCEPTO	VALOR UNITARIO	VALOR / SUB TOTAL
Una persona para realizar las encuestas y la tabulación	\$ 10,00 C/día	\$ 10,00
Un Director de Tesis	—	—
Un Lector de Tesis	—	—
Un Computador	\$ 1,00 C/hora	\$ 42,00
Viáticos y Subsistencia	\$ 5,00 C/Día	\$ 40,00
Material Logístico	\$ 50,00	\$ 200,00
Elaboración de 75 encuestas	\$ 0,05 C/una	\$ 7,50
Elaboración de la Tesis		\$ 200,00
TOTAL		\$ 499,50

6. PROPUESTA ALTERNATIVA

6.1. Tema

Implementación de cursos y seminarios de capacitación en la elaboración de programas informáticos para mejorar las técnicas de programación, a los Docentes del colegio “Nicolás Infante Díaz”, del Cantón Quevedo Provincia de los Ríos.

6.2. Justificación

Los jóvenes estudiantes tienen necesidades de investigar, se van a analizar los métodos, estrategias y aplicación de cada uno de los operadores que existen, de tal manera, esta técnica va a fortalecer el aprendizaje en cada uno de los estudiantes, se van a preparar dicentes capaces que van a poner en práctica sus conocimientos en el campo profesional. Se Cree que los estudiantes y profesores del colegio fiscal mixto “Nicolás Infante Díaz”, deben ser lo suficientemente reflexivos y analíticos para darse cuenta de la realidad actual, al estudiar con creatividad y responsabilidad les ayudará a alcanzar el éxito, esto permite buscar el bien personal así como el bien común.

6.3. Fundamentación

Al evaluar el plan educativo durante los últimos años, a nivel institucional se detectó un índice bastante bajo en los estudiantes, con rendimiento regular en su enseñanza y aprendizaje, la causa de esta problemática es sin

duda la falta del desarrollo de las técnicas en la elaboración de programas informáticos.

6.4. Contenido

Cuando evaluamos un plan estratégico de tipo operativo, nos damos cuenta que durante muchos años, se ha detectado a nivel institucional un alto índice de estudiantes que no desarrollan una lógica adecuada porque han escogido equivocadamente la especialidad, es necesario atender esta problemática, con el fin de establecer y eliminar el fracaso académico, mediante las prácticas metodológicas interactivas, esto va a ayudar a reflexionar a los alumnos(as) en el desarrollo de la conciencia.

Estructuras repetitivas: El computador está especialmente diseñado para aplicaciones en las que una operación o un conjunto de ellas deben repetirse muchas veces. En este sentido, definiremos bucle o lazo (Loop), como un segmento de un programa cuyas instrucciones se repiten bien un número determinado de veces o mientras se cumpla una determinada condición.

Es imprescindible que se establezcan mecanismos para controlar esta tarea respectiva, ya que si estos no existen, el bucle puede convertirse en un proceso infinito.

Los procesos que se repiten varias veces en un programa necesitan en muchas ocasiones contar el número de repeticiones habidas. Una forma de hacerlo es utilizar una variable llamada contador, cuyo valor se incrementa o decrementa en una cantidad constante en cada repetición que se produzca. El contador puede ser positivo (incrementos de uno en uno) o negativo (decrementos de uno en uno). La instrucción que actualiza al contador es la asignación.

```
CONT <- CONT + 1
```

Otro tipo de variable, normalmente asociada al funcionamiento de un bucle es un acumulador o totalizador, cuya misión es almacenar cantidad de variable resultante de operaciones sucesivas y repetidas. Un acumulador realiza una función parecida a la de un contador, con la diferencia de que el incremento o decremento de cada operación es variable en lugar de constante.

Una estructura repetitiva es aquella que marca la reiteración de una serie de acciones basándose en un bucle. De acuerdo con lo anterior, esta estructura debe constar de tres partes básicas.

- Decisión (para finalizar la repetición)

- Cuerpo del bucle (conjunto de instrucciones que se repiten)
- Salida del bucle (instrucción a la que se accede una vez se decide salir).

Una vez que se ha decidido el cuerpo del bucle, se plantea la cuestión de cuantas veces se debe repetir. De hecho conocemos ya la necesidad de contar con una condición para detener el bucle.

Estructura Desde – Hasta: Esta estructura consiste en que la condición de salida se basa en un contador que cuenta el número de intenciones, esta es

la llamada estructura *desde* (“*do*”) que se la utiliza desde el punto de vista de *Manual de: El algoritmo, una iniciación a la programación (Wilder Urbaez)* ligo. Su

utilidad reside en el hecho de que, en muchas ocasiones, se conoce de antemano, el número de interacciones. Esta estructura ejecuta las acciones del cuerpo del bucle, un número especificado de veces y de modo automático controla el número de interacciones. Su formato pseudocódigo es:

```
desde v = vi hasta vf hacer
    < acciones >
```

```
fin_desde
```

V: variables índice

Vi, vf: valores inicial y final de la variable

La variable índice o de control normalmente será de tipo entero y es normal emplear como identificador.

