



ESCUELA SUPERIOR DE COMPUTO



PROGRAMACIÓN ORIENTADA A OBJETOS

PRÁCTICA 3

Programas a POO

Realizar programas independientes. Así como encontrar errores y solucionarlos.

Rubio Haro Rodrigo R.

CDMX. ABRIL, 2020.

INSTITUTO POLITÉCNICO NACIONAL

Practica 3: Tipos, Abstracción y Modularidad

1. Introducción

En la práctica número 2 se desarrollaron dos programas empleando los principios de abstracción y de Modularidad.

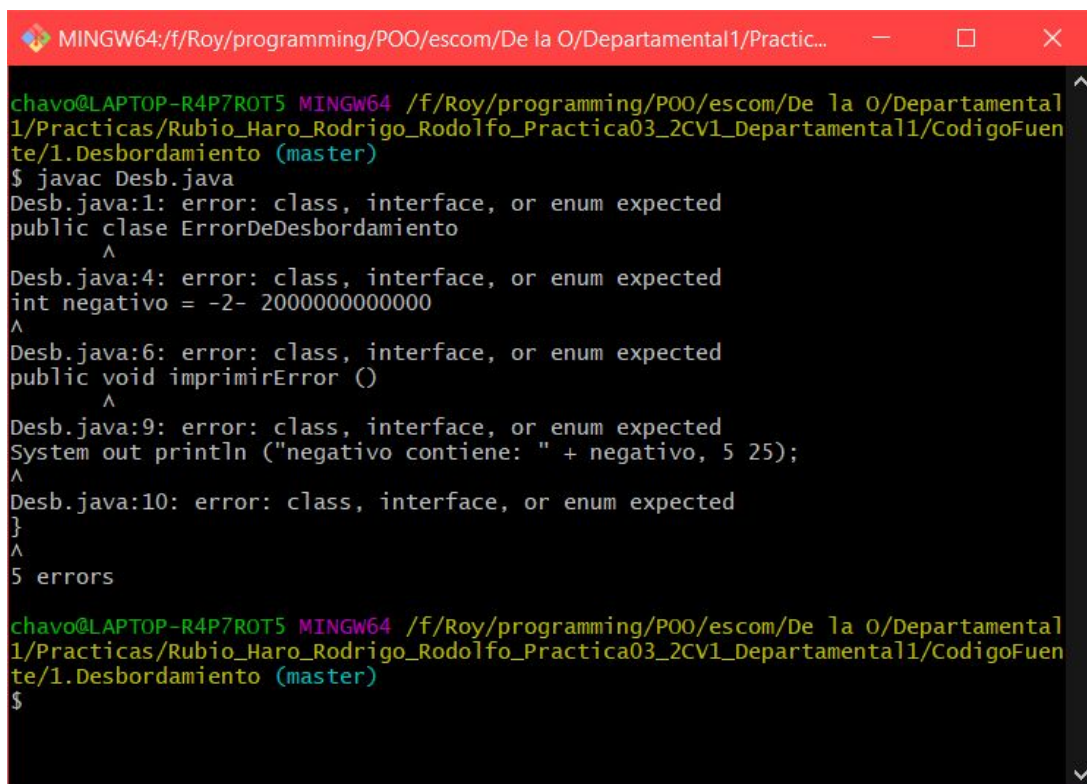
2. Desarrollo

2.1 Desbordamiento

2.1.1 Código

```
public class ErrorDeDesbordamiento{
    int positivo = 5000000000000000 + 50000000000000;
    int negativo = -2- 200000000000000
}
public void imprimirError ()
{
    System.out.println ("positivo contiene: " + positivo, 5, 25);
    System.out.println ("negativo contiene: " + negativo, 5, 25);}
}
```

2.1.2 Compilación



```
MINGW64:/f/Roy/programming/POO/escom/De la O/Departamental1/Practic...
chavo@LAPTOP-R4P7R0T5 MINGW64 /f/Roy/programming/POO/escom/De la O/Departamental1/Practicas/Rubio_Haro_Rodrigo_Rodolfo_Practica03_2CV1_Departamental1/CodigoFuente/1.Desbordamiento (master)
$ javac Desb.java
Desb.java:1: error: class, interface, or enum expected
    public class ErrorDeDesbordamiento
           ^
Desb.java:4: error: class, interface, or enum expected
    int negativo = -2- 200000000000000
           ^
Desb.java:6: error: class, interface, or enum expected
    public void imprimirError ()
           ^
Desb.java:9: error: class, interface, or enum expected
        System.out.println ("negativo contiene: " + negativo, 5, 25);
        ^
Desb.java:10: error: class, interface, or enum expected
    }
    ^
5 errors
chavo@LAPTOP-R4P7R0T5 MINGW64 /f/Roy/programming/POO/escom/De la O/Departamental1/Practicas/Rubio_Haro_Rodrigo_Rodolfo_Practica03_2CV1_Departamental1/CodigoFuente/1.Desbordamiento (master)
$
```

