



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

## Proyecto Integrador

Prototipo de problemas de solución de ingeniería aplicado a primer semestre

### Coordinador del proyecto:

Dr. Francisco Javier Cuadros Romero

### Integrantes del equipo:

1<sup>er</sup> Cruz Granados Dulce Marina  
*Estudiante de ITIC 's*  
*ITSOEH*  
Tezontepec de Aldama, Hgo.  
230110477@itsoeh.edu.mx

2<sup>do</sup> Fuentes Pérez Bryan  
*Estudiante de ITIC 's*  
*ITSOEH*  
El Rosario, Hgo.  
230110581@itsoeh.edu.mx

3<sup>er</sup> Hernández Monroy Ricardo  
*Estudiante de ITIC 's*  
*ITSOEH*  
Tezontepec de Aldama, Hgo.  
230110240@itsoeh.edu.mx

4<sup>to</sup> Salazar Guerrero Diego  
*Estudiante de ITIC 's*  
*ITSOEH*  
San Salvador, HGO.  
230110580@itsoeh.edu.mx

5<sup>to</sup> Santiago Reséndiz Melanie  
*Estudiante de ITIC 's*  
*ITSOEH*  
Actopan, HGO.  
230110616@itsoeh.edu.mx

---

## ÍNDICE

<b>I.</b>	<b>Introducción</b>	<b>3</b>
<b>II.</b>	<b>Objetivos</b>	<b>3</b>
II-A.	Objetivo General . . . . .	3
II-B.	Objetivos específicos . . . . .	3
<b>III.</b>	<b>Ecuación de la recta</b>	<b>3</b>
III-A.	Definición . . . . .	3
III-B.	Descripción del problema . . . . .	3
III-C.	Diseño de la solución . . . . .	3
III-D.	Desarrollo de la solución . . . . .	4
III-E.	Depuración y pruebas . . . . .	4
<b>IV.</b>	<b>Raíces de una Ecuación Cuadrática</b>	<b>4</b>
IV-A.	Definición . . . . .	4
IV-B.	Descripción del problema . . . . .	4
IV-C.	Diseño de solución . . . . .	5
IV-D.	Desarrollo de solución . . . . .	5
IV-E.	Depuración y pruebas . . . . .	5
<b>V.</b>	<b>Punto en una Circunferencia</b>	<b>5</b>
V-A.	Definición . . . . .	5
V-B.	Descripción del problema . . . . .	5
V-C.	Diseño de solución . . . . .	5
V-D.	Desarrollo de solución . . . . .	5
V-E.	Depuración y pruebas . . . . .	6
<b>VI.</b>	<b>Conversion de decimal a binario con signo</b>	<b>6</b>
VI-A.	Definición de la solución . . . . .	6
VI-B.	Descripción del problema . . . . .	6
VI-C.	Diseño de la solución . . . . .	6
VI-D.	Desarrollo de la solución . . . . .	6
VI-E.	Depuración y pruebas . . . . .	7
<b>VII.</b>	<b>Binario a decimal</b>	<b>7</b>
VII-A.	Definición . . . . .	7
VII-B.	Descripción del problema . . . . .	7
VII-C.	Diseño de solución . . . . .	7
VII-D.	Desarrollo de solución . . . . .	7
VII-E.	Depuración y prueba . . . . .	8
<b>VIII.</b>	<b>Tabla de verdad con expresión booleana</b>	<b>8</b>
VIII-A.	Definición . . . . .	8
VIII-B.	Descripción de el problema . . . . .	8
VIII-C.	Diseño de la solución . . . . .	8
VIII-D.	Desarrollo de la solución . . . . .	8
VIII-E.	Depuración y pruebas . . . . .	9
<b>IX.</b>	<b>Conclusión</b>	<b>10</b>
	<b>Referencias</b>	<b>10</b>
<b>X.</b>	<b>Biografías</b>	<b>10</b>

**Resumen**—During the period from August to December 2023, we covered different subjects in the Information and Communication Technology Engineering program. In each of these subjects, the exercises to be solved were complemented and related. In differential calculus, the first three exercises were addressed: equation of a line, quadratic equation, and point on a circumference. In the discrete mathematics subject, the following three problems were discussed: conversion of negative or positive decimal integers to binary, conversion of binary numbers to decimal, and truth tables. In the ethics subject, a connection was established with references and research fundamentals, delving deeper into the study of these exercises. In the Introduction to ICT subject, knowledge was applied to topics such as the use of repositories. Finally, in the programming subject, the solutions to the problems were coded.

**Index Terms**—recta, interseccion, coordenada, punto

## I. INTRODUCCIÓN

En el siguiente documento se hablará sobre seis problemas aplicados al primer semestre de Ingeniería en Tecnologías de la Información y Comunicaciones creando programas en JAVA.