Estructura Mientras: Cuando la condición de salida del bucle se realiza al principio del mismo, éste se ejecuta mientras se verifica una cierta condición. Es la llamada estructura repetitiva *mientras* (“*while*”); en ella el cuerpo del bucle se repite mientras se cumple una determinada condición. Su pseudocódigo es.

```
mientras condición hacer
    < acciones >
```

```
fin_mientras
```

Cuando se ejecuta la instrucción *mientras*, la primera cosa que sucede es la evaluación de la condición. Si es *falsa*, no se ejecuta ninguna acción y el programa prosigue en la siguiente instrucción a la finalización del bucle; si la condición es *verdadera*, entonces se ejecuta el cuerpo del bucle. No todos los lenguajes incluyen la estructura *mientras*.

Estructura Repetir – Hasta _Que: En esta estructura la condición de salida se sitúa al final del bucle, se ejecuta hasta que se verifique una cierta condición. Es la llamada estructura *Repetir – hasta* (*repeat – until*). Existen muchas situaciones en las que se desea que un bucle se ejecute al menos una vez, antes de comprobar la condición de repetición. Para ello la estructura

repetir – hasta_que se ejecuta hasta que se cumpla una condición determinada que se comprueba al final del bucle. En pseudocódigo se escribe.

repetir

< acciones >

hasta_que < condición >

El bucle **repetir – hasta_que** se repita mientras la condición sea falsa, justo lo opuesto a la estructura **mientras**.

Operadores y operandos: Son elementos que relacionan de forma diferente, los valores de una o más variables y/o constantes. Es decir los operadores nos permiten manipular valores.

Operadores Aritméticos: Los operadores aritméticos permiten la realización de operaciones matemáticas con los valores (variables y constantes). Los operadores aritméticos pueden ser utilizados con tipos de datos enteros o reales. Si ambos son enteros, el resultado es entero; si alguno de ellos es real, el resultado es real. Entre ellos tenemos:

Suma (+)

Resta (-)

Multipliación (*)

División (/)

Exponenciación (^)

Mod

Div

Los operadores en una misma expresión con igual nivel de prioridad se evalúan de izquierda a derecha.

Operadores Relacionales: Se utilizan para establecer una relación entre dos valores. Luego compara estos valores entre si y esta comparación produce un resultado de certeza o falsedad (verdadero o falso). Los operadores relacionales compara valores del mismo tipo (numéricos o cadenas). Estos tienen el mismo nivel de prioridad en su evaluación. Los operadores relacionales tienen menor prioridad que los aritméticos. Estos son:

Mayor que (>)

Menor que (<)

Mayor o igual que (>=)

Menor o igual que (<=)

Diferente (<>)

Igual (=)

Operadores Lógicos: Estos operadores se utilizan para establecer relaciones entre valores lógicos. Estos valores pueden ser resultado de una expresión relacional. Estos son:

And (Y)

Or (O)

Not (Negación) : **Operador And:** Basta que una de las dos condiciones sea falsa para que respuesta sea falsa.

Operador Or: Basta que una de las condiciones sea verdadera para que su respuesta sea verdadera

O

Manual de: El algoritmo, una iniciación a la programación (Wilder Urbaez)

à falsa.

6.5. Objetivos

6.5.1. Objetivo General

Aplicar los conceptos en la práctica como recursos para fortalecer los conocimientos, de tal forma se va a mejorar las técnicas cuando se elaboren programas informáticos usando cualquier software de aplicación o diseñador de sistemas informáticos, de esta forma se va a elevar el autoestima del estudiante.

6.5.2. Objetivos Específicos

- Sensibilizar y comprometer la participación de los estudiantes realizando talleres para fortalecer su aprendizaje.
- Diseñar sistemas informáticos (conjunto pseudocódigos) mediante la aplicación de talleres realizando el intercambio de ideas a nivel de la colectividad estudiantil.
- Brindar los conocimientos necesarios de una forma clara y precisa relacionados con la programación y diseño de sistemas informáticos.

ENCUENTRO N° 1

TEMA: Aplicación de las estructuras repetitivas: Estructura **For (desde – hasta)**

Fecha	Tiempo	Actividades	Estrategias Responsables	Recursos	Evaluación	Responsable
Lunes 17/11/12	9:00 a 12:00	Ambientación Dinámica Presentar la temática Video conferencias Organizar al grupo realizando talleres con tres participantes Realizar el trabajo con el Orientador Elaborar talleres Identificar los grupos Conclusiones Recomendaciones Compromisos	Taller pedagógico	Humanos Rector Docentes Facilitador Investigador Materiales Videos Cd's Computador Infocus Pizarrón Marcadores Papel A4 Cinta	Elaboración de la memoria del evento realizado Exposiciones Lluvias de preguntas Debates Mesas redondas Preguntas y respuestas	Investigador Docente Facilitador

