



TECNOLÓGICO  
NACIONAL DE MÉXICO



Instituto Tecnológico Superior del Occidente del Estado de Hidalgo  
Ingeniería en Tecnologías de la Información y Comunicaciones

## *Proyecto Integrador*

Software de resolución de problemas de ingeniería

### **Coordinador del proyecto:**

Dr. Francisco Javier Cuadros Romero

### **Integrantes del equipo:**

1<sup>er</sup> Cedillo Ramírez Jesús Emmanuel  
*Estudiante de ITIC 's*  
*ITSOEH*

Progreso de Obregón, HGO.  
230110478@itsoeh.edu.mx

2<sup>do</sup> Gómez Ángeles Evelin  
*Estudiante de ITIC 's*  
*ITSOEH*

Tezontepec de Aldama, HGO.  
230110617@itsoeh.edu.mx

3<sup>er</sup> Gonzaga López Luis Fernando  
*Estudiante de ITIC 's*  
*ITSOEH*

Tlahuelilpan, HGO.  
230110528@itsoeh.edu.mx

4<sup>to</sup> González Calva Yael  
*Estudiante de ITIC 's*  
*ITSOEH*

Progreso de Obregón, HGO.  
230110582@itsoeh.edu.mx

5<sup>to</sup> Jarquín Contreras Kevin Ichiro  
*Estudiante de ITIC 's*  
*ITSOEH*

Mixquiahuala, HGO.  
230110949@itsoeh.edu.mx

6<sup>to</sup> López Paz Gustavo  
*Estudiante de ITIC 's*  
*ITSOEH*

Progreso de Obregón, HGO.  
230110531@itsoeh.edu.mx

## ÍNDICE

<b>I.</b>	<b>Introducción</b>	4
<b>II.</b>	<b>Objetivos</b>	4
II-A.	Objetivo General . . . . .	4
II-B.	Objetivos específicos . . . . .	4
<b>III.</b>	<b>Problemas</b>	4
III-A.	Ecuación de la recta . . . . .	4
III-A1.	Definición del problema . . . . .	4
III-A2.	Descripción del problema . . . . .	4
III-A3.	Diseño de solución . . . . .	4
III-A4.	Desarrollo de solución . . . . .	5
III-A5.	Depuración y pruebas . . . . .	5
III-B.	Raíces de una ecuación cuadrática . . . . .	5
III-B1.	Definición del problema . . . . .	5
III-B2.	Descripción del Problema . . . . .	6
III-B3.	Definición de la solución . . . . .	6
III-B4.	Diseño de la solución . . . . .	6
III-B5.	Desarrollo de la solución . . . . .	6
III-B6.	Depuración y pruebas . . . . .	7
III-C.	Punto dada una circunferencia . . . . .	7
III-C1.	Definición del problema . . . . .	7
III-C2.	Descripción del problema . . . . .	7
III-C3.	Diseño de solución . . . . .	7
III-C4.	Desarrollo de la solución . . . . .	7
III-C5.	Depuración y pruebas . . . . .	8
III-D.	Conversión decimal a binario . . . . .	8
III-D1.	Definición del problema . . . . .	8
III-D2.	Descripción del Problema . . . . .	8
III-D3.	Diseño de solución . . . . .	8
III-D4.	Desarrollo de la solución . . . . .	8
III-D5.	Depuración y pruebas . . . . .	8
III-E.	Conversión binario a decimal . . . . .	8
III-E1.	Definición del problema . . . . .	8
III-E2.	Descripción del problema . . . . .	9
III-E3.	Diseño de solución . . . . .	9
III-E4.	Desarrollo de la solución del problema . . . . .	9
III-E5.	Depuración y pruebas . . . . .	9
III-F.	Tabla de verdad usando el teorema booleano . . . . .	9
III-F1.	Definición del problema . . . . .	9
III-F2.	Descripción del problema . . . . .	9
III-F3.	Diseño de solución . . . . .	10
III-F4.	Desarrollo de solución . . . . .	10
III-F5.	Depuración y pruebas . . . . .	10
<b>IV.</b>	<b>Conclusión</b>	11
<b>V.</b>	<b>Agradecimientos</b>	11
	<b>Referencias</b>	11

<b>VI.</b>	<b>Integrantes</b>	11
VI-A.	<b>Cedillo Ramírez Jesús Emmanuel</b>	11
VI-B.	<b>Gómez Ángeles Evelin</b>	11
VI-C.	<b>Gonzaga López Luis Fernando</b>	11
VI-D.	<b>González Calva Yael</b>	11
VI-E.	<b>Jarquín Contreras Kevin Ichiro</b>	12
VI-F.	<b>López Paz Gustavo</b>	12

**Resumen**—Pasado el tiempo desde inicio de semestre Agosto-Diciembre 2023, se han abarcado diferentes temas aprendidos con diferentes metodologías correspondientes a cada uno de los docentes de las siete materias que contiene el primer semestre de ingeniería en Tecnologías de la Información y Comunicaciones, dentro de estos temas se escogieron tres problemas de Cálculo Diferencial y tres problemas de Matemáticas Discretas, desarrollándolos con conocimientos adquiridos en Fundamentos de Programación, y subiendo la solución de estos mismos en un repositorio de GitHub, y haciendo el siguiente informe con el proceso de solución de cada uno de los ejercicios, resueltos con uso de la metodología de las 6D's.