El primer problema habla sobre la pendiente de la recta en el cual dadas las coordenadas  $(x_1, y_1)$  y  $(x_2, y_2)$  al igual que los puntos A y B nos regrese la ecuación de la recta y el ángulo interno que se forma entre el eje horizontal y la recta. El segundo problema es encontrar las raíces de una ecuación de grado dos o ecuación cuadrática de la forma  $ax^2 + bx + c = 0$ . El tercer problema nos pide determinar si un punto en el plano cartesiano está dentro o fuera de una circunferencia con centro fuera del origen y de un radio  $r$ . El cuarto problema requiere comprender la numeración binaria para así poder crear el programa que nos haga la conversión de un número decimal a un número binario equivalente. El quinto problema es parecido al anterior solo que viceversa, ahora es encontrar el número decimal equivalente a un número binario dado por el usuario. El sexto problema nos pide que dados ciertos números de bits se cree una tabla de verdad correspondiente a la cantidad de bits ingresada por el usuario.

Estos seis problemas matemáticos que se presentan a continuación en este informe fueron resueltos con la metodología de las 6Ds que consiste en dar primero una **Definición** del problema o información acerca del tema que aborda el problema, después **Describir** el problema que queremos solucionar, luego de ello **Diseñar** una solución analizando los datos dados, para después **Desarrollar** la solución paso por paso y mencionando el código hecho en JAVA para cada problema, y continuar con **Depuración** y pruebas que consiste en generar una tabla de corridas o compilaciones del código, para terminar con la última D que es **Documentar** todo lo anterior, lo cual se muestra a continuación.

## II. OBJETIVOS

### II-A. Objetivo General

- El objetivo general del uso de software de resolución de problemas de ingeniería es mejorar la eficiencia

y efectividad en la resolución de desafíos técnicos y científicos en el campo de la ingeniería, a través de la utilización de herramientas informáticas especializadas.

### II-B. Objetivos específicos

1. Adquirir conocimientos sobre las características y funcionalidades de diferentes software de resolución de problemas de ingeniería utilizados en el campo específico de estudio.
2. Familiarizarse con la interfaz y la forma de operar de los software seleccionados, incluyendo la entrada de datos, la selección de algoritmos y la interpretación de resultados.
3. Aplicar el software en el modelado y diseño de sistemas y componentes, con el fin de optimizar su funcionamiento y mejorar su eficiencia.
4. Utilizar el software para realizar análisis de datos y visualización de resultados, con el fin de tomar decisiones fundamentadas y extraer conclusiones relevantes.
5. Desarrollar habilidades para la resolución de problemas técnicos mediante la aplicación de software de ingeniería, incluyendo la identificación de variables clave, la formulación de hipótesis y la realización de experimentos virtuales.
6. Promover el trabajo en equipo y la colaboración utilizando el software, permitiendo compartir proyectos, datos y resultados entre diferentes miembros del equipo.
7. Desarrollar habilidades de comunicación técnica al presentar y explicar los resultados obtenidos mediante el software de resolución de problemas de ingeniería.
8. Fomentar la actualización y el aprendizaje continuo sobre nuevas herramientas y técnicas de software de ingeniería, para estar al tanto de los avances tecnológicos en el campo.
9. Preparar a los estudiantes para enfrentar los desafíos del mundo laboral en ingeniería, donde el uso de software de resolución de problemas es una habilidad altamente demandada.

## III. ECUACIÓN DE LA RECTA

### III-A. Definición

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-B. Descripción del problema

En este problema, se tienen dos puntos A y B con coordenadas  $(x_1, y_1)$  y  $(x_2, y_2)$  respectivamente. El objetivo es calcular la ecuación de la recta que pasa por estos puntos y determinar el ángulo interno  $\alpha$  que dicha recta forma con el eje horizontal.

### III-C. Diseño de la solución

La ecuación de la recta se obtiene utilizando la fórmula de la pendiente:

$$m = \frac{y_2 - y_1}{x_2 - x_1} \quad (1)$$

Una vez obtenida la pendiente  $m$ , el ángulo interno  $\alpha$  se puede calcular utilizando la función trigonométrica arcotangente:

$$\alpha = \arctan(m) \quad (2)$$

### III-D. Desarrollo de la solución

Para resolver el problema, se puede utilizar un programa en JAVA. A continuación se muestra un ejemplo de código:

```

1 import java.util.Scanner;
2
3 public class recta {
4     public static void main(String[] args) {
5         Scanner scanner = new Scanner(System.in);
6
7         System.out.println("Ingrese las coordenadas
8             del punto A:");
9         System.out.print("x1: ");
10        double x1 = scanner.nextDouble();
11        System.out.print("y1: ");
12        double y1 = scanner.nextDouble();
13
14        System.out.println("Ingrese las coordenadas
15            del punto B:");
16        System.out.print("x2: ");
17        double x2 = scanner.nextDouble();
18        System.out.print("y2: ");
19        double y2 = scanner.nextDouble();
20
21        // Calcular la pendiente de la recta
22        double pendiente = (y2 - y1) / (x2 - x1);
23
24        // Calcular el ngulo en radianes
25        double anguloRadianes =
26            Math.atan(pendiente);
27
28        // Convertir el ngulo a grados
29        double anguloGrados =
30            Math.toDegrees(anguloRadianes);
31
32        // Calcular el ngulo interno entre el
33            eje horizontal y la recta
34        double anguloInterno = 90 - anguloGrados;
35
36        // Calcular el punto de intersección con
37            el eje y (cuando x = 0)
38        double puntoInterseccionY = y1 - pendiente
39            * x1;
40
41        // Construir la ecuación de la recta en
42            formato y = mx + b
43        String ecuacionRecta = "y = " + pendiente +
44            "x + " + puntoInterseccionY;
45
46        System.out.println("Ecuación de la recta:
47            " + ecuacionRecta);
48        System.out.println("ngulo interno : " +
49            anguloInterno + "grados");
50        System.out.println("Punto de intersección
51            con el eje y: (0, " +
52            puntoInterseccionY + ")");
53        System.out.println("Pendiente =
54            "+pendiente);
55    }
56 }

```

### III-E. Depuración y pruebas

Durante el desarrollo y ejecución del programa, es posible encontrar errores o problemas. Para depurar el programa, se pueden utilizar técnicas como la impresión de mensajes de depuración, el uso de herramientas de depuración y la revisión del código en busca de posibles errores. Luego de depurar el programa, se puede proceder a realizar pruebas para verificar su funcionamiento correcto.

En el cuadro 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

Cuadro I: Tabla de corridas.

## IV. RAÍCES DE UNA ECUACIÓN CUADRÁTICA

### IV-A. Definición

Es una expresión que puede aplicarse para calcular el valor de una variable a partir de determinados datos. Usualmente, con fórmula general se hace referencia a aquella que permite resolver ecuaciones cuadráticas, es decir, ecuaciones de segundo grado. Estas son aquellas donde el máximo exponente al que está elevada la incógnita es 2 y que tiene la siguiente forma:

$$ax^2 + bx + c = 0 \quad (3)$$

Resolverla significa encontrar todos aquellos números reales, que cumplen la igualdad cuando sustituyen a  $x$  en la ecuación. Cada uno de esos números, se llama solución de la ecuación. También es frecuente llamarles raíces o ceros de la ecuación. Tomando como referencia la ecuación 3, la fórmula general para resolver la ecuación es la siguiente:

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (4)$$

Se observa que hay dos posibles soluciones, una por cada signo de la raíz cuadrada, por esto la  $x$  aparece con dos subíndices: uno por cada solución. Le llamamos discriminante a la expresión que se encuentra dentro de la raíz:  $b^2 - 4ac$ , este discriminante nos permitirá determinar si las soluciones de la ecuación se encuentran dentro del conjunto de los números reales ( $\mathbb{R}$ ) o en el conjunto de los números complejos ( $\mathbb{C}$ ), establecido por el signo de este. [2]

### IV-B. Descripción del problema

Dada una ecuación cuadrática regresar los valores de las raíces en caso de que estén sobre el conjunto de los números reales, en caso contrario indicar que la solución está en el conjunto de los números complejos.

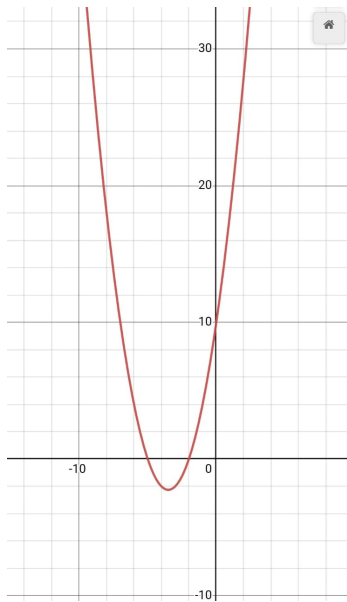


Figura 1: Gráfica de una ecuación cuadrática.

#### IV-C. Diseño de solución

El discriminante que definiremos como  $D = b^2 - 4ac$  determina la naturaleza de las raíces:

- Si  $D > 0$ , las raíces son números reales distintos.
- Si  $D = 0$ , las raíces son números reales iguales.
- Si  $D < 0$ , las raíces son números complejos conjugados.