ENCUENTRO N° 2

TEMA: Aplicación de las estructuras repetitivas: Estructura **While (mientras)**

Fecha	Tiempo	Actividades	Estrategias Responsables	Recursos	Evaluación	Responsable
Viernes	9:00 a 12:00	Ambientación Dinámica Presentar la temática Entregar material de apoyo Video conferencias Organizar al grupo realizando talleres con 5 participantes Realizar el trabajo con el Orientador Elaborar talleres Identificar los grupos Elaborar papelotes Conclusiones Recomendaciones Compromisos	Taller pedagógico	Humanos Rector Docentes Facilitador Investigador Materiales Videos Cd's Computador Infocus Pizarrón Marcadores Papel A4 Cinta	Elaboración de la memoria del evento realizado Exposiciones Lluvias de preguntas Debates Mesas redondas Preguntas y respuestas	Investigador Docente Facilitador

ENCUENTRO N° 3

TEMA: Aplicación de las estructuras repetitivas: Estructura **Repeat - Until (repetir – hasta_que)**

Fecha	Tiempo	Actividades	Estrategias Responsables	Recursos	Evaluación	Responsable
Lunes	9:00 a 12:00	Ambientación Dinámica Presentar la temática Entregar material de apoyo Video conferencias Organizar al grupo realizando talleres con 5 participantes Realizar el trabajo con el Orientador Elaborar talleres Identificar los grupos Elaborar papelotes Conclusiones Recomendaciones Compromisos	Taller pedagógico	Humanos Rector Docentes Facilitador Investigador Materiales Videos Cd's Computador Infocus Pizarrón Marcadores Papel A4 Cinta	Elaboración de la memoria del evento realizado Exposiciones Lluvias de preguntas Debates Mesas redondas Preguntas y respuestas	Investigador Docente Facilitador

ENCUENTRO N° 4

TEMA: Aplicación de los operadores: Operadores Aritméticos, Operadores relacionales y Operadores Lógicos en la elaboración de programas informáticos.

Fecha	Tiempo	Actividades	Estrategias Responsables	Recursos	Evaluación	Responsable
Viernes	9:00 a 12:00	Ambientación Dinámica Presentar la temática Entregar material de apoyo Video conferencias Organizar al grupo realizando talleres con 5 participantes Realizar el trabajo con el Orientador Elaborar talleres Identificar los grupos Elaborar papelotes Conclusiones Recomendaciones Compromisos	Taller Pedagógico, teórico y práctico	Humanos Rector Docentes Facilitador Investigador Materiales Videos Cd's Computador Infocus Pizarrón Marcadores Papel A4 Cinta	Elaboración de la memoria del evento realizado Elaboración de ejercicios Exposiciones Lluvias de preguntas Debates Mesas redondas Preguntas y respuestas	Investigador Docente Facilitador

6.6. Descripción de los aspectos operativos de la propuesta

- Sesión de trabajo con el rector y docentes del Colegio
- Organizar cronograma de trabajo para la capacitación de los docentes
- Preparar materiales didácticos para la capacitación a los docentes.
- Desarrollar talleres prácticos, una cada semana.

Determinación del Lugar para la Capacitación: La guía para la aplicación de los conocimientos y el desarrollo de las Técnicas, en la aplicación de los operadores lógicos y de relación cuando se realicen los programas en las aulas de la institución.

Análisis Creativo antes de las Actividades: Determinar cuan importante es la aplicación de los operadores lógicos y de relación en el desarrollo de la lógica de programación

Relación del Trabajo Organizacional:

- Determinar cada una de las necesidades de los estudiantes
- Clasificación y aplicación de los operadores lógicos
- Elaboración del material de apoyo para la guía de los estudiantes

Actividades: Los Docentes realizarán las prácticas necesarias para luego compartirla en las aulas de clase.

Actividades Individuales: Dentro de estas actividades los estudiantes tienen la necesidad de familiarizarse con los siguientes aspectos.

- Tener conocimiento de cada una de las experiencias de los demás
- Colaborar con todas las actividades de diagnóstico a los estudiantes

Análisis Crítico: después de las Actividades: Evaluación individual y colectiva sobre cada uno de los conceptos que se han analizado en las horas de clase

Métodos y Técnicas a Utilizarse

- Exposiciones
- Evaluaciones
- Lluvia de Preguntas

- Mesas Redondas
- Debates
- Preguntas Lógicas
- Aplicación de Ejercicios

6.7. RECURSOS

Humanos

- Docentes
- Estudiantes
- Padres de Familia
- Autoridades del Plantel

Materiales

- Computadora
- Impresora
- Proyector
- Sala de clases
- Pizarrón
- Tiza Líquida
- Hojas A-4

Técnicos (Funciones de los Capacitadores): Los Docentes distribuirán el material a cada uno de los grupos de trabajo que se encuentren participando en la capacitación.

