**2.1** **Algoritmo: “Clavando tablones”**

Desarrollado en JavaScript en Visual Studio Code:

var A = [];

var B = [];

var C = [];

var J = 0;

var N = 0;

var M = 0;

//Asignacion de ejemplos

A[0] = 1;

B[0] = 4;

A[1] = 4;

B[1] = 5;

A[2] = 5;

B[2] = 9;

A[3] = 8;

B[3] = 10;

C[0] = 4;

C[1] = 6;

C[2] = 0;

C[3] = 10;

C[4] = 2;

N = A.length;

M = C.length;

//ejecucion de ejemplo

J = solucion(A, B, N, C, M);

//resultado

console.log("numero de clavos minimo: "+ J);

function solucion(a, b, n, c, m) {

    let j = 0;

    let clavados = [];

    let hechos = [];

    // itaracion de los clavos

    for (let i = 0; i < m; i++) {

        let verificador = 0;

        clavados[i] = [];

        // ve a que tablon puede clavar cada clavo

        for (let k = 0; k < n; k++) {

            if (a[k] <= c[i] & c[i] <= b[k])

                clavados[i].push(k);

        }

        console.log("clavo N°" + i + " clava al tablon: " + clavados[i]);

        // itero los tablones que clavo el clavo seleccionado para guardarlos

        for (let q = 0; q < clavados[i].length; q++) {

            hechos.push(clavados[i][q]);

            //elimino id repedidos del array de tabloques ya clavados

            hechos = unique(hechos);

            //ordeno el array de tabloques ya clavados

            hechos.sort(function (a, b) {

                return a - b;

            });

            console.log(hechos);

        }

        //se compara los id de tablones clavados con la lista de tablones

        for (let k = 0; k < n; k++) {

            if (hechos[k] == k) {

                verificador++;

            }

        }

        /\*si el numero de ids correctos es igual

         \*a el numero de ids total devulve la

         \*cantidad minima de clavos

        \*/

        if (verificador == n) {

            j = verificador;

        } else {

            j = -1

        }

    }

    return j;

}

//funcion para eliminar repetidos

function unique(arrayZ) {

    return arrayZ.filter((valor, indiceActual, arreglo) => arreglo.indexOf(valor) === indiceActual);

}

**Algoritmo: “Buscando Anagramas”:**

Desarrollado en lenguaje Java IDE Eclipse:

public class Anagrama {public static void main(String[] args) {

int a= *solucion*("hola, que buena ola Laomir", "ALO");

System.***out***.print(a);

}

public static int solucion(String A, String B){

A=A.toLowerCase();

B=B.toLowerCase();

int larB = B.length();

int cantidad=0;

int ubicacion=0;

String a[] = A.split(" ");

char aB[] = B.toCharArray();

ArrayList<String> aprobados= new ArrayList<String>();

*combinaciones* = "";

String co = *generarAnagramas*(aB, "", larB);

String comb[] = co.split(" ");

for (int i = 0; i < a.length; i++) {

if (a[i].length()>=larB) {

int posee =0;

for (int j = 0; j < aB.length; j++) {

if (a[i].indexOf(aB[j])> -1)

posee++;

}

if(posee == larB)

aprobados.add(a[i]);

}

}

for (int i = 0; i < aprobados.size(); i++) {

for (int j = 0; j < comb.length; j++) {

System.***out***.print(aprobados.get(i)+ " item "+comb[j]);

if(aprobados.get(i).indexOf(comb[j])>-1) {

System.***out***.println(" si ");

cantidad++;

}else {

System.***out***.println(" no ");

}

}

}

return cantidad;

}

static boolean[] *control* = new boolean[100];

static String *combinaciones* = "";

//Genero todos los anagramas posibles de la palabra dada y los guardo en un string separados por espacios

public static String generarAnagramas(char[] elementos, String actual, int cantidad) {

if(cantidad==0) {// Detecta secuencia generada

System.***out***.println(actual);

*combinaciones*+= actual+" ";

}

else {

for(int i=0; i<elementos.length; i++) {

if(*control*[i]==true) continue;

*control*[i]=true;

*generarAnagramas*(elementos, actual+elementos[i],cantidad-1);

*control*[i]=false;