#### IV-D. Desarrollo de solución

```

1      System.out.print("Coeficiente a: ");
2      double a = input.nextDouble();
3      System.out.print("Coeficiente b: ");
4      double b = input.nextDouble();
5      System.out.print("Coeficiente c: ");
6      double c = input.nextDouble();
7      double discriminante = b * b - 4 * a * c;
8      if (discriminante > 0) {
9          double raiz1 = (-b +
10             Math.sqrt(discriminante)) / (2 *
11                a);
12             double raiz2 = (-b -
13                Math.sqrt(discriminante)) / (2 *
14                   a);
15             System.out.println("Las raices son
16                 numeros reales distintos:");
17             System.out.println("Raiz 1: " + raiz1);
18             System.out.println("Raiz 2: " + raiz2);
19         } else if (discriminante == 0) {
20             double raiz = -b / (2 * a);
21             System.out.println("Las raices son
22                 numeros reales iguales:");
23             System.out.println("Raiz: " + raiz);
24         } else {
25             System.out.println("Las raices son
26                 numeros complejos conjugados.");
27         }

```

#### IV-E. Depuración y pruebas

En el cuadro 1 se muestra una tabla de pruebas con diferentes valores en  $a$ ,  $b$  y  $c$  en las tres primeras columnas,

$a$	$b$	$c$	$x_1$	$x_2$
1	1	1	Complejos	Complejos
1	10	25	-5	-5
3	-5	2	1	2/3
6	-36	0	6	0
1	2	3	Complejos	Complejos

Cuadro II: Tabla de corridas.

y el valor de las raíces arrojadas por el programa, en las dos últimas columnas.

### V. PUNTO EN UNA CIRCUNFERENCIA

#### V-A. Definición

El objetivo de este problema de la ingeniería es determinar la inclusión de un punto  $T$ , definido por sus coordenadas  $(x_2, y_2)$ , en el área de una circunferencia específica. Esta posee un centro en el punto  $C$ , con coordenadas  $(x_1, y_1)$ , y un radio de longitud  $r$ . Se evalúa si el punto  $T$  se encuentra dentro del perímetro de la circunferencia o si, por el contrario, se ubica fuera de dicho perímetro

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

Resulta necesario llevar a cabo una comparación exhaustiva entre la distancia que separa al centro de la circunferencia y el punto  $T$ , y la longitud del radio. Si la distancia obtenida resulta ser menor o igual a la longitud del radio, se puede concluir que el punto  $T$  se encuentra ubicado dentro de los límites de la circunferencia. Por el contrario, si la distancia es mayor a la longitud del radio, se establece que el punto  $T$  se encuentra situado en una posición exterior con respecto a la circunferencia.

#### V-B. 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 del área de la circunferencia.

#### V-C. Diseño de solución

La distancia entre el centro  $C$  y el punto  $T$  se calcula utilizando la fórmula de la distancia entre dos puntos en el plano la cual se define como anteriormente en la ecuación 5.

#### V-D. Desarrollo de solución

```

1      Scanner scanner = new Scanner(System.in);
2
3      System.out.println("Ingrese la coordenada
4          x1");
5      double x1 = scanner.nextDouble();
6      System.out.println("Ingrese la coordenada
7          y1");
8      double y1 = scanner.nextDouble();
9
10     System.out.println("Ingrese el radio de la
11         circunferencia (r):");
12     double r = scanner.nextDouble();

```

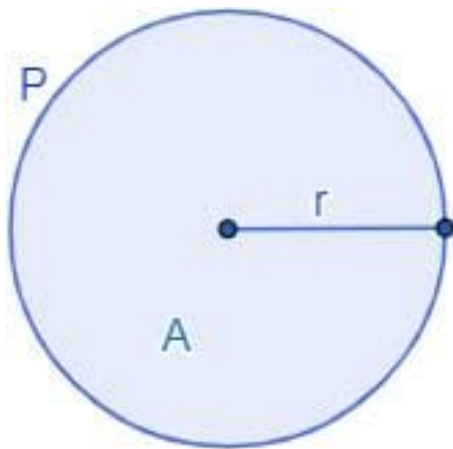


Figura 2: Gráfica de una circunferencia

```

14
15 System.out.println("Ingrese la coordenada a
16     evaluar con el punto x2");
17 double Tx2 = scanner.nextDouble();
18 System.out.println("ingrese la coordenada a
19     evaluar con el punto y2");
20 double Ty2 = scanner.nextDouble();
21
22 double distancia = Math.sqrt(Math.pow(Tx2 -
23     x1, 2) + Math.pow(Ty2 - y1, 2));
24
25 if (distancia <= r) {
26     System.out.println("El punto se encuentra
27         dentro del area de la
28         circunferencia.");
29 } else {
30     System.out.println("El punto esta fuera
31         del area de la circunferencia.");
32 }
33 }
34 }

```

#### V-E. Depuración y pruebas

En el cuadro 2 se muestran los resultados obtenidos al compilar el código en JAVA de este problema.