Desarrollo de cada una de las Actividades en la Capacitación.

Responsables

- Investigador
- Director de Tesis
- Lector de Tesis

Financiamiento

- Autogestión

Cronograma de Ejecución de la propuesta

Nº	Actividades	Meses y Sem. Activ.	SEP.				OCT.				NOV.				DIC.				ENER.			
			1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1.	Selección de Tema		X	X																		
2.	Aprobación del tema				X	X																
3.	Recopilación de la Información						X															
4.	Desarrollo del I capítulo							X	X	X												
5.	Desarrollo del II capítulo								X	X	X											
6.	Desarrollo del III capítulo										X	X										
7.	Elaboración de las encuesta											X	X									
8.	Aplicación de las encuestas												X	X								
9.	Tamización de la información													X	X							
10	Desarrollo de IV Capítulo														X	X						
11	Elaboración de Conclusión y Recomendación															X	X					
12	Desarrollo del V Capítulo																X	X				
13	Desarrollo del VI Capítulo																	X	X	X		
14	Presentación de la Tesis																		X	X	X	

BIBLIOGRAFIA

Bibliografía de Textos

- ❖ Manual de: El algoritmo, una iniciación a la programación (Wilder Urbaez).
- ❖ Manual de: Diseño de una metodología Ágil de Desarrollo de Software (Schenome Marcelo Hernán).
- ❖ Manual de: Algoritmos y Programas, Fundamentos de Informática y Programación.
- ❖ Revista de: Programación Lógica; Un recorrido por la programación lógica y uno de sus lenguajes más representativos (Gerardo Rossel investigador del CAETI).
- ❖ Revista de: Metodología para la enseñanza aprendizaje de la lógica de la programación orientada a objetos (Leobardo López Román).
- ❖ Revista de: Programación Lógica: Un enfoque para desarrollar aplicaciones (Fernando Alonso Amo – Marco Villalobos Abarca).
- ❖ Revista de: Metodología para la construcción de Programas (Prof. Dolores Cuiñas).

FAKES



UNIVERSIDAD TÉCNICA DE BABAHOYO
FACULTAD DE CIENCIAS DE LA EDUCACION

COMPUTACION

LA PRESENTE INVESTIGACION TIENE COMO FINALIDAD REALIZAR UN ESTUDIO ACERCA DEL DESARROLLO DE LA LÓGICA EN LOS ESTUDIANTES DEL PRIMER AÑO DE BACHILLERATO PARALELO "A" AL REALIZAR UN PROGRAMA INFORMÁTICO EN CUALQUIER LENGUAJE DE PROGRAMACIÓN, DEL COLEGIO NICOLÁS INFANTE DÍAZ DE LA CIUDAD DE QUEVEDO PROVINCIA DE LOS RIOS EN EL AÑO 2012.

1. ¿SE ENCUENTRA EQUIPADO EL LABORATORIO CON LAS NECESIDADES DEL ESTUDIANTE PARA REALIZAR PROGRAMAS DE APLICACION?

SI

NO

2. ¿LE ENTIENDES AL MAESTRO CUANDO ELABORA PROGRAMAS INFORMATICOS?

MUCHO

POCO

NADA

3. ¿REALIZAN TALLERES DE ESTUDIO CON EL MAESTRO PARA DESARROLLAR LA LOGICA EN LA PROGAMACION?

SI

NO

4. ¿FUNCIONAN TODOS LOS PROGRAMAS INFORMATICOS DADOS EN CLASE?

SIEMPRE

A VECES

NUNCA

5. CUANDO NO ENTIENDES UN PROGRAMA, ¿REPITE LA CLASE EL MAESTRO?

SI NO A VECES

6. ¿TE SIENTES CAPAZ DE COMPETIR CON OTRO COLEGIO REALIZANDO PROGRAMAS INFORMATICOS?

MUCHO POCO NADA

7. ¿ESTAS CAPACITADO PARA ENSEÑAR PROGRAMAS INFORMATICOS A OTRAS PERSONAS?

MUCHO POCO NADA

8. ¿TE SIENTES CAPAZ DE REALIZAR UN SISTEMA INFORMATICO?

SI NO

9. TIENES COMPUTADORA EN CASA PARA REALIZAR LOS PROGRAMAS INFORMATICOS ?

SI NO

10. ¿TE GUSTA EÑABORAR PROGRAMAS INFORMATICOS?

MUCHO POCO NADA

11. ¿CUANDO TIENES UNA INQUIETUD EN CUANTO A UN PROGRAMA INFORMATICO, LE PREGUNTAS A TU MAESTRO ¿COMO TE RESPONDE?