**Palabras clave:**—Matemáticas, conocimientos, problemas, solución, metodología.

## I. INTRODUCCIÓN

En el siguiente documento se presentan seis problemas de primer semestre de ingeniería: el primero de ellos es encontrar la ecuación de la recta dados dos puntos en el plano cartesiano  $(x_1, y_1)$  y  $(x_2, y_2)$ , obteniendo la pendiente  $m$  que se define como  $m = (y_2 - y_1)/(x_2 - x_1)$  para arrojar el resultado de la ecuación de la recta de la forma  $y = mx + b$  donde  $b$  es el punto de intersección con el eje  $y$ ; el segundo problema es encontrar el valor de las raíces de una ecuación cuadrática o ecuación de segundo grado de la forma  $ax^2 + bx + c = 0$ , por medio de la fórmula general que se define como  $x = (-b \pm \sqrt{b^2 - 4ac})/(2a)$  donde  $a, b, c$  son los coeficientes que acompañan a la ecuación cuadrática; el tercer problema es arrojar si un punto dado del plano cartesiano está dentro o fuera de una circunferencia con centro fuera del origen comparando el valor del radio de la circunferencia y la distancia que hay entre el punto dado con el punto que marca el centro de la circunferencia, esta distancia la encontraremos con la fórmula de distancia entre dos puntos en el plano cartesiano que se define como  $d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$ , y así de esta manera resolver este problema; el cuarto problema es encontrar el número binario correspondiente a un número decimal dado por el usuario el cual tiene un método de división sucesiva para encontrar el binario del decimal; el quinto problema es inverso al problema anterior el cual consiste en encontrar el valor en numeración decimal de un número binario dado por el usuario, usando un método de potenciación de base dos y exponente con valor de posición del dígito del número binario comenzando de derecha a izquierda, por muy similar que se noten estos dos últimos problemas tienen una metodología diferente para su solución; el sexto y último problema consiste en encontrar una expresión booleana que genere las salidas en una tabla de  $n$  bits, donde  $n$  será el valor dado por el usuario.

Estos seis problemas mencionados brevemente fueron analizados y resueltos con una metodología en común que es la metodología de las 6D's: Definición del problema, donde mencionamos información acerca del tema que contiene el problema; Descripción del problema, donde solo hacemos mención del problema a resolver; Diseño de solución, donde se hace un análisis de como resolver el problema y dar la

matemática que contiene cada uno de estos; Desarrollo de solución, donde se da a detalle y por paso la solución de estos agregando el código hecho en lenguaje de programación JAVA; Depuración y pruebas, donde se agregan tablas de corrida del código con datos dados por el usuario y el dato arrojado por el programa; y Documentación, la cual consiste en el siguiente informe.

## II. OBJETIVOS

### II-A. Objetivo General

- Desarrollar un software en lenguaje JAVA para la solución útil y precisa de problemas matemáticos y de ingeniería, que permita a los usuarios realizar cálculos y análisis numéricos de manera efectiva.

### II-B. Objetivos específicos

1. Comprender la ecuación de la recta, la fórmula general y la distancia entre dos puntos, para resolver problemas matemáticos relacionados con el álgebra y geometría, y aplicados a la ingeniería.
2. Comprender y aplicar los métodos de conversión de numeración binaria a numeración decimal, y viceversa.
3. Aplicar los conceptos de tablas de verdad para analizar expresiones lógicas.

## III. PROBLEMAS

### III-A. Ecuación de la recta

#### III-A1. Definición del problema:

Se trata de una noción de la geometría que refiere a la línea unidimensional que, formada por una cantidad infinita de puntos, se prolonga en una misma dirección.

1. Una línea recta es el lugar geométrico en un plano formado por una sucesión de puntos que tienen la misma dirección. Dados dos puntos diferentes, sólo una recta pasa por esos dos puntos.
2. Es el lugar geométrico de los puntos de un plano, de los cuales al tomar dos cualesquiera, el valor de la pendiente  $m$ , es siempre constante.
3. Es el lugar geométrico formado por un polinomio de primer grado de la forma  $y = mx + b$ .
4. Es el lugar geométrico obtenido al unir dos puntos, tal que la distancia recorrida, es la más corta posible.

#### III-A2. Descripción del problema:

Dados 2 puntos A y B con coordenadas  $(x_1, y_1)$  y  $(x_2, y_2)$  respectivamente. Regresar la ecuación de la recta y el ángulo interno  $\alpha$  que se forma entre el eje horizontal y la recta.

#### III-A3. Diseño de solución:

En la ecuación de la recta si dos puntos distintos  $P(x_1, y_1)$  y  $Q(x_2, y_2)$  se ubican en la curva  $y = f(x)$ , la pendiente de la recta secante que une los dos puntos es:

$$m_{secc} = \frac{y_1 - y_2}{x_1 - x_2} = \frac{f(x_1) - f(x_2)}{x_1 - x_2} \quad (1)$$