$x_1$	$y_1$	$r$	$x_2$	$y_2$	Afuera o Adentro
1	2	4	2	1	Adentro
3	4	2	2	1	Afuera
1	6	4	4	5	Adentro
7	4	10	-5	-3	Afuera
-4	-5	6	-7	-8	Adentro

Cuadro III: Tabla de corridas.

## VI. CONVERSION DE DECIMAL A BINARIO CON SIGNO

### VI-A. Definición de la solución

El problema planteado consiste en obtener el equivalente binario de un número decimal con signo. Se debe respetar la regla de conversión de complemento a dos para los números negativos, con el objetivo de diseñar un algoritmo capaz de tomar cualquier número decimal como entrada y devolver su representación binaria correspondiente. [5]

### VI-B. Descripción del problema

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

### VI-C. Diseño de la solución

Para diseñar una solución viable al progreso del algoritmo, necesitaremos establecer los siguientes pasos:

1. Definir el número decimal para después convertir a binario.
2. Invertir los bits.
3. Sumar 1 al bit menos significativo. Así obtendremos el complemento a dos.

10110010	Número binario
01001101	Complemento a 1
+ 1	Sumar 1
<u>01001110</u>	Complemento a 2

Figura 3: Procedimiento del complemento a dos.

### VI-D. Desarrollo de la solución

Se presentará el algoritmo, desarrollado en el lenguaje de programación JAVA.

```

1
2
3 import java.util.Scanner;
4 public class Ejercicio4 {
5
6
7     public static void main(String[] args) {
8         //entrada
9         Scanner entrada = new Scanner(System.in);
10
11         // Solicitar al usuario que ingrese un nmero
12         // decimal
13         System.out.print("Ingrese un nmero decimal:
14         ");
15         int numeroDecimal = entrada.nextInt();
16
17         // Verificar si el nmero es negativo
18         boolean esNegativo = false;
19         if (numeroDecimal < 0) {
20             esNegativo = true;
21             numeroDecimal = Math.abs(numeroDecimal);
22         }
23
24         // Inicializar una cadena para almacenar el
25         // nmero binario
26         String numeroBinario = "";
27
28         // Manejar el caso especial cuando el nmero
29         // decimal es 0
30         if (numeroDecimal == 0) {
31             numeroBinario = "0";
32         } else {
33             // Convertir el nmero decimal a binario
34             while (numeroDecimal > 0) {
35                 int residuo = numeroDecimal % 2; //
36                 // Obtener el residuo de la division
37                 // por 2
38                 numeroBinario = residuo +
39                     numeroBinario; // Agregar el
40                     residuo al inicio de la cadena
41                     binaria
42                 numeroDecimal = numeroDecimal / 2; //
43                 // Dividir el nmero decimal por 2

```

```

34     }
35
36     // Calcular el complemento a dos si el
37     // nmero es negativo
38     if (esNegativo) {
39         StringBuilder complementoBuilder = new
40         StringBuilder();
41         boolean encontradoPrimerUno = false;
42
43         // Invertir los bits
44         for (int i = numeroBinario.length() -
45             1; i >= 0; i--) {
46             char bit = numeroBinario.charAt(i);
47             if (!encontradoPrimerUno) {
48
49                 if (bit == '1') {
50                     encontradoPrimerUno = true;
51                 }
52                 complementoBuilder.insert(0, bit);
53             } else {
54                 // Invertir los bits restantes
55                 if (bit == '0') {
56                     complementoBuilder.insert(0,
57                         '1');
58                 } else {
59                     complementoBuilder.insert(0,
60                         '0');
61                 }
62             }
63         }
64
65         numeroBinario =
66             complementoBuilder.toString();
67     }
68
69     // Agregar signo negativo si corresponde
70     if (esNegativo) {
71         numeroBinario = "-" + numeroBinario;
72     }
73
74     // Imprimir el nmero binario resultante
75     System.out.println("El nmero binario
76     equivalente es: " + numeroBinario);
77 }
78 }

```

#### VI-E. Depuración y pruebas

La siguiente tabla muestra los resultados del programa en JAVA, que constituyen el resultado final.

No.de corridas	No.Decimal	Complemento a 2
1	-28	00100
2	222	11011110
3	-1480	01000111000
4	10	1010

### VII. BINARIO A DECIMAL

#### VII-A. Definición

El problema consiste en convertir un número binario de  $n$  bits a su equivalente en decimal. Se desea desarrollar un programa que realice esta conversión de manera eficiente y precisa.

#### VII-B. Descripción del problema

Un número binario está compuesto únicamente por los dígitos 0 y 1, y su representación en decimal se basa en el sistema numérico posicional, donde cada dígito tiene un valor

asociado dependiendo de su posición. El bit más a la derecha tiene un valor de  $2^0$ , el siguiente bit tiene un valor de  $2^1$ , el siguiente  $2^2$ , y así sucesivamente. Para convertir un número binario a decimal, se deben multiplicar los dígitos binarios por las potencias de 2 correspondientes y luego sumar los resultados.