CORTES

DESCORTES

PACIENTE

IMPACIENTE

AMABLE

DESPOTA



UNIVERSIDAD TÉCNICA DE BABAHOYO
FACULTAD DE CIENCIAS DE LA EDUCACION

COMPUTACION

LA PRESENTE INVESTIGACION TIENE COMO FINALIDAD REALIZAR UN ESTUDIO ACERCA DE LAS ESTRATEGIAS QUE UTILIZA EL MAESTRO PARA QUE EL ALUMNO DESARROLLE LA LOGICA EN LA ELABORACION DE PROGRAMAS INFORMATICOS, DEL COLEGIO NICOLÁS INFANTE DÍAZ DE LA CIUDAD DE QUEVEDO PROVINCIA DE LOS RIOS EN EL AÑO 2012

1. ¿LOS ALUMNOS CUMPLEN A CABALIDAD CON LAS TAREAS?
SI NO
2. ¿CADA QUE TIEMPO LOS PADRES DE FAMILIA SE ACERCAN A PREGUNTAR POR LAS NOTAS DE SU REPRESENTADO?
DIARIO SEMANAL MENSUAL NUNCA
3. ¿EL COLEGIO LE BRINDA LOS RECURSOS NECESARIOS PARA DESARROLLAR SUS CLASES?
SI NO
4. ¿LE BRINDA LA OPORTUNIDAD DE RECUPERACION A LOS ALUMNOS?
SI NO
5. ¿DA SEMINARIOS DE CAPACITACION A LOS ESTUDIANTES?
SI NO

6. ¿REALIZA EJERCICIOS DE ALGORITMO EN CLASES?

SI

NO

7. ¿CADA CUANTO TIEMPO EVALUA A LOS ESTUDIANTES PARA COMPROBAR SU APRENDIZAJE EN CUANTO A LA TECNICAS DE PROGRAMACION?

DIARIO

SEMANAL

MENSUAL

TRIMESTRAL

NUNCA

8. ¿INSENTIVA A LOS ESTUDIANTES A PARTICIPAR EN CLASES UTILIZANDO LA LOGICA EN LA ELBORACION DE PROGRAMAS INFORMATICOS?

SI

NO

9. EN EL AULA, ¿EXISTEN ESTUDIANTES QUE DEMUESTRAN INTERES POR APRENDER A DESARROLLAR LAS TECNICAS DE PROGRAMACION?

SI

NO

10. ¿ESTA FAMILIARIZADO CON LOS DIFERENTES LENGUAJES DE PROGRAMACION?

SI

NO



UNIVERSIDAD TÉCNICA DE BABAHOYO
FACULTAD DE CIENCIAS DE LA EDUCACION

COMPUTACION

LA PRESENTE INVESTIGACION TIENE COMO FINALIDAD REALIZAR UN ESTUDIO A LOS PADRES DE FAMILIA PARA SABER COMO ES EL APRENDIZAJE DE SU REPRESENTADO, LAS ESTRATEGIAS QUE UTILIZA EL DOCENTE PARA QUE EL ALUMNO DESARROLLE LA LOGICA EN LA ELABORACION DE PROGRAMAS INFORMATICOS, DEL COLEGIO NICOLÁS INFANTE DÍAZ DE LA CIUDAD DE QUEVEDO PROVINCIA DE LOS RIOS EN EL AÑO 2012.

1. ¿CONOCE UD. SI EL DOCENTE REALIZA EJERCICIOS LÓGICOS EN SU CLASE?

SI NO A VECES

2. ¿UD. CREE QUE LOS EJERCICIOS REALIZADOS EN CLASE ESTAN ACORDE CON EL TEMA TRATADO?

SI NO A VECES

3. ¿LA ELABORACIÓN DE EJERCICIOS LOGICOS EN CLASE MOTIVA LA PARTICIPACION DE SU HIJO?

SI NO A VECES

4. ¿LA ELABORACIÓN DE EJERCICIOS EN CLASE FACILITA EL APRENDIZAJE DE SU HIJO?