}

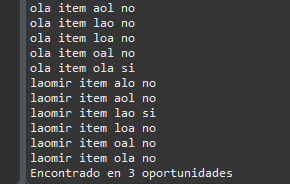
}

return *combinaciones*;

}

}

Resultado:



**2.2 Diseño:**

Resultado:  


Código HTML:

<!DOCTYPE html>

<html lang="es">

<head>

  <meta charset="utf-8">

  <meta name="author" content="Juan Rodriguez" />

  <title>Diseño </title>

  <link rel="stylesheet" href="styles/index.css">

</head>

<body>

  <!-- Titulo -->

  <header>

    <H1>Resumen del monto recibido</H1>

  </header>

  <section>

    <!-- valores -->

    <div>

      <table>

      <thead>

        <tr>

          <th>CANTIDAD</th>

          <th>VALOR</th>

          <th>PESOS</th>

          <th class="separador"></th>

          <th>CANTIDAD</th>

          <th>VALOR</th>

          <th>PESOS</th>

        </tr>

      </thead>

      <tbody>

        <tr>

          <td>0</td>

          <td><img src="img/billete-de-un-dolar.png" alt="icon" height="42" width="42"></td>

          <td>0</td>

          <td class="separador"></td>

          <td>100</td>

          <td><img src="img/billete-de-un-dolar.png" alt="icon" height="42" width="42"></td>

          <td>5.000</td>

        </tr>

        <tr>

          <td>5</td>

          <td><img src="img/billete-de-un-dolar.png" alt="icon" height="42" width="42"></td>

          <td>25</td>

          <td class="separador"></td>

          <td>200</td>

          <td><img src="img/billete-de-un-dolar.png" alt="icon" height="42" width="42"></td>

          <td>20.000</td>

        </tr>

        <tr>

          <td>10</td>

          <td><img src="img/billete-de-un-dolar.png" alt="icon" height="42" width="42"></td>

          <td>100</td>

          <td class="separador"></td>

          <td>50</td>

          <td><img src="img/billete-de-un-dolar.png" alt="icon" height="42" width="42"></td>

          <td>10.000</td>

        </tr>

        <tr>

          <td>20</td>

          <td><img src="img/billete-de-un-dolar.png" alt="icon" height="42" width="42"></td>

          <td>400</td>

          <td class="separador"></td>

          <td>100</td>

          <td><img src="img/billete-de-un-dolar.png" alt="icon" height="42" width="42"></td>

          <td>50.000</td>

        </tr>

      </tbody>

    </table>

    <!-- totales -->

    <article class="totales">

      <h3>CANTIDAD TOTAL</h3>

      <p class="total">328</p>

      <h3>MONTO TOTAL</h3>

      <p class="total">$53.564</p>

    </article>

    </div>

    <!-- botones -->

    <article class="botones">

      <button>Agregar mas Billetes</button>

      <button>Aceptar</button>

    </article>

  </section>

  <!-- Scripts -->

  <script src="scripts/main.js"></script>

</body>

</html>

CodigoCSS:

body{

  background-color: rgb(51, 51, 51);

}

h1{

  padding: 50px 5px 20px 5px;

  color:  rgb(39, 151, 72);

  text-align: center;

  margin: 0%;

}

section{

  background-image: url("../img/verde.jpg");

  background-size: cover;

  text-align: center;

}

div{

  display: flex;

  justify-content: center;

}

th{

  padding: 5px 10px;

  color: rgb(51, 51, 51);

}

td{

  color: white;

}

table{

  display: inline-block;

}

.totales{

  color: rgb(51, 51, 51);

  display: inline-block;

  margin: 0px 2%;

}

.separador{

  padding: 20px;

}

.total{

  background-color: rgb(255, 255, 255);

  padding: 10px;

  color:  rgb(39, 151, 72);

  border-radius: 15px;

  box-shadow: inset 4px 0px 4px rgba(0,0,0,.5),

              inset -4px 0px 4px rgba(0,0,0,.5);

}

.botones{

  display: block;

}

button{

  background-color: rgba(255, 255, 255, 0);

  padding: 10px 7%;

  margin: 30px;

  border: 0px;

  color:  white;

  border-radius: 15px;