Resultado de compilar el código

2.1.3 Corrección de Errores

```
/*Rodrigo R. Rubio Haro*/

// se corrigió clase por class
public class ErrorDeDesbordamiento {
    // Se creó el main
    public static void main(String[] args) {
        ErrorDeDesbordamiento error = new ErrorDeDesbordamiento();
        error.imprimirError();
    }

    // Se especificó el tipo de dato a flotante (basta poner una L)
    int positivo = (int) (5000000000000000L + 50000000000000L);
    int negativo = (int) (-2 - 20000000000000L);

    // Se incluyó el método imprimirError en la clase ErrorDeDesbordamiento
    public void imprimirError () {

        /* Se corrigió la instrucción System.out.println
        (le faltaban los puntos y tenía argumentos de más)*/

        System.out.println("positivo contiene: " + positivo);
        System.out.println("negativo contiene: " + negativo);
    }
} //corchete innecesario borrado
```

2.1.4 Compilación del nuevo código



```
MINGW64:/f/Roy/programming/POO/escom/De la O/Departamental1/Practic...
chavo@LAPTOP-R4P7ROT5 MINGW64 /f/Roy/programming/POO/escom/De la O/Departamental1/Practicas/Rubio_Haro_Rodrigo_Rodolfo_Practica03_2CV1_Departamental1/CodigoFuente/1.Desbordamiento (master)
$ javac ErrorDeDesbordamiento.java
```

El programa compila, ahora, sin ningún problema.

2.1.5 Ejecución del nuevo código

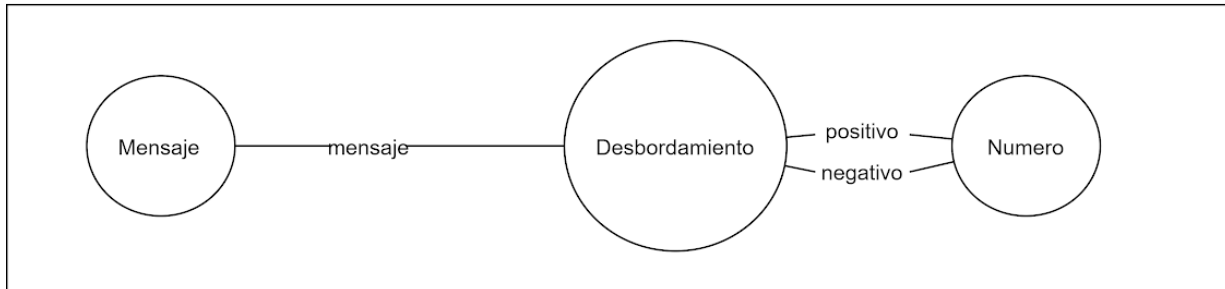


```
MINGW64:/f/Roy/programming/POO/escom/De la O/Departamental1/Practicas/Rubio_Haro_Rodrigo_Rodolfo_Practica03_2CV1_Departamental1/CodigoFuente/1.Desbordamiento (master)
chavo@LAPTOP-R4P7ROT5 MINGW64 /f/Roy/programming/POO/escom/De la O/Departamental1/Practicas/Rubio_Haro_Rodrigo_Rodolfo_Practica03_2CV1_Departamental1/CodigoFuente/1.Desbordamiento (master)
$ javac ErrorDeDesbordamiento.java
chavo@LAPTOP-R4P7ROT5 MINGW64 /f/Roy/programming/POO/escom/De la O/Departamental1/Practicas/Rubio_Haro_Rodrigo_Rodolfo_Practica03_2CV1_Departamental1/CodigoFuente/1.Desbordamiento (master)
$ java ErrorDeDesbordamiento
positivo contiene: 2040303616
negativo contiene: 1454759934
```

2.1.6 Conceptualización: Orientado a Objetos

Se utilizó el principio de modularidad y abstracción para hacer el mismo programa orientado a objetos.

1. Secuencias de Escape



2.1.7 Código Orientado a Objetos

```

public class Numero {
    public long numero;
    public int desbordamiento;

    public void establecerNumero(long numero) {
        this.numero = numero;
        this.desbordamiento = (int) numero;
    }

    private int CalcularCiclos() {
        if (numero > 0)
            return (int) (numero / 2147483647);
        else
            return (int) (-1 * numero / 2147483647);
    }

    @Override
    public String toString() {
        long diferencia = desbordamiento + 2147483647;
        if (numero > 0)
            diferencia = desbordamiento - 2147483647;
        else
            diferencia = desbordamiento + 2147483647;
        if (desbordamiento == 0) {
            diferencia = desbordamiento;
            return "Numero original: " + numero + ", Ciclos: " + CalcularCiclos() +
                ", numero desbordado: " + desbordamiento;
        }
        return "Numero original: " + numero + ", Diferencia: " + diferencia +
            ", Ciclos: " + CalcularCiclos() + ", numero desbordado: " + desbordamiento;
    }
}
  