SI NO A VECES

5. CON LA APLICACIÓN DE EJERCICIOS LÓGICOS EN CLASE, ¿SU HIJO PARTICIPA ACTIVAMENTE?

SI NO A VECES

6. SU HIJO, ¿REALIZA LOS EJERCICIOS LÓGICOS EN CASA SIN NINGÚN PROBLEMA?

SI NO A VECES

7. SU HIJO, ¿AL ELABORAR UN EJERCICIO LÓGICO PIDE AYUDA A OTRA PERSONA?

SI NO A VECES

8. SU HIJO, ¿SE ADAPTA A LAS ENSEÑANZAS DEL MAESTRO CUANDO IMPARTEN SUS CONOCIMIENTOS?

SI NO A VECES

Juan Carlos Loor Robles

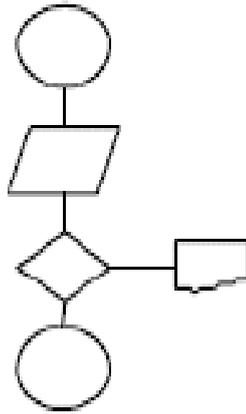


Figura 1: Algoritmo gráfico

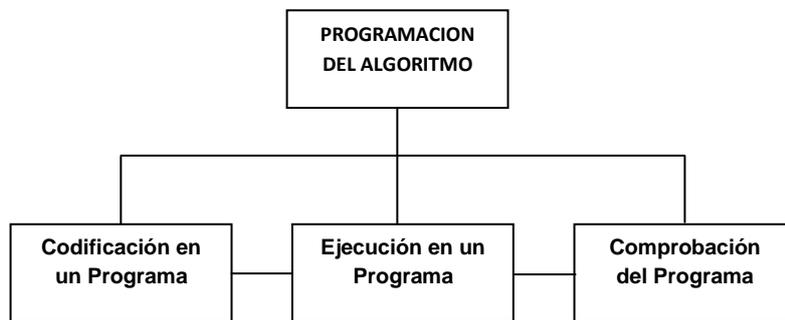


Figura 2: Programación del algoritmo

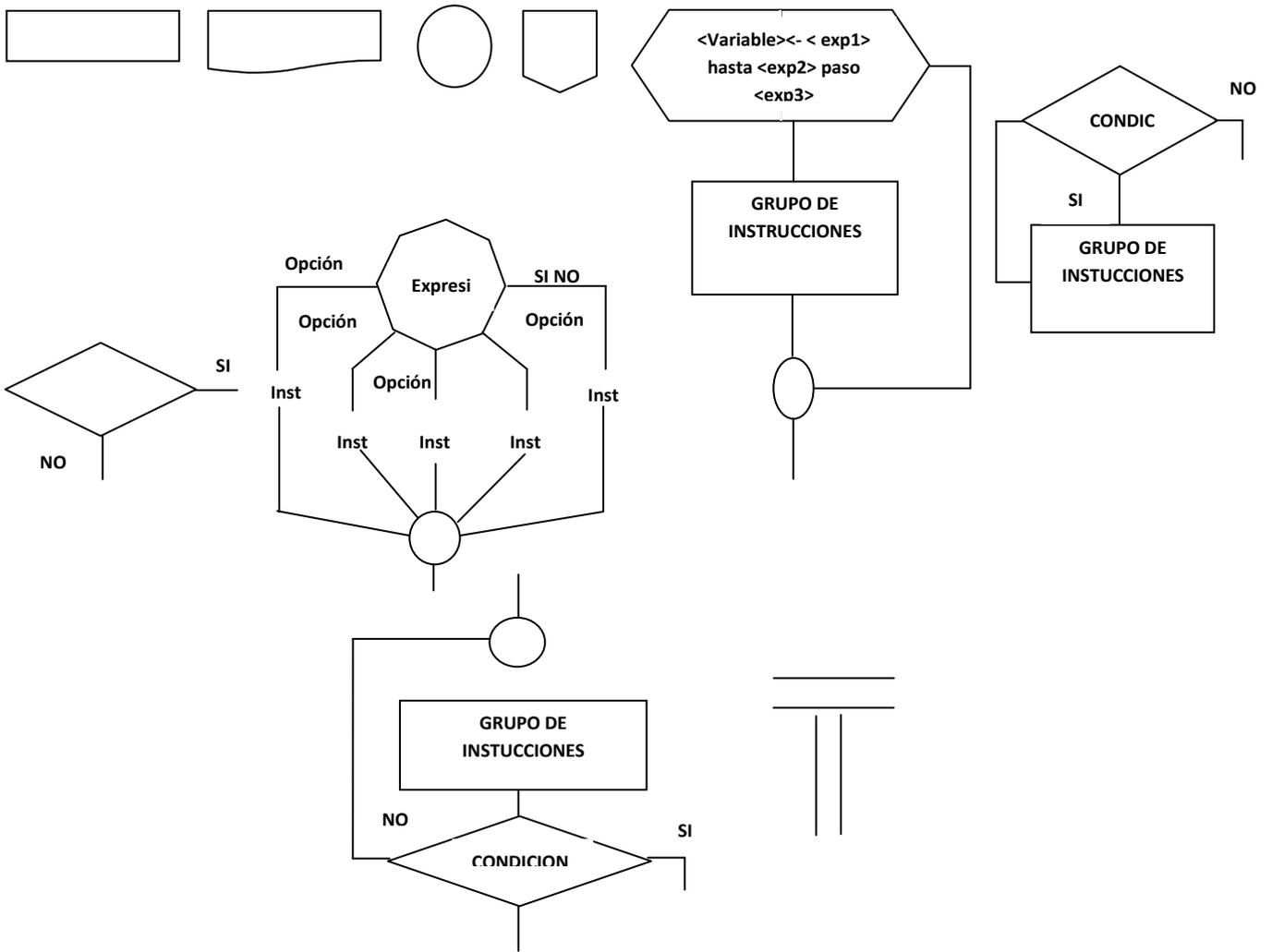


Figura 3: Gráficos para la elaboración de algoritmos

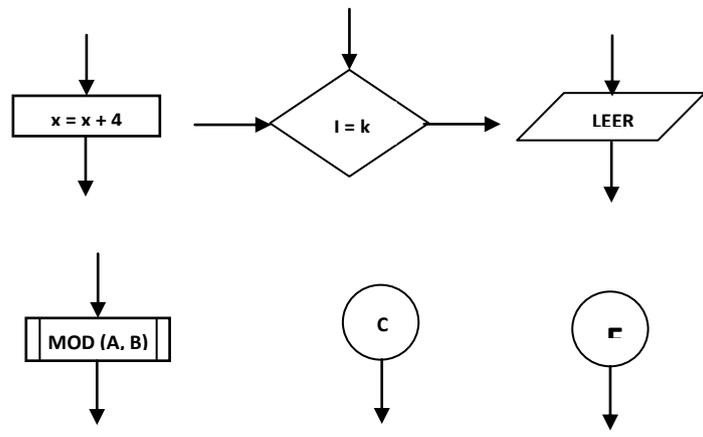


Figura 4: Gráficos que presentan a los algoritmos

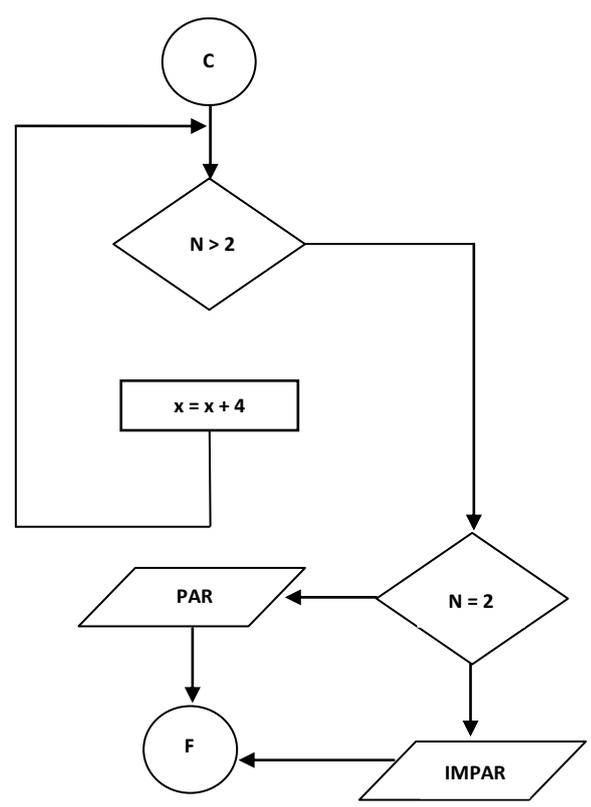


Figura 5: Organigrama del ejercicio

Cuadro 1. Operacionalización de la primera hipótesis Independiente

CONCEPTOS	CATEGORIA	INDICADORES	ÍTEMS BÁSICOS	TECNICAS E INSTRUMENTOS
<p>La elaboración de programas informáticos es el Conjunto de conocimientos científicos y técnicos que se ocupa del tratamiento de la información por medios automáticos, principalmente mediante equipos electrónicos de procesos de datos que Ayudan en su coeficiente intelectual y en el comportamiento de una sociedad que se encuentra en constante progreso, en los diferentes campos socioeconómicos.