Para identificar la intersección en el eje vertical se utiliza cualquiera de los dos puntos para este caso se utilizó  $P(x_1, y_1)$  de la siguiente forma:

$$y = mx + b \quad (2)$$

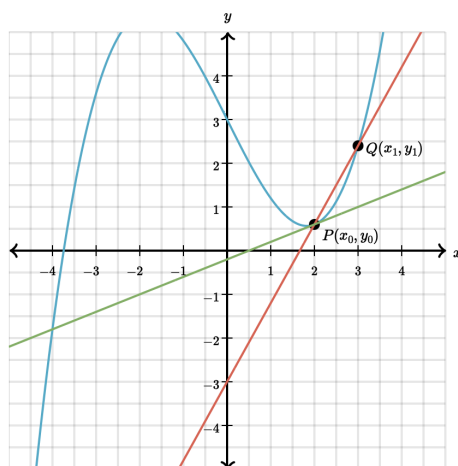


Figura 1: Gráfica de la ecuación de la recta

Utilizando este método, puedes encontrar la ecuación de la recta a partir de dos puntos dados. Recordando que si dos puntos son idénticos, la recta será una línea vertical [1]. Y para calcular el ángulo de dicha pendiente se usa:

$$\angle = \arctan(m) \quad (3)$$

#### III-A4. Desarrollo de solución:

1. El algoritmo de solución implementado en el lenguaje JAVA empieza por la entrada de los valores  $P(x_1, y_1)$  y  $Q(x_2, y_2)$  solicitando que sean separados por una coma.

```

1 Scanner punto = new Scanner(System.in);
2
3 //solicitar puntos
4 System.out.println(
5     ""
6     Ingrese las coordenadas (x,y) separadas
7     por una coma(,) del punto 1:
8     "");
9 String [] puntoUno =
10     punto.nextLine().split(",");
11
12 System.out.println(
13     ""
14     Ingrese las coordenadas (x,y) separadas
15     por una coma(,) del punto 2: """);
16 String [] puntoDos =
17     punto.nextLine().split(",");
18
19 punto.close();

```

2. Después de ingresar los valores, se obtienen los valores ingresados para después convertirlos de valores de cadena a valores enteros, para esto se aplicó el siguiente método de conversión.

```

1 String[] coordenadasArrayPunto1 =
2     coordenadasPunto1.split(",");
3 String[] coordenadasArrayPunto2 =
4     coordenadasPunto2.split(",");
5
6 //obtener coordenadas X y Y
7 int coordenadaX1 =
8     Integer.parseInt(coordenadasArrayPunto1[0].
9     trim());
10
11 int coordenadaY1 =
12     Integer.parseInt(coordenadasArrayPunto1[1].
13     trim());
14
15 int coordenadaX2 =
16     Integer.parseInt(coordenadasArrayPunto2[0].
17     trim());
18
19 int coordenadaY2 =
20     Integer.parseInt(coordenadasArrayPunto2[1].
21     trim());

```

```

7 int coordenadaY1 =
8     Integer.parseInt(coordenadasArrayPunto1[1].
9     trim());
10
11 int coordenadaX2 =
12     Integer.parseInt(coordenadasArrayPunto2[0].
13     trim());
14
15 int coordenadaY2 =
16     Integer.parseInt(coordenadasArrayPunto2[1].
17     trim());
18
19 System.out.println("Coordenada X1: " +
20     coordenadaX1);
21 System.out.println("Coordenada Y1: " +
22     coordenadaY1);
23
24 System.out.println("Coordenada X2: " +
25     coordenadaX2);
26 System.out.println("Coordenada Y2: " +
27     coordenadaY2);

```

3. Luego se calcula la pendiente  $m$  utilizando la fórmula de la pendiente [1].

```

1 double m = (coordenadaY2 -
2     coordenadaY1) / (coordenadaX2 -
3     coordenadaX1);

```

4. A continuación se calcula el término  $b$  de la fórmula de intersección.

```

1 double b = coordenadaY1 - m *
2     coordenadaX1;

```

5. Por último calculamos el ángulo interno que se forma entre el eje horizontal y la recta pendiente.

```

1 double anguloRadianes = Math.atan(m);
2 double anguloGrados = anguloRadianes *
3     (180/Math.PI);

```

6. Obteniendo finalmente el resultado de la intersección de la recta y su ángulo interno que se formó.

#### III-A5. Depuración y pruebas:

En la tabla 1 se muestran los resultados obtenidos al compilar el código.

$x_1$	$x_2$	$y_1$	$y_2$	Pendiente	Inclinación	Intersección
2	4	4	6	1	45°	-2
3	2	4	1	3	71.57°	-5
2	2	2	2	0	0°	0
1	3	2	4	1	45°	1

Tabla I: Tabla de corridas.

#### III-B. Raíces de una ecuación cuadrática

##### III-B1. Definición del problema:

La ecuación cuadrática o fórmula cuadrática es utilizada para encontrar las raíces de una ecuación cuadrática de la forma  $ax^2 + bx + c = 0$ , la fórmula se compone por 3 parámetros:

1. Parámetro  $a$ : Representa la posición del vértice de la parábola en el eje  $Y$ .

2. Parámetro  $b$ : Representa la posición del vértice en la parábola del eje  $X$ .
3. Parámetro  $c$ : Representa el punto de intersección de la parábola con el eje  $Y$ .

El uso de esta fórmula es útil cuando se requiere resolver alguna ecuación cuadrática que ya se intentó resolver por distintos métodos o por medio de la factorización, por ello la fórmula cuadrática es la siguiente:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (4)$$

En esta fórmula se toma en cuenta el discriminante que es representado en la fórmula por  $b^2 - 4ac$ , el cual determina la naturaleza y el número de soluciones de la ecuación, analizando las propiedades de una ecuación cuadrática y determinar su comportamiento de soluciones reales. [2]

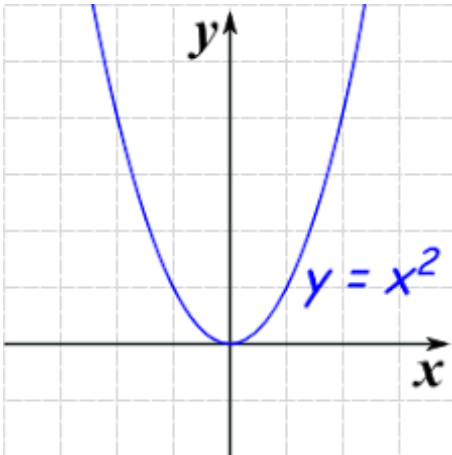


Figura 2: Gráfica de la fórmula cuadrática

### III-B2. Descripción del Problema:

Se solicita al usuario que ingrese los tres parámetros  $a$ ,  $b$ , y  $c$ , donde estos se evaluarán en un discriminante el cual es  $b^2 - 4ac$ , y que al final dependiendo de este se resuelva el resto de la operación de esta fórmula.