Por ejemplo, el número binario 1011 se convierte a decimal de la siguiente manera:

$$1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 11$$

El objetivo es desarrollar un programa que tome como entrada un número binario de  $n$  bits y devuelva su equivalente en decimal.

#### VII-C. Diseño de solución

El diseño de la solución implica el desarrollo de un algoritmo que realice la conversión del número binario a decimal. El algoritmo seguirá los siguientes pasos:

1. Solicitar al usuario que ingrese un número binario de  $n$  bits.
2. Inicializar una variable decimal en 0 para almacenar el resultado de la conversión.
3. Recorrer el número binario de derecha a izquierda, comenzando desde el bit más a la derecha:
  - Obtener el dígito binario en la posición actual.
  - Multiplicar el dígito por  $2^i$ , donde  $i$  es la posición actual.
  - Sumar el resultado a la variable decimal.
4. Mostrar el resultado decimal, que es el número decimal equivalente al número binario ingresado.

#### VII-D. Desarrollo de solución

A continuación se presenta la implementación del algoritmo en el lenguaje de programación JAVA:

```

1 import java.util.Scanner;
2 public class bin {
3
4     public static void main(String[] args) {
5         // entrada
6         Scanner entrada = new Scanner(System.in);
7         System.out.println("Ingresa un numero
8         binario");
9         String numeroBinario=entrada.nextLine();
10
11         int longitud=numeroBinario.length();
12
13         int numeroDecimal=0;
14
15         for(int i=0; i<longitud; i++){
16             char digito=numeroBinario.charAt(i);
17             //verificar si es 0 o 1
18             if(digito=='0'){
19                 numeroDecimal= numeroDecimal*2;
20
21             }else if(digito == '1'){
22                 numeroDecimal= numeroDecimal*2+1;
23             }else{
24                 System.out.println("El numero binario
25                 ingresado no es valido");
26                 return;
27             }
28         }
29     }
30 }

```

```

27     }
28     System.out.println("El numero decimal
    equivalente es:" + numeroDecimal);
29 }
30 }

```

El programa solicita al usuario que ingrese un número binario y luego utiliza la función `binario_a_decimal()` para convertirlo a su equivalente en decimal. El resultado se muestra en pantalla.

#### VII-E. Depuración y prueba

Durante el proceso de desarrollo y posterior ejecución del programa, pueden surgir errores o problemas. Para depurar el programa, se pueden utilizar técnicas como la impresión de mensajes de depuración, el uso de herramientas de depuración y la revisión completa del código en busca de posibles errores.

Entrada	Salida Esperada	Salida Obtenida
1011	11	11
1101	13	13
10001	17	17
1111	15	15

### VIII. TABLA DE VERDAD CON EXPRESIÓN BOOLEANA

#### VIII-A. Definición

Los teoremas booleanos son reglas o propiedades que ayudan a simplificar expresiones y resolver problemas con lógica, los operadores booleanos son And, Or y Not. [6]

#### VIII-B. Descripción de el problema

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

#### VIII-C. Diseño de la solución

Con ayuda de los teoremas booleanos el programa deberá pedir al usuario ingresar un número de bits para crear las salidas de la tabla de verdad al igual que la expresión booleana basándose en la misma.

#### VIII-D. Desarrollo de la solución

Para la solución del problema se creó un programa en el cual se le pide al usuario que proporcione un número de bits y a su vez crear una tabla con el número de bits dado y así crear una expresión booleana con los resultados obtenidos.

```

1 import java.util.Scanner;
2 import java.util.HashSet;
3 import java.util.Set;
4
5 public class tabla_de_verdad {
6     public static void main(String[] args) {
7         Scanner scanner = new Scanner(System.in); //
            Se crea un objeto Scanner para la
            entrada del usuario.
8         Set<Integer> filasCambiar = new HashSet<>();
            // Se crea un conjunto para almacenar
            las filas que el usuario desea cambiar.
9
10        System.out.print("Ingresa el numero de bits
            "); // Se solicita al usuario el nmero
            de bits.

```

```

int numBits = scanner.nextInt(); // Se lee el
    nmero de bits proporcionado por el
    usuario.

```

```

// IMPRIMIR TABLA
for (int i = 0; i < numBits; i++) {
    System.out.print((char) ('A' + i) + "\t");
    // Se imprimen los encabezados de las
    variables (A, B, C, etc.).
}

```

```

System.out.println("Resultado"); // Se
    imprime el encabezado de la columna
    "Resultado".