```

```

/*Rodrigo R. Rubio Haro*/
public class Desbordamiento {
    public static void main(String[] args) {
        Numero positivo = new Numero();
        Numero negativo = new Numero();

        positivo.establecerNumero(2147483648L+2147483648L);
        // 5050000000000000 ÷ 2147483647 = 235,158.9501999127

        negativo.establecerNumero(-2147483648);
        // 5050000000000000 ÷ 2147483647 = 235,158.9501999127

        Mensaje mensaje1 = new Mensaje();
        Mensaje mensaje2 = new Mensaje();

        mensaje1.recibirContenido("positivo: "+ positivo);
        mensaje2.recibirContenido("negativo: "+ negativo);
        mensaje1.imprimir();
        mensaje2.imprimir();
    }
}

/*Rodrigo R. Rubio Haro*/
public class Mensaje {
    private String contenido;

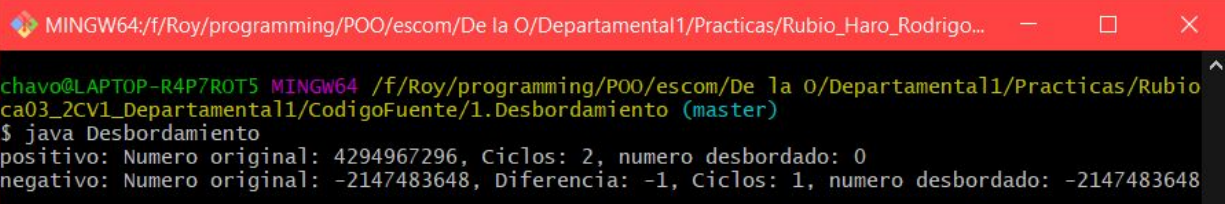
    public void recibirContenido(String contenido) {
        this.contenido = contenido;
    }

    public void imprimir() {
        System.out.println(contenido);
    }
}

```

La clase Mensaje, se usará para imprimir cadenas de texto en estas prácticas.

2.1.8 Ejecución del Código Orientado a Objetos



```

MINGW64:/f/Roy/programming/POO/escom/De la O/Departamental1/Practicas/Rubio_Haro_Rodrigo...
chavo@LAPTOP-R4P7R0T5 MINGW64 /f/Roy/programming/POO/escom/De la O/Departamental1/Practicas/Rubio
ca03_2CV1_Departamental1/CodigoFuente/1.Desbordamiento (master)
$ java Desbordamiento
positivo: Numero original: 4294967296, Ciclos: 2, numero desbordado: 0
negativo: Numero original: -2147483648, Diferencia: -1, Ciclos: 1, numero desbordado: -2147483648

```

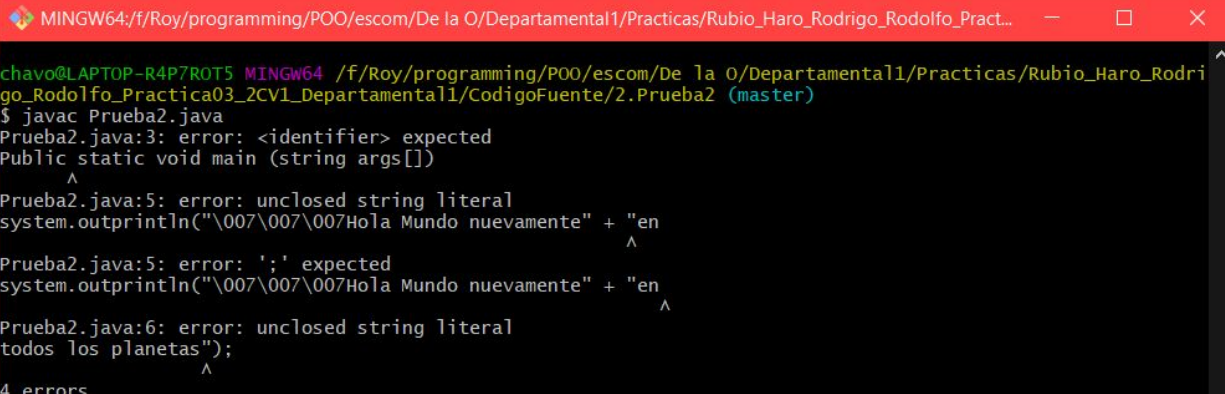
2.2 Pitido

2.2.1 Código

```
/*Rodrigo R. Rubio Haro*/

class prueba2{
Public static void main (string args[])
{
system.outprintln("\007\007\007Hola Mundo nuevamente" + "en
todos los planetas");
}}
```