### III-B3. Definición de la solución:

Para tener solución a este problema se consideran 3 posibles respuestas que dependen del discriminante, las cuales son:

1. La ecuación tiene dos soluciones sobre el conjunto de los números reales,  $x_1$  y  $x_2$ .
2. La ecuación solo tiene una solución en el conjunto de los números reales,  $x$ .
3. La ecuación no tiene una solución en el conjunto de los números reales, esta se encuentra en el conjunto de los números complejos.

### III-B4. Diseño de la solución:

Para resolver este problema se requiere que el programa reciba 3 coeficientes de tres parámetros que vendrán de la ecuación cuadrática que se tenga, donde primero se establece el desarrollo del discriminante y por consiguiente la evaluación del discriminante donde determinará si existen dos

soluciones, una solución o solución en el conjunto de los complejos.

### III-B5. Desarrollo de la solución:

1. La solución a este problema se diseñó un programa el cual se tendrá que ingresar los 3 coeficientes de los parámetros denominados de  $a$ ,  $b$ , y  $c$  para resolver la fórmula cuadrática.

```

1      System.out.println("Ingrese el
2      parametro de a de la ecuación
3      cuadrática");
4      double a = in.nextDouble();
5      System.out.println("Ingrese el
6      parametro de b de la
7      ecuación cuadrática");
8      double b = in.nextDouble();
9      System.out.println("Ingrese el
10     parametro de c de la
11     ecuación cuadrática");
12     double c = in.nextDouble();

```

2. Por consiguiente primero se resolverá el discriminante, donde el resto de las operaciones de la fórmula cuadrática dependerá de este resultado.

```

1      double discriminante = (Math.pow(b,
2      2)) - (4*a*c);

```

3. Por último, si el discriminante pasa por la primera condición, se obtendrán 2 soluciones, la segunda condición solo tiene una solución, en caso contrario de todo, no hay una solución posible.

```

1      if (discriminante > 0) {
2          double x1 = (-b +
3          Math.sqrt(discriminante))
4          / (2 * a);
5          double x2 = (-b -
6          Math.sqrt(discriminante))
7          / (2 * a);
8
9          System.out.println("La
10         ecuación tiene dos
11         soluciones sobre el
12         conjunto de los números
13         reales: \nx1 = "+x1+"\nx2 = "+x2);
14     } else {
15         if (discriminante==0) {
16             double x = -b/(2*a);
17             System.out.println("La
18             ecuación solo tiene
19             una solución sobre
20             el conjunto de los
21             números reales: \nx = "+x);
22         } else {
23             System.out.println("La
24             ecuación no tiene
25             una solución en el
26             conjunto de los
27             números reales, esta
28             se encuentra en los
29             números complejos");
30         }
31     }

```

**III-B6. Depuración y pruebas:**

En la tabla II se muestran los resultados obtenidos al compilar el código.

a	b	c	$x_1, x_2$ o $x$
2	10	2	-0.20, -4.79
3	4	6	"solución en los números complejos"
-5	6	-1	0.2, 1.0
7	4	10	"solución en los números complejos"
1	-4	4	2

Tabla II: Tabla de corridas.

**III-C. Punto dada una circunferencia****III-C1. Definición del problema:**

La circunferencia es el lugar geométrico de los puntos del plano que equidistan de un punto fijo llamado centro (C) con coordenadas:

$$C : (x, y)$$

Esta contiene un elemento llamado radio (r) el cual se define como la línea que une el centro con cualquier punto de la circunferencia. La magnitud del radio determinará que tanto se abrirá la circunferencia. [3]

**III-C2. Descripción del problema:**

Dada una circunferencia con centro en el punto  $C$  con coordenadas  $(x_1, y_1)$  y radio  $r$ , evaluar si un punto  $T$  con coordenadas  $(x_2, y_2)$  está dentro de la circunferencia.

**III-C3. Diseño de solución:**

Para darle solución a este problema se analizaron los casos que pueden ocurrir dentro de este, encontrando tres casos diferentes:

1. El punto T se encuentra dentro de la circunferencia.
2. El punto T se encuentra fuera de la circunferencia.
3. El punto T se encuentra sobre la circunferencia.

Para determinar cual de los tres casos se cumplen se tiene que encontrar primero la distancia que hay entre el centro de la circunferencia y el punto T.

Usando la fórmula para la distancia  $d$  entre dos puntos en el plano cartesiano que se define como:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (5)$$

Con la ecuación 5 encontramos la magnitud entre el punto T y el centro de la circunferencia.

Se hace la comparación entre la distancia  $d$  que hay entre ambos puntos, con el valor del radio de la circunferencia:

1. Si  $d > r$  entonces el punto T se encuentra fuera de la circunferencia.
2. Si  $d < r$  entonces el punto T se encuentra dentro de la circunferencia.
3. Si  $d = r$  entonces el punto T se encuentra sobre la misma circunferencia.