```

```

for (int i = 0; i < Math.pow(2, numBits);
    i++) {
    imprimirFilaTabla(i, numBits); // Se
    imprime una fila de la tabla de
    verdad original.
    System.out.println("0"); // Se imprime el
    resultado, que es siempre 0 en la
    tabla original.
}

```

```

// IMPRIMIR TABLA

```

```

while (true) { // Bucle para Cambiar Bits
    System.out.print("Ingresa el nmero de la
        fila que desea cambiar el resultado a
        1 (1-" + (int) Math.pow(2, numBits) +
        ") si desea finalizar ingresar 0: ");
    int fila = scanner.nextInt(); // Se lee la
        fila que el usuario desea cambiar.

```

```

    if (fila == 0) { break; } // Si el usuario
        ingresa 0, se sale del bucle.

```

```

    if (fila < 1 || fila > Math.pow(2,
        numBits)) { // Se valida que la fila
        est en el rango vlido.
        System.out.println("Nmero de fila
            invlido. Debe estar entre 1 y " +
            (int) Math.pow(2, numBits) + ".");
        continue; // Se vuelve al inicio del
            bucle si la fila no es vlida.
    }

```

```

    filasCambiar.add(fila); // Se agrega la
        fila al conjunto de filas a cambiar.
    imprimirTablaActualizada(filasCambiar,
        numBits); // Se imprime la tabla
        actualizada.
}

```

```

// Resultado BOOLEANO
System.out.println("Expresiones booleanas al
    finalizar:");
filasCambiar.forEach(fila ->
    System.out.println(generarExpresionBooleana(fila,
        numBits))); // Se imprimen las
        expresiones booleanas de las filas
        modificadas.
System.out.println("Expresin booleana final:
    " + generarExpresionFinal(filasCambiar,
        numBits)); // Se imprime la expresin
        booleana final.
// Resultado BOOLEANO
}

```

```

private static void imprimirFilaTabla(int valor,
    int numBits) {
    for (int j = numBits - 1; j >= 0; j--) {
        System.out.print(((valor >> j) & 1) +
            "\t"); // Se imprime cada bit de la
            fila en formato binario.
    }
}

```



```

50     }
51 }
52
53 private static void
    imprimirTablaActualizada(Set<Integer>
        filasCambiar, int numBits) {
54     System.out.println("Tabla de verdad
        actualizada:");
55     for (int i = 0; i < Math.pow(2, numBits);
        i++) {
56         imprimirFilaTabla(i, numBits);
57         boolean resultado = false;
58         if (filasCambiar.contains(i + 1)) {
59             resultado = true;
60         }
61         System.out.println(resultado ? "1" : "0");
        // Se imprime el resultado (1 o 0) de
        acuerdo a las filas cambiadas.
62     }
63 }
64
65 // Funcin para generar expresiones booleanas
66 private static String
    generarExpresionBooleana(int fila, int
        numBits) {
67     StringBuilder expression = new StringBuilder();
68     for (int i = 0; i < numBits; i++) {
69         expression.append(((fila - 1) >> (numBits -
            1 - i) & 1) == 1 ? (char) ('A' + i) :
            (char) ('A' + i) + "'");
70         if (i < numBits - 1) {
71             expression.append(" + ");
72         }
73     }
74     return expression.toString(); // Se genera la
        expresin booleana para una fila
        especifica.
75 }
76
77 private static String
    generarExpresionFinal(Set<Integer>
        filasCambiar, int numBits) {
78     StringBuilder expresionesConcatenadas = new
        StringBuilder();
79     filasCambiar.forEach(fila -> {
80         String expression =
            generarExpresionBooleana(fila,
                numBits);
81         if (expresionesConcatenadas.length() > 0) {
82             expresionesConcatenadas.append(" + ");
83         }
84         expresionesConcatenadas.append
            ("(").append(expression).append(")");
85     });
86     return expresionesConcatenadas.toString();
87     // Se genera la expresin booleana final
88     concatenando las expresiones de las
        filas modificadas.
89 }
90 }
91 }
92 }

```

### VIII-E. Depuracion y pruebas

En el cuadro 3 se muestran los resultados arrojados al compilar el código.

No	A	B	C	RESULTADO
1	0	0	0	1
2	0	0	1	1
3	0	1	0	0
4	0	1	1	1
5	1	0	0	0
6	1	0	1	1
7	1	1	0	1
8	1	1	1	1

Cuadro IV: Tabla de corridas.

## IX. CONCLUSIÓN

Para concluir, a lo largo de este periodo de estudio en el que hemos abordado los temas relacionados con la resolución de problemas en ingeniería, hemos llevado a cabo un exhaustivo análisis, un diseño creativo y una implementación efectiva de diversas soluciones. Como resultado, hemos logrado superar con éxito los desafíos planteados, lo cual resalta la relevancia de la colaboración interdisciplinaria y el amplio uso de herramientas tecnológicas innovadoras como apoyo en la resolución de problemas.

Además, hemos reafirmado la importancia de la ingeniería como medio para abordar los diferentes desafíos de la vida real, poniendo especial énfasis en la consideración de aspectos éticos, sociales y ambientales. Este enfoque integral nos ha enseñado la importancia de la creatividad, la colaboración y la aplicación práctica de los conocimientos adquiridos.

## REFERENCIAS

- [1] Khan Academy | Free online courses, lessons practice. (s. f.). Khan Academy. <https://www.khanacademy.org/>
- [2] La fórmula general - Unidad de Apoyo Para el Aprendizaje. (2023). Unam.mx. [https://uapas2.bunam.unam.mx/matematicas/la\\_formula\\_general/](https://uapas2.bunam.unam.mx/matematicas/la_formula_general/)
- [3] Fórmula de la distancia (video). (n.d.). Khan Academy. <https://es.khanacademy.org/math/geometry/hs-geo-analytic-geometry/hs-geo-distance-and-midpoints/v/distance-formula>
- [4] Convert integer into byte array (Java) ButtercupButtercup 2 et al. <https://stackoverflow.com/questions/1936857/convert-integer-into-byte-array-java>
- [5] Ramos, A. Á., & Gálvez, J. M. B. (2021, March 1). Cambio de signo de un número en binario. Portal Académico Del CCH. <https://portalacademico.cch.unam.mx/cibernetica1/sistemas-de-numeracion/cambio-de-signo>
- [6] Desarrollada, & Boole, G. (n.d.). ÁLGEBRA BOOLEANA. [https://nlaredo.tecnm.mx/takeyas/apuntes/matematicas\\_discretas/apuntes/Algebra\\_Booleana.pdf](https://nlaredo.tecnm.mx/takeyas/apuntes/matematicas_discretas/apuntes/Algebra_Booleana.pdf)

## X. BIOGRAFÍAS

### Santiago Reséndiz Melanie

Es una estudiante del ITSOEH en la Ingeniería en Tecnologías de la Información y Comunicaciones tiene una pasión por las aplicaciones. Nacida en Actopan, Hgo. Fue criada en una comunidad llamada Dajiedhit, desde muy temprana edad mostró un gran interés por la computación, sobre todo en el diseño uno de sus principales objetivos es acabar la carrera y tener un trabajo en una empresa como Huawei o Google, una de sus principales pasiones son el anime, la música pop, el volleyball y la programación en especial el área de aplicaciones para android y iOS.

[GitHub de Melanie Santiago Reséndiz](#)

### Hernández Monroy Ricardo

Es una persona muy audaz y hábil en deportes, interesado por los videojuegos y el anime. Actualmente, se crió y nació en las Presas de la comunidad de Tezontepec de Aldama, Hidalgo. Nació el 6 de noviembre de 2005, teniendo ahora 18 años. Es estudiante en el ITSOEH, cursando la carrera de

Tecnologías de la Información y Comunicación, con mucho interés hacia las tecnologías que se enseñan en la carrera.

[GitHub de Ricardo Hernández Monroy](#)

### Fuentes Pérez Bryan

Es estudiante del Instituto Tecnológico Superior Del Occidente Del Estado De Hidalgo(ITSOEH), que ingreso en la carrera de Ingenieria en Tecnologias de la Informacion y Comunicacion (ITICs), por el gusto y de las habilidades que puede adquirir en su estancia que dure en la escuela y poder aplicar eso en problemas relacionados con el mundo cotidiano , para la amplia gama de beneficios diversos aspectos de nuestras vidas.

[GitHub de Bryan Fuentes Perez](#)

### Salazar Guerrero Diego

Es un estudiante del Instituto Tecnológico Superior del Occidente del Estado de Hidalgo(ITSOEH), que ingreso a la carrera de Ingeniería en Tecnologías de la Información y Comunicaciones con una pasión hacia las telecomunicaciones, la tecnología y los video juegos. Nacido en el municipio de Progreso de Obregón, Hgo. y criado en el municipio de San Salvador, Hgo. El objetivo principal de Diego es acabar sus estudios en en ITICs, con especialización en telecomunicaciones. Aspira a trabajar en la NASA para contribuir al desarrollo y avance de México.

[GitHub de Diego Salazar Guerrero](#)

### Cruz Granados Dulce Marina

Es una estudiante del Instituto Tecnológico Superior del Occidente del Estado de Hidalgo(ITSOEH), que ingreso a la carrera de Ingeniería en Tecnologías de la Información y Comunicaciones interesada en el area de programacion. Nacida en el municipio de Tula de Allende, Hgo. y criada en Santa Maria Nativitas Atengo Municipio de Tezontepec de Aldama, Hgo.

[GitHub de Dulce Marina Cruz Granados](#)