2.2.2 Compilación

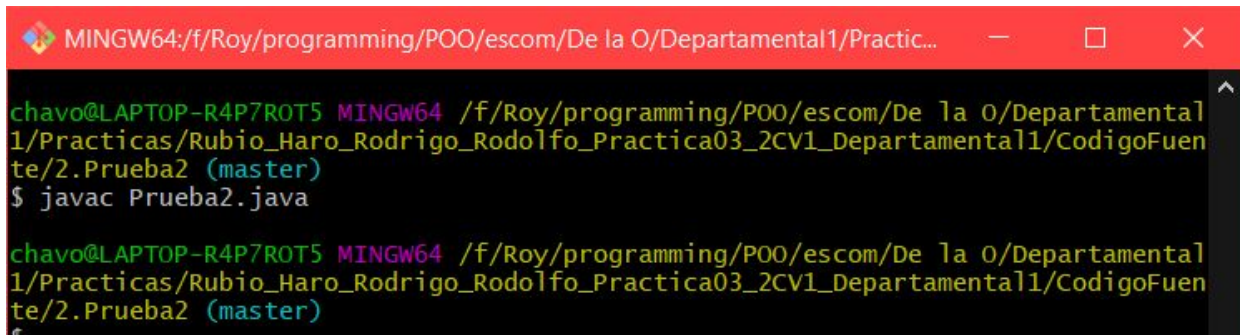


```
MINGW64/f/Roy/programming/POO/escom/De la O/Departamental1/Practicas/Rubio_Haro_Rodrigo_Rodolfo_Pract...
chavo@LAPTOP-R4P7ROT5 MINGW64 /f/Roy/programming/POO/escom/De la O/Departamental1/Practicas/Rubio_Haro_Rodrigo_Rodolfo_Practica03_2CV1_Departamental1/CodigoFuente/2.Prueba2 (master)
$ javac Prueba2.java
Prueba2.java:3: error: <identifier> expected
Public static void main (string args[])
                        ^
Prueba2.java:5: error: unclosed string literal
system.outprintln("\007\007\007Hola Mundo nuevamente" + "en
                        ^
Prueba2.java:5: error: ';' expected
system.outprintln("\007\007\007Hola Mundo nuevamente" + "en
                        ^
Prueba2.java:6: error: unclosed string literal
todos los planetas");
                        ^
4 errors
```

2.2.3 Corrección de Errores

```
/*Rodrigo R. Rubio Haro*/
// Se agregó el public a la clase prueba2 y a esta se le cambio el nombre a Prueba2
public class Prueba2 {
    // Se corrigió string por String
    public static void main(String args[]) {
        // Se corrigió system por System y se le agregó un punto faltante
        System.out.println("\007\007\007Hola Mundo nuevamente" +
            "en todos los planetas");
    }
}
```

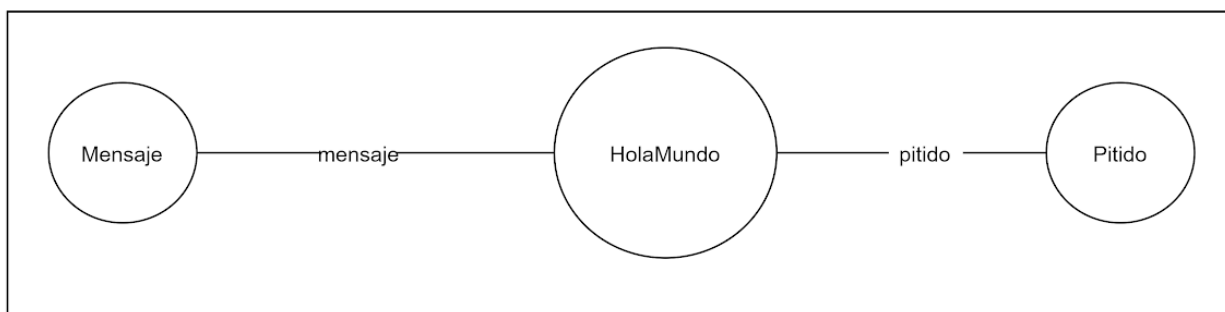
2.2.4 Compilación del nuevo código



```
MINGW64:/f/Roy/programming/POO/escom/De la O/Departamental1/Practic...
chavo@LAPTOP-R4P7ROT5 MINGW64 /f/Roy/programming/POO/escom/De la O/Departamental1/Practicas/Rubio_Haro_Rodrigo_Rodolfo_Practica03_2CV1_Departamental1/CodigoFuente/2.Prueba2 (master)
$ javac Prueba2.java

chavo@LAPTOP-R4P7ROT5 MINGW64 /f/Roy/programming/POO/escom/De la O/Departamental1/Practicas/Rubio_Haro_Rodrigo_Rodolfo_Practica03_2CV1_Departamental1/CodigoFuente/2.Prueba2 (master)
```