Con esto entendido se procede a desarrollar la solución en lenguaje de programación Java.

**III-C4. Desarrollo de la solución:**

1. Se le solicita al usuario las coordenadas en x y en y del centro de la circunferencia.

```

1      System.out.println("Ingrese el valor
      de la coordenada en x del centro
      de la circunferencia: ");
2      double Cx = entrada.nextDouble();
3      System.out.println("Ingrese el valor
      de la coordenada en y del centro
      de la circunferencia: ");
4      double Cy = entrada.nextDouble();

```

2. Se le solicita al usuario los valores de las coordenadas en x y en y del punto T a evaluar.

```

1      System.out.println("Ingrese el valor
      de la coordenada en x del punto
      T: ");
2      double Tx = entrada.nextDouble();
3      System.out.println("Ingrese el valor
      de la coordenada en y del punto
      T: ");
4      double Ty = entrada.nextDouble();

```

3. Se le solicita al usuario el valor del radio de la circunferencia.

```

1      System.out.println("Ingrese el valor
      del radio de la circunferencia");
2      double r = entrada.nextDouble();

```

4. Se procesa la información ingresada por el usuario para obtener la distancia entre el centro de la circunferencia y el punto T.

```

1      double v1 = Math.pow((Tx-Cx), 2);
2      double v2 = Math.pow((Ty-Cy), 2);
3      double v3 = v1+v2;
4      double d = Math.sqrt(v3);

```

5. Se compara la distancia obtenida con el radio, haciendo uso del if, else if y else para saber que caso ocurre e imprimirlo al usuario.

```

1      if (d>r){
2          System.out.println("El punto T con
          coordenadas (" +Tx+", "+Ty+")
          se encuentra fuera de la
          circunferencia con centro en
          (" +Cx+", "+Cy+") y radio "+r);
3      }else if (d<r){
4          System.out.println("El punto T con
          coordenadas (" +Tx+", "+Ty+")
          se encuentra dentro de la
          circunferencia con centro en
          (" +Cx+", "+Cy+") y radio "+r);
5      }else if (d==r){
6          System.out.println("El punto T con
          coordenadas (" +Tx+", "+Ty+")
          se encuentra sobre la
          circunferencia con centro en
          (" +Cx+", "+Cy+") y radio "+r);
7      }

```

**III-C5. Depuración y pruebas:**

A continuación, en la tabla II se muestran los resultados de prueba al compilar el código.

$C_x$	$C_y$	$T_x$	$T_y$	$r$	Fuera/Dentro/Sobre
4	4	5	5	1	Fuera
3	4	6	8	5	Sobre
-5	6	-1	2	15	Dentro
0	0	0	5	5	Sobre
3	7	9	2	6	Dentro

Tabla III: Tabla de corridas.

**III-D. Conversión decimal a binario****III-D1. Definición del problema:**

El sistema decimal, es un sistema de numeración posicional en el que las cantidades se representan utilizando como base el número diez, por lo que se compone de diez dígitos diferentes: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. El valor de cada dígito está asociado a la posición que ocupa: unidades, decenas, centenas, millares, etc. Un sistema binario utiliza sólo dos dígitos: 0 y 1. El valor de cada posición se obtiene de una potencia de base dos, elevada a un exponente igual a la posición del dígito menos uno. [4]

**III-D2. Descripción del Problema:**

Dado un número decimal entero positivo o negativo regresar su equivalente en binario.

**III-D3. Diseño de solución:**

Para darle solución a este problema de un número entero decimal a binario, se debe utilizar el método de división sucesiva. Este método consiste en dividir el número decimal entre 2 y anotar el residuo. Luego, se divide el cociente obtenido entre 2 y se anota el nuevo residuo. Este proceso se repite hasta obtener un cociente igual a 0. Los residuos obtenidos, leídos de abajo hacia arriba, forman la representación binaria del número decimal. Desarrollando un código en Java, en donde el usuario ingrese un número entero decimal y obtenga como resultado su número binario.

**III-D4. Desarrollo de la solución:**

1. En el programa que se desarrolló en Java, el usuario tiene que ingresar un número entero decimal positivo siendo este el valor primordial para que se lleve a cabo la conversión a binario.

```
1      System.out.println("Ingrese un n mero
2      entero decimal positivo");
      int nmeroDecimal=entrada.nextInt();
```

2. Se utiliza la estructura en cascada, en donde se verifica si el número no es negativo e inicializa una cadena para almacenar el número binario y si el número decimal es igual a 0, se asigna "0." a la variable número binario. En caso de que el número no sea negativo, se utiliza un bucle while para convertir el número decimal a su representación binaria. Se calcula el residuo de la

división del número decimal entre 2 y se guarda en la variable residuo. Se concatena el residuo al inicio de la cadena número binario y después se divide el número decimal entre 2 y se actualiza el valor de número decimal. Finalmente se imprime el número binario resultante.

```
1      if (nmeroDecimal<0) {
2      System.out.println("El n mero debe ser
      entero no negativo");
3      return;
4      }
5      String nmeroBinario="";
6      if (nmeroDecimal==0) {
7      nmeroBinario="0";
8      } else {
9      while (nmeroDecimal>0) {
10     int residuo=nmeroDecimal%2;
11     nmeroBinario=residuo+nmeroBinario;
12     nmeroDecimal=nmeroDecimal/2;
13     }
14     }
15     System.out.println("El n mero binario
      equivalente es:" +
      nmeroBinario);
```

3. Por lo tanto, obtenemos el resultado de la conversión de el número entero decimal en binario.

**III-D5. Depuración y pruebas:**

A continuación en la tabla 4 se muestran resultados arrojados por el código.

Número	Número decimal	Número binario
1	222	11011110
2	338	101010010
3	110	1101110

Tabla IV: Tabla de corridas.

**III-E. Conversión binario a decimal****III-E1. Definición del problema:**

El código binario se basa en un sistema de dos elementos constituido por el cero y uno, el código binario es una secuencia de dígitos que representa el lenguaje que utilizan los equipos con circuitos digitales para procesar información y comandos.

El sistema binario es la base de la tecnología digital, cualquier dispositivo que tenga circuitos integrados (chips) es posible gracias a este sistema numérico.

El sistema binario se emplea por los ordenadores generalmente, tanto por los componentes hardware como por el software. [5]



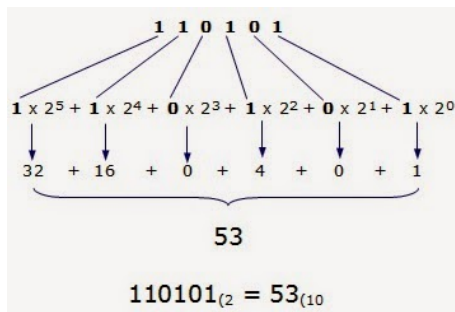


Figura 3: Ejemplo Binario-Decimal

**III-E2. Descripción del problema:**

Dado un número binario de  $n$  bits regresar su equivalente en decimal.

**III-E3. Diseño de solución:**

El proceso para convertir un número binario a decimal es un método diferente.

Consta de numerar los dígitos de derecha a izquierda comenzando desde cero, a cada número se le asigna la correspondiente potencia base 2 y al final se suman las potencias.

Lo primero que se debe realizar es ubicar una serie de potencias de dos que es la base del sistema binario, cuyo exponente va aumentando a medida que cambiamos la posición de derecha a izquierda, después se multiplica cada dígito del número binario por el resultado de la potencia, después se deben sumar los resultados para poder encontrar el número decimal.

**III-E4. Desarrollo de la solución del problema:**

1. Se diseñó un diagrama de flujo para poder realizar un programa en JAVA en el cual pide ingresar un número binario.

```
//ENTRADA
System.out.println("Ingresa un numero binario");
String numeroBin = in.nextLine();
int longitud = numeroBin.length();
int numeroDecimal = 0;
```

2. A base de una codificación se mostrará como resultado el número decimal correspondiente al número binario ingresado.

```
//PROCESO
for (int i = 0; i < longitud; i++) {
    char digito = numeroBin.charAt(i);
    //verificacion si es 0 o 1
    if (digito == '0') {
        numeroDecimal *= 2;
    } else if (digito == '1') {
        numeroDecimal = (numeroDecimal * 2) + 1;
    }
}
```

```
} else {
    System.out.println("El numero binario ingresado no es valido");
    return;
}
```

3. Y al final se mirará reflejado el número decimal correspondiente al número binario solicitado.

```
//SALIDA
System.out.println("El numero decimal equivalente es " + numeroDecimal);
```

**III-E5. Depuración y pruebas:**

A continuación en la siguiente tabla 5 se presentan los resultados de la prueba de compilación del código.

Número de corrida	Número binario	Número decimal
1	10101	21
2	1110001	113
3	101010111	343

Tabla V: Tabla de corridas.

**III-F. Tabla de verdad usando el teorema booleano****III-F1. Definición del problema:**

Las tablas de verdad son instrumentos de la lógica que permiten conocer los valores de verdad de proposiciones compuestas teniendo en cuenta las posibles interpretaciones de las proposiciones simples que la conforman. Algo que tener en cuenta es que el número de filas de la tabla de verdad será igual a  $2^n$  más la cabecera, donde  $n$  es el número de proposiciones simples (que podemos llamar variables). Los pasos para crear la tabla podemos resumirlos así:

1. Calcular el número de filas que tendrá la tabla de verdad. Este es igual a  $2^n$  más la cabecera, donde  $n$  es el número de proposiciones simples.
2. Calcular el número de columnas que tendrá la tabla. Este es igual a el número de proposiciones simples más la cantidad de conectores lógicos que aparecen en la expresión, contando cada repetición de ellos.
3. Desglosar la proposición compuesta en la cabecera teniendo en cuenta la jerarquía de los conectores. En las primeras columnas irán las proposiciones simples y en la última irá la proposición original.
4. Escribir en las filas todas las posibles interpretaciones de las proposiciones simples. Por ejemplo, si se trata de dos proposiciones, existen cuatro interpretaciones posibles, lo veremos más adelante con los ejemplos.
5. Completar el resto de la tabla teniendo en cuenta las tablas de verdad de los conectores lógicos. La última columna, la de la proposición original que analizamos, nos dirá su valor de verdad de acuerdo a las posibles interpretaciones. [6]

**III-F2. Descripción del problema:**

Dada una tabla de verdad de " $n$ " bits generar la expresión booleana que genere de manera fidedigna las salidas de esta tabla.

**III-F3. Diseño de solución:**

Este sistema consolida los valores de verdad de todas las combinaciones de premisas y operadores lógicos en un solo lugar en columna. De esta manera, evita que tengas que “calcular” manualmente la conclusión de premisas falsas y verdaderas, muestran todos los posibles escenarios y condiciones de valores de entrada para una operación lógica y su resultado correspondiente, normalmente se utilizan los valores 1 y 0 donde 1 es verdadero y 0 es falso.

Su función base es mostrar cómo funcionan un circuito electrónico y los programas de una computadora, siendo también un pilar de la lógica proposicional.

También la tabla de verdad tiene sus funciones mas importantes como :

1. Representan relaciones lógicas.
2. Analizan todas las combinaciones posibles de valores.
3. Ayudan a deducir conclusiones lógicas.
4. Evalúan comportamiento de sistemas ante diferentes entradas.
5. Mejoran la eficiencia de algoritmos en programación.

También para la correcta elaboración del programa referente a la tabla de verdad se utiliza lo que es el teorema booleano donde:

1.  $x * 0 = 0$
2.  $x * 1 = x$
3.  $x * x = x$
4.  $x . * x = 0$
5.  $x + 0 = x$
6.  $x + 1 = 1$
7.  $x + x = x$
8.  $x + x = 0$

Se hizo un código donde se tiene la opción de ingresar la cantidad de bits que uno quiera para eso debemos tener como ingresar la cantidad de bits que deseas para poder elaborar la tabla de valores.

**III-F4. Desarrollo de solución:**

1. Para poder ejecutar el problema el código del programa se realizo en el lenguaje de programación java.

```

1  import java.util.Scanner;
2  public class tabla_de_verdad {
3      public static void main(String[] args)
4      {
5          Scanner valor=new Scanner
6              (System.in);
7          System.out.println("ingresa el
8              numero de bits");
9          int numVariables = valor.nextInt();
10         if(numVariables < 5 ){
11             generateTruthTable(numVariables);
12         }else{
13             System.out.println("Numero
14                 fuera de rango permitido
15                 :");
16         }
17     }
18     public static void
19         generateTruthTable(int
20             numVariables) {
21         int numRows = (int) Math.pow(2,
22             numVariables);

```

```

15         for (int i = 0; i < numVariables;
16             i++) {
17             System.out.print("Var" + (i +
18                 1) + "\t");
19         }
20         System.out.println("Result");
21         for (int i = 0; i < numRows; i++) {
22             for (int j = numVariables - 1;
23                 j >= 0; j--) {
24                 int value = (i / (int)
25                     Math.pow(2, j)) % 2;
26                 System.out.print(value +
27                     "\t");
28             }
29             boolean result =
30                 calculateLogicOperation(i,
31                     numVariables);
32             System.out.println(result ?
33                 "1" : "0");
34         }
35     }
36     public static boolean
37         calculateLogicOperation(int row,
38             int numVariables) {
39
40         boolean var1 = ((row >> 2) & 1) ==
41             1;
42         boolean var2 = ((row >> 1) & 1) ==
43             1;
44         boolean var3 = (row & 1) == 1;
45         return var1 && var2 && var3;
46     }
47 }

```

Para obtener el resultado en expresión booleana primero se debé de modificar una fila como ejemplo en la columna a se modifican las filas que serían del 1 al 8, porque ocho buenos porque es la cantidad de filas que la columna la tiene. Quiero modificar la fila solo pones el número que corresponda, vas a quieres modificar la fila 2 donde la primera es 0 se cambia a 1 eso varía donde puedes iniciar con 1 y cambia a 0 o viceversa, después de modificar los números que quieras modificar, ya después te dará la ecuación booleana

Ingrese el número de la fila que desea cambiar el resultado a 1 (1-8) o ingrese 0 para finalizar: 0 Expresiones booleanas al finalizar:  $A' + B' + C' A' + B' + C A' + B + C A + B' + C A + B + C' A + B + C$ .

Expresión booleana final:  $(A' + B' + C') + (A' + B' + C) + (A' + B + C) + (A + B' + C) + (A + B + C') + (A + B + C)$

**III-F5. Depuración y pruebas:**

En la tabla 6 se muestra un resultado obtenido por el código.

A	B	B	RESULTADO
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Tabla VI: Tabla de corridas.

## IV. CONCLUSIÓN

Con cada uno de los problemas que se resolvieron, se logró una mejor capacitación en distintas áreas como el uso del cálculo diferencial, las búsquedas con números binarios y encontrar sus expresiones booleanas, así como principalmente se desarrolló una mejor comprensión de como poder trabajar en estas áreas con el uso de la programación y mejorar una mejor lógica de desarrollo, al igual de como poder incluir herramientas para trabajar de una manera remota y muy eficiente para poder trabajar mejor en equipo.

## V. AGRADECIMIENTOS

Como equipo mostramos nuestro agradecimiento a cada uno de nuestros docentes de cada una de las materias que contiene el primer semestre de Ingeniería en Tecnologías de la Información y Comunicaciones, específicamente al **Mtro. José Martín Oropeza Méndez** quien con entusiasmo nos impartió la asignatura de Matemáticas Discretas I, al **Ing. Giovany Humberto Neri Pérez** quien con sus conocimientos en matemáticas nos impartió la asignatura de Calculo Diferencial, a la **Mtra. Silvia Sánchez Bautista** quien con alegría nos impartió la asignatura de Ética, a la **Lic. Irais López Rojo** quien con su aprendizaje en enseñanza de la segunda lengua nos impartió la asignatura de Inglés I, al **LSC. Agustín Soto Arista** quien nos ayudó a tener un mejor lenguaje y nos impartió la asignatura de Fundamentos de Investigación, al **Dr. Francisco Javier Cuadros Romero** quien con mucho conocimiento y clases divertidas nos impartió la asignatura de Introducción a las TIC's, y especialmente agradecidos con la **MGTI. Citlali Azucena Martínez Calva** quien es tutora de nuestro primer semestre en la carrera y, con sus conocimientos y apoyo extra-clase nos impartió la asignatura de Fundamentos de Programación. Agradecidos con nuestros padres y amigos que nos apoyaron durante la realización de este proyecto. Gracias a todos.