</p>	<p>Si aplicamos la elaboración de programas informáticos en las practicas, talleres para el fortalecimiento de las tareas</p>	<ul style="list-style-type: none"> -Aplicación -Talleres -Conocimientos -Intelectos 	<ul style="list-style-type: none"> -¿Los docentes realizan talleres prácticos durante las clases? -¿Los programas que se aplican son previamente escogidos para actividades concretas? -¿Les brindan la atención necesaria a los programas informáticos en clases? 	<p>Entrevista a autoridades y docentes y encuestas a los padres de familia.</p>

Cuadro 2. Operacionalización de la segunda hipótesis Dependiente

CONCEPTOS	CATEGORIA	INDICADORES	ÍTEMS BÁSICOS	TECNICAS E INSTRUMENTOS
<p>Aprendizaje.- Se denomina aprendizaje al proceso de adquisición de conocimientos, habilidades, valores y actitudes, posibilitado mediante el estudio, la enseñanza o la experiencia. Dicho proceso puede ser entendido a partir de diversas posturas, lo que implica que existen diferentes teorías vinculadas al hecho de aprender. La psicología conductista, por ejemplo, describe el aprendizaje de acuerdo a los cambios que pueden observarse en la conducta de un sujeto.</p>	<p>Como influye en el aprendizaje de los alumnos del primer año paralelo "A" de Bachillerato del Colegio "Nicolás Infante Díaz"</p>	<p>Participación -Dedicación -Realización de tareas -Presentación de tareas</p>	<p>¿La utilización de los programas informáticos a más en clases ayuda a su coeficiente intelectual?</p>	<p>Entrevista a docentes y encuestas a los padres de familia y alumnos.</p>