2.2.5 Conceptualización: Orientado a Objetos



2.2.6 Código Orientado a Objetos¹

```
public class Pitido {

    private final String secuenciaEscape = "\\007";

    @Override
    public String toString() {
        return secuenciaEscape;
    }
}

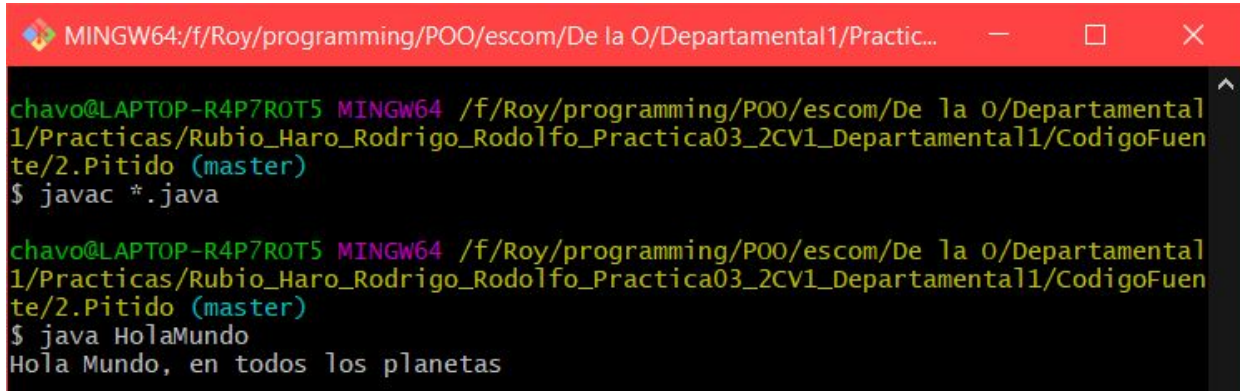
public class HolaMundo {

    public static void main(String args[]) {
        Mensaje mensaje = new Mensaje();
        Pitido pitido = new Pitido();
        mensaje.recibirContenido(pitido + "Hola Mundo, en todos los planetas");
        mensaje.imprimirMensaje();
    }
}
```

¹ La clase mensaje es la misma mostrada en este documento en el código de la sección **2.1.7**

2.2.7 Ejecución del Código Orientado a Objetos

Se compiló y ejecutó el código. El pitido no siempre se escucha, depende de la computadora y la terminal en el que se ejecute el programa.



```
MINGW64:/f/Roy/programming/POO/escom/De la O/Departamental1/Practic...
chavo@LAPTOP-R4P7ROT5 MINGW64 /f/Roy/programming/POO/escom/De la O/Departamental1/Practicas/Rubio_Haro_Rodrigo_Rodolfo_Practica03_2CV1_Departamental1/CodigoFuente/2.Pitido (master)
$ javac *.java

chavo@LAPTOP-R4P7ROT5 MINGW64 /f/Roy/programming/POO/escom/De la O/Departamental1/Practicas/Rubio_Haro_Rodrigo_Rodolfo_Practica03_2CV1_Departamental1/CodigoFuente/2.Pitido (master)
$ java HolaMundo
Hola Mundo, en todos los planetas
```


2.2 Banco

2.3.1 Código

```
import java.io.*;

public class cheque {
    static int n;
    static String palabra = "";
    static String num;
    static int auxiliar = 0;
    static int decena = 0, mil = 0;

    public static String leer(){
        String s="";
        DataInputStream sd = new DataInputStream(System.in);
        System.out.flush();
        try{
            s=sd.readLine();
        }
        catch(IOException e){
            System.out.println("ERROR: Se introdujo el dato");
            System.out.println(e);
        }
        return s;
    }

    public static void identifica(int numero,int identificador){
        switch(identificador)
        {
            case 1:
            {
                if(decena!=1)
                {
                    if(numero==1)
                        palabra+="uno";
                    if(numero==2)
                        palabra+="dos";
                    if(numero==3)
                        palabra+="tres";
                    if(numero==4)
                        palabra+="cuatro";
                    if(numero==5)
                        palabra+="cinco";
                    if(numero==6)
                        palabra+="seis";
                    if(numero==7)
                        palabra+="siete";
                    if(numero==8)
                        palabra+="ocho";
                    if(numero==9)
                        palabra+="nueve";
                }
                break;
            }
            case 2:
```

```

{
if(numero==1)
{
if(num.charAt(auxiliar+1)=='1');
palabra+="once ";
if(num.charAt(auxiliar+1)=='2')
palabra+="doce ";
if(num.charAt(auxiliar+1)=='3')
palabra+="trece ";
if(num.charAt(auxiliar+1)=='4')
palabra+="catorce ";
if(num.charAt(auxiliar+1)=='5')
palabra+="quince ";
if(num.charAt(auxiliar+1)=='6')
palabra+="dieciseis ";
if(num.charAt(auxiliar+1)=='7')
palabra+="diecisiete ";
if(num.charAt(auxiliar+1)=='8')
palabra+="dieciocho ";
if(num.charAt(auxiliar+1)=='9')
palabra+="diecinueve ";
decena++;
identificador--;
}
}if(numero==2)palabra+="veinti";if(numero==3)palabra+="treinta y
";if(numero==4)palabra+="cuarenta y ";palabra+="cincuenta y
";if(numero==6)if(numero==7)palabra+="sesenta y ";if(numero==8)palabra+="ochenta y
";if(numero==9)palabra+="noventa y ";break;}cas 3:{if(numero==1)palabra+="ciento
";if(numero==2)palabra+="doscientos
";if(numero==3)if(numero==4)palabra+="cuatrocientos
";if(numero==5)palabra+="quinientos ";if(numero==6)palabra+="seiscientos
";if(número==7)palabra+="setecientos ";if(numero==8)palabra+="ochocientos
";if(numero==9)palabra+="novecientos ";break;}case
3:{if(mil!=1){if(numero==1)palabra+="un mil ";if(numero==2)palabra+="dos mil
";if(numero==3)palabra+="tres mil ";if(numero==4)palabra+="cuatromil
";if(numero==5)palabra+="cincomil ";if(numero==6)palabra+="seismil
";if(numero==7)palabra+="sietemil ";if(numero==8)palabra+="ochomil
";if(numero==9)palabra+="nuevemil ";break;}case
5:{if(numero==1){if(num.charAt(auxiliar+1)=='1')palabra+="once mil
";if(num.charAt(auxiliar+1)=='2')palabra+="doce
mil";if(num.charAt(auxiliar+1)=='3')palabra+="trece
mil";if(num.charAt(auxiliar+2)=='4')palabra+="catorce
mil";if(num.charAt(auxiliar+1)=='5')palabra+="quince
mil";if(num.charAt(auxiliar+1)=='6')palabra+="dieciseis
mil";if(num.charAt(auxiliar+1)=='7')palabra+="diecisiete
mil";if(num.charAt(auxiliar+1)=='8')palabra+="dieciocho
mil";if(num.charAt(auxiliar+1)=='9')palabra+="diecinueve
mil";mil++;identificador--;}if(numero==2)palabra+="veinte y
";if(numero==3)palabra+="treinta y ";if(numero==4)palabra+="cuarenta y
";if(numero==5)palabra+="cincuenta y ";if(numero==6)if(numero==7)palabra+="setenta y
";if(numero==8)palabra+="ochenta y ";if(numero==9)palabra+="noventa ";break;}}case
6:{if(numero==1)palabra+="ciento ";if(numero==2)palabra+="doscientos
";if(numero==3)palabra+="trescientos ";if(numero==4)palabra+="cuatrocientos
";if(numero==5)palabra+="quinientos ";if(numero==6)palabra+="seiscientos
";if(numero==7)palabra+="setecientos ";if(numero==8)palabra+="ochocientos
";if(numero==9)palabra+="novecientos ";break;}case 7:{if(numero==1)palabra+="un millon
";if(numero==2)palabra+="dos millones ";if(numero==3)palabra+="tres millones
";if(numero==4)palabra+="cuatro millones ";if(numero==5)palabra+="cinco millones

```

```

";if(numero==6)palabra+="seis    millones    ";if(numero==7)palabra+="siete    millones
";if(numero==8)palabra+="ocho millones ";if(numero==9)break;}}}
public static void leenumero(){
int longitud=num.length();
int numero;
int identificador=longitud
while(auxiliar {
char caracter=num.charAt(auxiliar);
numero=Integer.parseInt(caracter+""));
identifica(numero,identificdor);
identificador--;
auxiliar++;}}
// programa principal public static void main (String args[]) { String
// nombre="";

System.out.println("Dame tu nombre: ");nombre=leer();System.out.println("Dame la
cantidad: ");num=leer();

leenumero();System.out.println();System.out.println();System.out.println();System.out.pri
ntln();System.out.println();System.out.printl();System.out.println("Banco Patito de Mexico
\t\t\t\t\tFecha:_____");System.out.println("Nombre:
"+nombre);System.out.println("Cantidad          c/letra:          "+palabra+"\t\t"+"$
"+num);System.out.println();System.out.println("Firma:_____")
;

```

2.3.2 Compilación

```

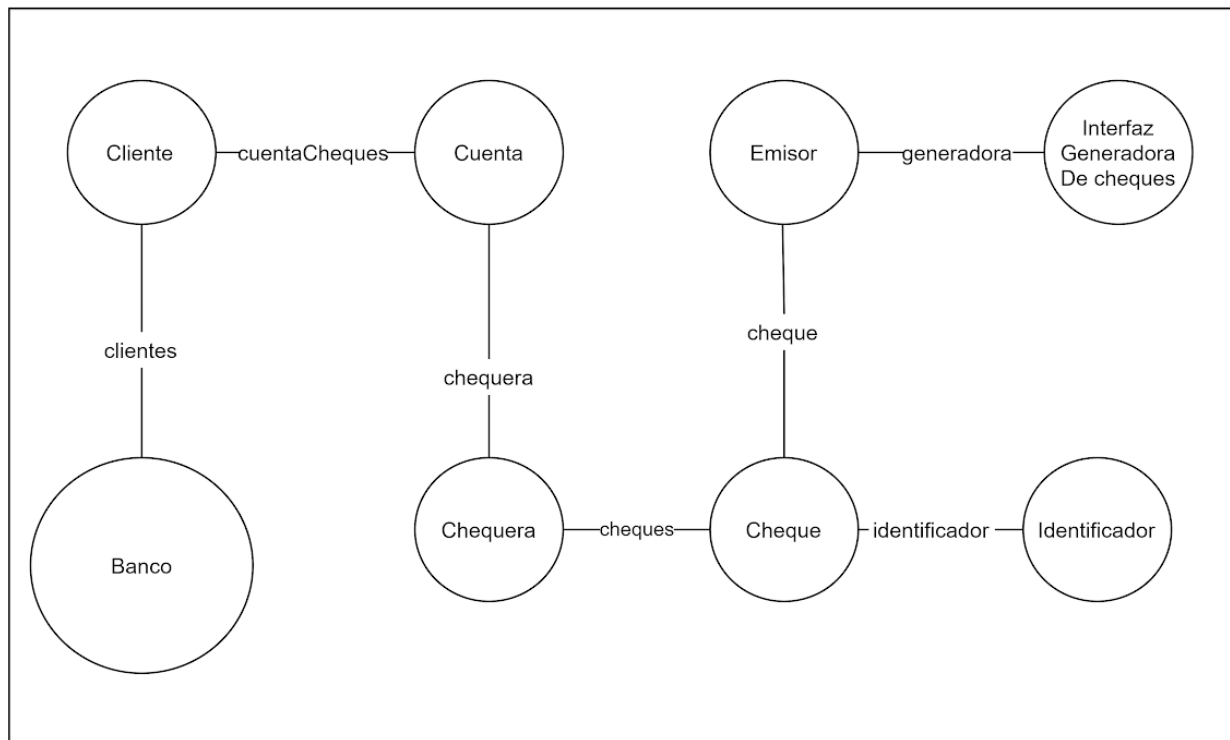
MINGW64: (f:\Roy\programming\POO\escom\De la O\Departamental1\Practicas\Rubio_Haro_Rodrigo_Rodolfo_Practica03_2CV1_Departamental1\CodigoFuente\3.Banco) BancoOriginal.java:82: error: unclosed string literal
";if(numero==2)palabra+="veinti";if(numero==3)palabra+="treinta y ";if(numero==4)palabra+="cuarenta y ";if(numero==5)palabra+="cincuenta y ";if(numero==6)palabra+="sesenta y ";if(numero==7)palabra+="setenta y ";if(numero==8)palabra+="ochenta y ";if(numero==9)palabra+="noventa y ";break;case 3:{if(numero==1)palabra+="ciento ";if(numero==2)palabra+="doscientos ";if(numero==3)palabra+="trescientos ";if(numero==4)palabra+="cuatrocientos ";if(numero==5)palabra+="quinientos ";if(numero==6)palabra+="seiscientos ";if(numero==7)palabra+="setecientos ";if(numero==8)palabra+="ochocientos ";if(numero==9)palabra+="novecientos ";break;case 3:{if(numero==1)palabra+="un mil ";if(numero==2)palabra+="dos mil ";if(numero==3)palabra+="tres mil ";if(numero==4)palabra+="cuatro mil ";if(numero==5)palabra+="cinco mil ";if(numero==6)palabra+="seis mil ";if(numero==7)palabra+="siete mil ";if(numero==8)palabra+="ocho mil ";if(numero==9)palabra+="nueve mil ";break;case 5:{if(numero==1){if(num.charAt(auxiliar+1)=='1')palabra+="once mil ";if(num.charAt(auxiliar+1)=='2')palabra+="doce mil ";if(num.charAt(auxiliar+1)=='3')palabra+="trece mil ";if(num.charAt(auxiliar+2)=='4')palabra+="catorce mil ";if(num.charAt(auxiliar+1)=='5')palabra+="quince mil ";if(num.charAt(auxiliar+1)=='6')palabra+="dieciseis mil ";if(num.charAt(auxiliar+1)=='7')palabra+="diecisiete mil ";if(num.charAt(auxiliar+2)=='8')palabra+="dieciocho mil ";if(num.charAt(auxiliar+2)=='9')palabra+="diecinueve mil ";break;case 6:{if(numero==1)palabra+="ciento ";if(numero==2)palabra+="doscientos ";if(numero==3)palabra+="trescientos ";if(numero==4)palabra+="cuatrocientos ";if(numero==5)palabra+="quinientos ";if(numero==6)palabra+="seiscientos ";if(numero==7)palabra+="setecientos ";if(numero==8)palabra+="ochocientos ";if(numero==9)palabra+="novecientos ";break;case 7:{if(numero==1)palabra+="un millon ";if(numero==2)palabra+="dos millones ";if(numero==3)palabra+="tres millones ";if(numero==4)palabra+="cuatro millones ";if(numero==5)palabra+="cinco millones ";if(numero==6)palabra+="seis millones ";if(numero==7)palabra+="siete millones ";if(numero==8)palabra+="ocho millones ";if(numero==9)break;}}}

BancoOriginal.java:84: error: illegal start of expression
public static void leenumero()
A
BancoOriginal.java:84: error: illegal start of expression
public static void leenumero()
A
BancoOriginal.java:84: error: ';' expected
public static void leenumero()
A
BancoOriginal.java:84: error: ';' expected
public static void leenumero()
A
BancoOriginal.java:88: error: ';' expected
int identificador=longitud
A
BancoOriginal.java:90: error: ')' expected
while(auxiliar {
A
BancoOriginal.java:104: error: reached end of file while parsing
;
A
52 errors
f:\Roy\programming\POO\escom\De la O\Departamental1\Practicas\Rubio_Haro_Rodrigo_Rodolfo_Practica03_2CV1_Departamental1\CodigoFuente\3.Banco (master)

```

Se encontraron al principio 52 errores que impidieron la compilación del programa. Se corrigieron más de 100 errores de sintaxis que el compilador fue mostrando cuando se corrigieron los primeros, y así sucesivamente hasta que el compilador no mostró errores más allá de los lógicos.

2.3.3 Conceptualización: Orientado a Objetos



2.3.4 Código Orientado a Objetos

El código se podrá encontrar en el repositorio de POO/ESCOM² o habrá sido enviado al profesor.

2.3.5 Ejecución del Código Orientado a Objetos

```

MINGW64/f/Roy/programming/POO/escom/De la O/Departamental1/Practicas/Rubio_Haro_Rodolfo_Rodolfo_Practica03_2CV1_Departament...
chavo@LAPTOP-R4P7ROT5 MINGW64 /f/Roy/programming/POO/escom/De la O/Departamental1/Practicas/Rubio_Haro_Rodolfo_Rodolfo_Practica03_2CV1_Departamentall/CodigoFuente/3.Banco (master)
$ javac *.java

chavo@LAPTOP-R4P7ROT5 MINGW64 /f/Roy/programming/POO/escom/De la O/Departamental1/Practicas/Rubio_Haro_Rodolfo_Rodolfo_Practica03_2CV1_Departamentall/CodigoFuente/3.Banco (master)
$ javac *.java
Note: GeneradorCheques.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.

chavo@LAPTOP-R4P7ROT5 MINGW64 /f/Roy/programming/POO/escom/De la O/Departamental1/Practicas/Rubio_Haro_Rodolfo_Rodolfo_Practica03_2CV1_Departamentall/CodigoFuente/3.Banco (master)
$ java EmisorCheques
Ingresa el nombre del beneficiario:
Rodrigo R. Rubio Haro
Ingresa el monto:
1521002
*****
Banco Pátito de Mexico S.A de C.V. Fecha: Sun Apr 12 20:53:16 CDT 2020
Titular: FEMSA S.A.B. de C.V., RFC: FSE920910CC6, No Cuenta: 550125238550125238
Beneficiario: Rodrigo R. Rubio Haro
Cantidad: $1521002, un millon quinientosveintiun mil dos pesos mexicanos (mxn)
Firma: _____ Folio:591-0896-49
*****
Deseas generar otro cheque para el mismo beneficiario? (S/N)
S
Ingresa el monto:
-111
El monto debe de ser mayor a cero y menor a diez millones de pesos mexicanos
    
```

² Código disponible en: https://github.com/RubioHaro/POO/tree/master/escom/De%20la%20O/Departamental1/Practicas/Rubio_Haro_Rodolfo_Rodolfo_Practica03_2CV1_Departamentall/CodigoFuente/3.Banco (12/04/2020)

3. Conclusión

El error más común en los programas considero que fueron de sintaxis. La mayor parte del tiempo más que corrigiendo errores, se llevó en aplicar los conceptos de la programación orientada a objetos. El programa primero: Desbordamiento imprime la cantidad de ciclos que se realizan al desbordar enteros por el lado positivo y el negativo. El segundo programa hace sonar un pitido al imprimir un mensaje mediante secuencias de escape. El tercero, más complejo, genera un cliente con una cuenta bancaria de tipo cheques, con su respectiva chequera, y, mediante una interfaz, va generando e imprimiendo los cheques.