## REFERENCIAS

- [1] Khan Academy, October 20, 2023, Tangent lines and rates of change, <https://www.khanacademy.org/math/calculus-all-ol/taking-derivatives-calc/using-the-formal-definition-of-derivative-calc/a/tangent-lines-and-rates-of-change>.
- [2] Tutorela. (2021, 16 noviembre). La fórmula cuadrática. Tutorela. <https://www.tutorela.es/matematicas/la-formula-cuadratica>
- [3] Gómez, I. P. Y. F. (2019, 29 junio). Ecuación de la circunferencia - [Ordinaria y general. ¡Ejercicios resueltos!]. Álgebra y Geometría Analítica. <https://aga.frba.utn.edu.ar/circunferencia/>
- [4] Porto, J. P., Merino, M. (2021, 26 agosto). Sistema decimal - qué es, definición, importancia y clasificación. Definición.de. <https://definicion.de/sistema-decimal/>
- [5] Equipo editorial, Etecé. (2022, 2 febrero). Sistema binario - qué es, concepto, aplicaciones y ejercicios. Concepto. <https://concepto.de/sistema-binario/>
- [6] Machado, D. (2023, 4 noviembre). Cómo hacer tablas de verdad: ejercicios resueltos. Flamath. <https://flamath.com/tablas-de-verdad>

## VI. INTEGRANTES

VI-A. *Cedillo Ramírez Jesús Emmanuel*

Un estudiante de ingeniería que desconoce muchos temas de programación pero que muestra interés en poder aprender diversos temas sobre la programación. Nacido en el municipio de Ixmiquilpan, Hgo, con gustos musicales muy diversos, desde pequeño demostró un interés en temas relacionados con audio, actualmente se encuentra estudiando la carrera de Tecnologías de la Información y Comunicación en el Instituto Superior del Occidente del Estado de Hidalgo, en la cual le ha surgido el interés por el tema de ciberseguridad.

[GitHub de Emmanuel](#)

VI-B. *Gómez Ángeles Evelin*

Una estudiante de ingeniería emocionada por aprender de tecnología y todo lo relacionado con ella. Sus intereses radican en aprender más sobre las diferentes tecnologías y sus aplicaciones en el mundo actual, además de que aspira a adquirir nuevas habilidades de programación para poder diseñar y desarrollar una aplicación que sea útil y beneficie a las personas.

[GitHub de Evelin](#)

VI-C. *Gonzaga López Luis Fernando*

Un estudiante de ingeniería apasionado por la ciencia, la tecnología y las matemáticas. Abierto a diferentes gustos musicales. Nacido en el municipio de Tlaxcoapan, Hgo. y actualmente viviendo en el municipio de Tlahuelilpan, Hgo. Desde la niñez mostró interés en computadoras y en resolver problemas matemáticos. Es jugador de baloncesto desde los 10 años. Sus principales principios como persona son la lealtad y la perseverancia. El principal objetivo de Fernando actualmente es completar sus estudios de nivel superior que actualmente está haciendo en el Instituto Tecnológico Superior del Occidente del Estado de Hidalgo en el municipio de Mixquiahuala de Juárez, Hgo.

[GitHub de Fernando](#)

VI-D. *González Calva Yael*

Un estudiante de Ingeniería que recientemente ha empezado con cariño a la tecnología que se esfuerza por aprender y de apoyar a sus compañeros cuando lo requieran. Nacido en Francisco I. Madero, Hgo. Cuando era pequeño siempre tuvo el amor hacia el deporte, practicando fútbol y baloncesto a la edad de 9 años, de pocos amigos pero muy buenas personas. Actualmente esta estudiando en el Instituto Superior del Occidente del Estado de Hidalgo, su plan en mente es terminar la carrera y adquirir buenas habilidades para desenvolverse de gran manera en el ámbito laboral.

[GitHub de Yael](#)

**VI-E. Jarquín Contreras Kevin Ichiro**

Un estudiante que creció en Mixquiahuala, Hgo. parte de su vida, que tiene un interés por las nuevas tecnologías y el funcionamiento de ellas, por el momento trato de entender el funcionamiento de eso por eso estudio esta ingeniería en el Instituto Tecnológico Superior del Occidente del Estado de Hidalgo , por el momento tiene una meta en esta carrera y es crear una inteligencia artificial para después de eso poder trabajar de lo que estudia, para esto falta mucho, por el momento tendrá que enfocarse en el ahora para poder lograr sus objetivos a corto y largo plazo.

[Github de Ichiro](#)

**VI-F. López Paz Gustavo**

Un estudiante que creció en Progreso de Obregón, Hgo. Desde pequeño paso por varios problemas de salud que enfrentar pero con apoyo de sus padres logro salir de ellos, en ese transcurso empezó a tomar cariño de algunos deportes como fútbol y el culturismo, al igual ser curioso con la tecnología que vivía en su alrededor. Con el transcurso del tiempo empezó a aprender por su cuenta programación básica y otras herramientas para trabajar. Hoy ahora que estudia en el Instituto Superior del Occidente del Estado de Hidalgo, en la ingeniería en Tecnologías de la Información y Comunicación, donde también se ha interesado en conocer Internet de las cosas y el manejo de Redes.

[Github de Gustavo](#)