Cuadro 3: Población del primer año de bachillerato

	Población	Muestra	Mujeres	Hombres
Maestros	8	8	4	4
Autoridades	2	2	1	1
Padre de Familia	20	20	16	4
Estudiantes	45	45	26	19

Cuadro 4: Población, Frecuencia y Porcentaje del primer año de bachillerato

POBLACION	FRECUENCIA	PORCENTAJE
Docentes	10	13
Estudiantes	45	60
Padres de Familia	20	27
Total	75	100 %

Cuadro 5: Gastos Realizados

CONCEPTO	VALOR UNITARIO	VALOR / SUB TOTAL
Una persona para realizar las encuestas y la tabulación	\$ 10,00 C/día	\$ 10,00
Un Director de Tesis	—	—
Un Lector de Tesis	—	—
Un Computador	\$ 1,00 C/hora	\$ 42,00
Viáticos y Subsistencia	\$ 5,00 C/Día	\$ 40,00
Material Logístico	\$ 50,00	\$ 200,00
Elaboración de 75 encuestas	\$ 0,05 C/una	\$ 7,50
Elaboración de la Tesis		\$ 200,00
TOTAL		\$ 499,50

ENCUENTRO Nº 1**TEMA: Aplicación de las estructuras repetitivas: Estructura For (desde – hasta)**

Fecha	Tiempo	Actividades	Estrategias Responsables	Recursos	Evaluación	Responsable
Lunes 05/11/12	9:00 a 12:00	Ambientación Dinámica Presentar la temática Video conferencias Organizar al grupo realizando talleres con tres participantes Realizar el trabajo con el Orientador Elaborar talleres Identificar los grupos Conclusiones Recomendaciones Compromisos	Taller pedagógico	Humanos Rector Docentes Facilitador Investigador Materiales Videos Cd's Computador Infocus Pizarrón Marcadores Papel A4 Cinta	Elaboración de la memoria del evento realizado Exposiciones Lluvias de preguntas Debates Mesas redondas Preguntas y respuestas	Investigador Docente Facilitador

ENCUENTRO Nº 2**TEMA: Aplicación de las estructuras repetitivas: Estructura While (mientras)**

Fecha	Tiempo	Actividades	Estrategias Responsables	Recursos	Evaluación	Responsable
Viernes 09/11/12	9:00 a 12:00	Ambientación Dinámica Presentar la temática Entregar material de apoyo Video conferencias Organizar al grupo realizando talleres con 5 participantes Realizar el trabajo con el Orientador Elaborar talleres Identificar los grupos Elaborar papelotes Conclusiones Recomendaciones Compromisos	Taller pedagógico	Humanos Rector Docentes Facilitador Investigador Materiales Videos Cd's Computador Infocus Pizarrón Marcadores Papel A4 Cinta	Elaboración de la memoria del evento realizado Exposiciones Lluvias de preguntas Debates Mesas redondas Preguntas y respuestas	Investigador Docente Facilitador

ENCUENTRO Nº 3

TEMA: Aplicación de las estructuras repetitivas: Estructura **Repeat - Until (repetir – hasta_que)**

Fecha	Tiempo	Actividades	Estrategias Responsables	Recursos	Evaluación	Responsable
Lunes 12/11/12	9:00 a 12:00	Ambientación Dinámica Presentar la temática Entregar material de apoyo Video conferencias Organizar al grupo realizando talleres con 5 participantes Realizar el trabajo con el Orientador Elaborar talleres Identificar los grupos Elaborar papelotes Conclusiones Recomendaciones Compromisos	Taller pedagógico	Humanos Rector Docentes Facilitador Investigador Materiales Videos Cd's Computador Infocus Pizarrón Marcadores Papel A4 Cinta	Elaboración de la memoria del evento realizado Exposiciones Lluvias de preguntas Debates Mesas redondas Preguntas y respuestas	Investigador Docente Facilitador

ENCUENTRO Nº 4

TEMA: Aplicación de los operadores: Operadores Aritméticos, Operadores relacionales y Operadores Lógicos en la elaboración de programas informáticos.

Fecha	Tiempo	Actividades	Estrategias Responsables	Recursos	Evaluación	Responsable
Viernes 23/11/12	9:00 a 12:00	Ambientación Dinámica Presentar la temática Entregar material de apoyo Video conferencias Organizar al grupo realizando talleres con 5 participantes Realizar el trabajo con el Orientador Elaborar talleres Identificar los grupos Elaborar papelotes Conclusiones Recomendaciones Compromisos	Taller Pedagógico, teórico y práctico	Humanos Rector Docentes Facilitador Investigador Materiales Videos Cd's Computador Infocus Pizarrón Marcadores Papel A4 Cinta	Elaboración de la memoria del evento realizado Elaboración de ejercicios Exposiciones Lluvias de preguntas Debates Mesas redondas Preguntas y respuestas	Investigador Docente Facilitador