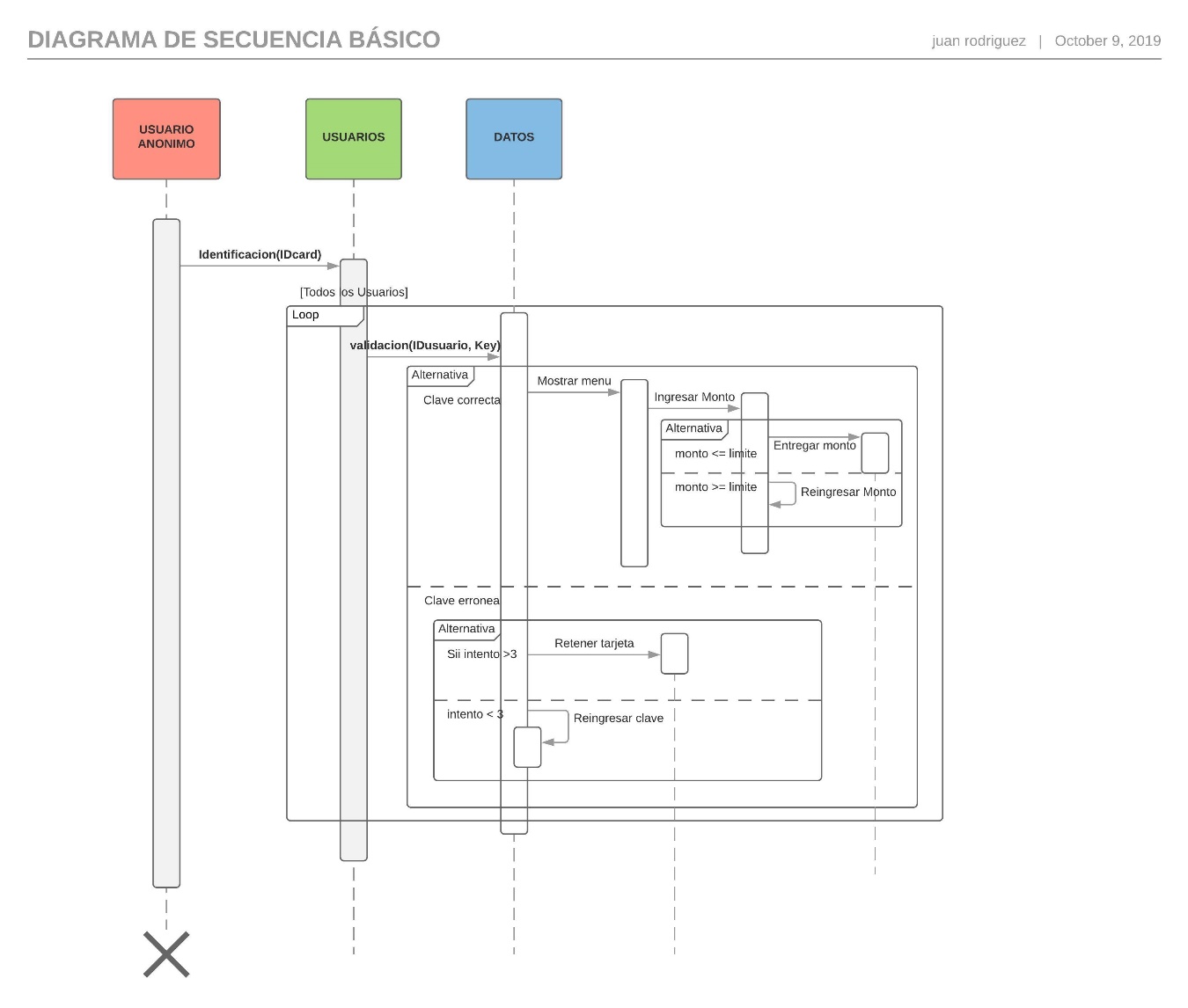
  box-shadow: inset 4px 0px 4px rgba(0,0,0,.5),

              inset -4px 0px 4px rgba(0,0,0,.5);

}

**2.3 Modelado:**

1. Diagrama de secuencias extracción cajero automático:



1. Diagrama de Clases Usuarios:

