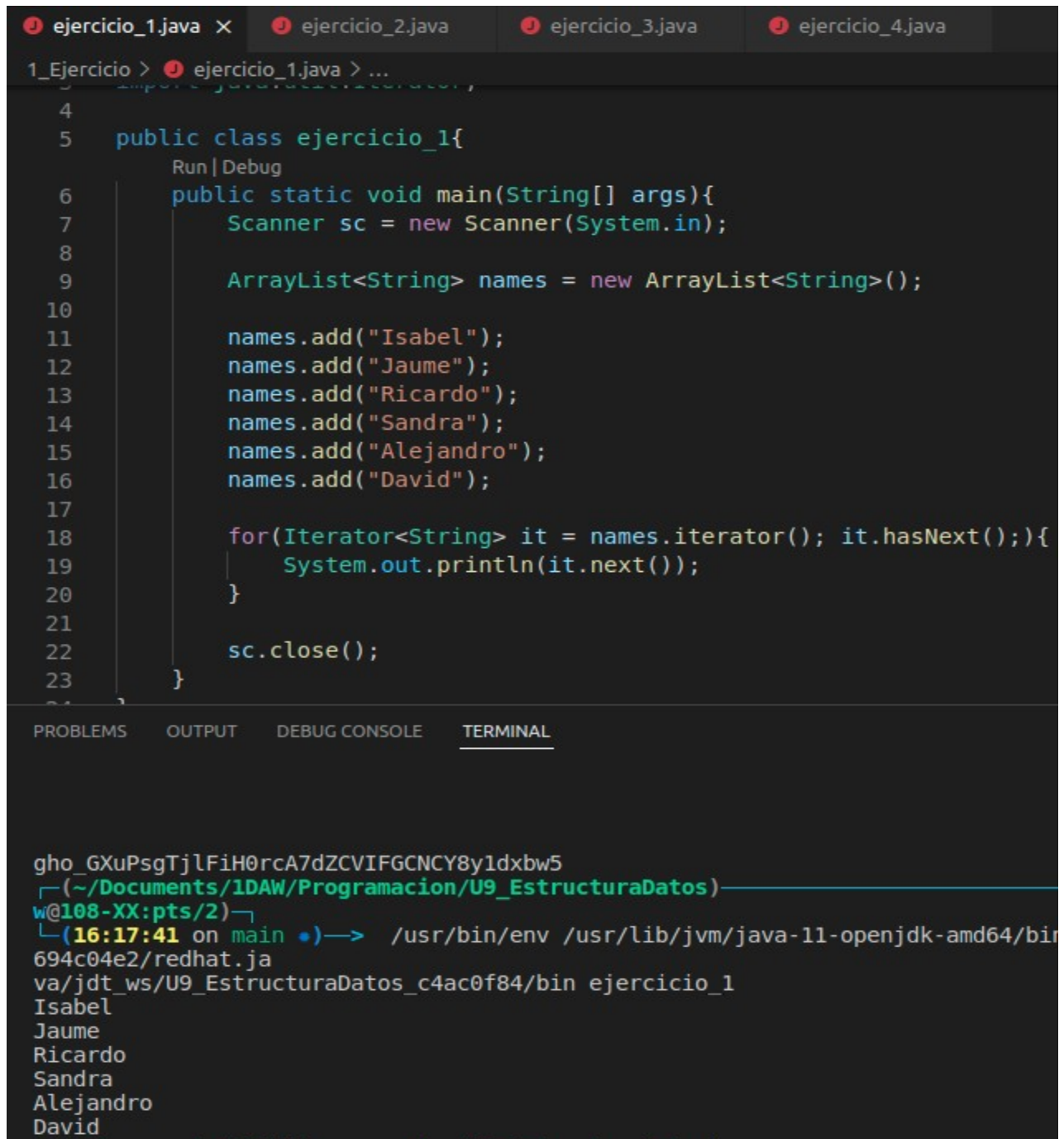


COLECCIONES Y DICCIONARIOS (Edison Alcocer)

Ejercicio 1

Crea un ArrayList con los nombres de 6 compañeros de clase. A continuación, muestra esos nombres por pantalla. Utiliza para ello un bucle for que recorra todo el ArrayList sin usar ningún índice.



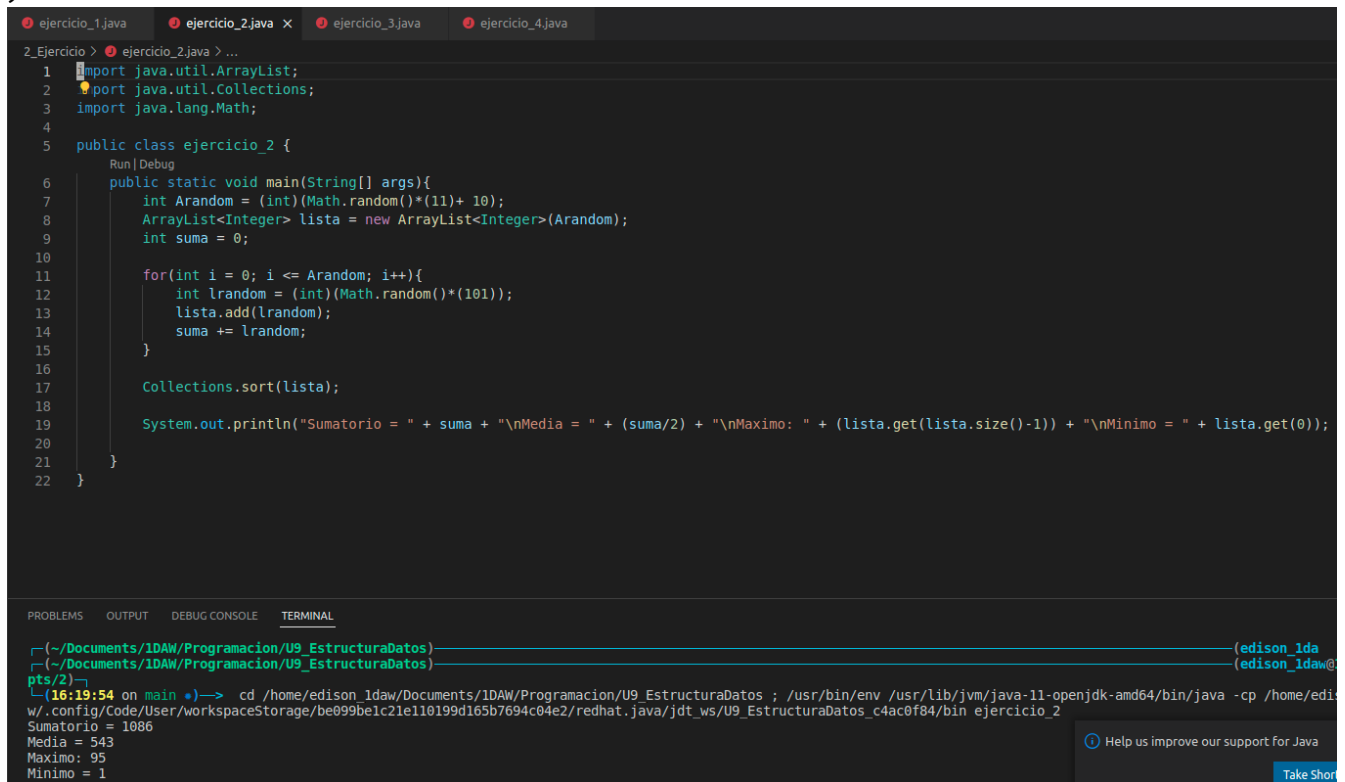
```
ejercicio_1.java x  ejercicio_2.java  ejercicio_3.java  ejercicio_4.java
1_Ejercicio > ejercicio_1.java > ...
4
5 public class ejercicio_1{
    Run | Debug
6     public static void main(String[] args){
7         Scanner sc = new Scanner(System.in);
8
9         ArrayList<String> names = new ArrayList<String>();
10
11         names.add("Isabel");
12         names.add("Jaume");
13         names.add("Ricardo");
14         names.add("Sandra");
15         names.add("Alejandro");
16         names.add("David");
17
18         for(Iterator<String> it = names.iterator(); it.hasNext();){
19             System.out.println(it.next());
20         }
21
22         sc.close();
23     }
24 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
gho_GXuPsgTj1FiH0rcA7dZCVIFGCNCY8y1dxbw5
└─(~/Documents/1DAW/Programacion/U9_EstructuraDatos)─┐
w@108-XX:pts/2)└─┐
└─(16:17:41 on main •)─> /usr/bin/env /usr/lib/jvm/java-11-openjdk-amd64/bin
694c04e2/redhat.ja
va/jdt_ws/U9_EstructuraDatos_c4ac0f84/bin ejercicio_1
Isabel
Jaume
Ricardo
Sandra
Alejandro
David
```

Ejercicio 2

Realiza un programa que introduzca valores aleatorios (entre 0 y 100) en un ArrayList y que luego calcule la suma, la media, el máximo y el mínimo de esos números. El tamaño de la lista también será aleatorio y podrá oscilar entre 10 y 20 elementos ambos inclusive.



```
2_Ejercicio > ejercicio_2.java > ...
1  import java.util.ArrayList;
2  import java.util.Collections;
3  import java.lang.Math;
4
5  public class ejercicio_2 {
6      public static void main(String[] args){
7          int Arandom = (int)(Math.random()*(11)+ 10);
8          ArrayList<Integer> lista = new ArrayList<Integer>(Arandom);
9          int suma = 0;
10
11          for(int i = 0; i <= Arandom; i++){
12              int lrandom = (int)(Math.random()*(101));
13              lista.add(lrandom);
14              suma += lrandom;
15          }
16
17          Collections.sort(lista);
18
19          System.out.println("Sumatorio = " + suma + "\nMedia = " + (suma/2) + "\nMaximo: " + (lista.get(lista.size()-1)) + "\nMinimo = " + lista.get(0));
20
21      }
22  }
```

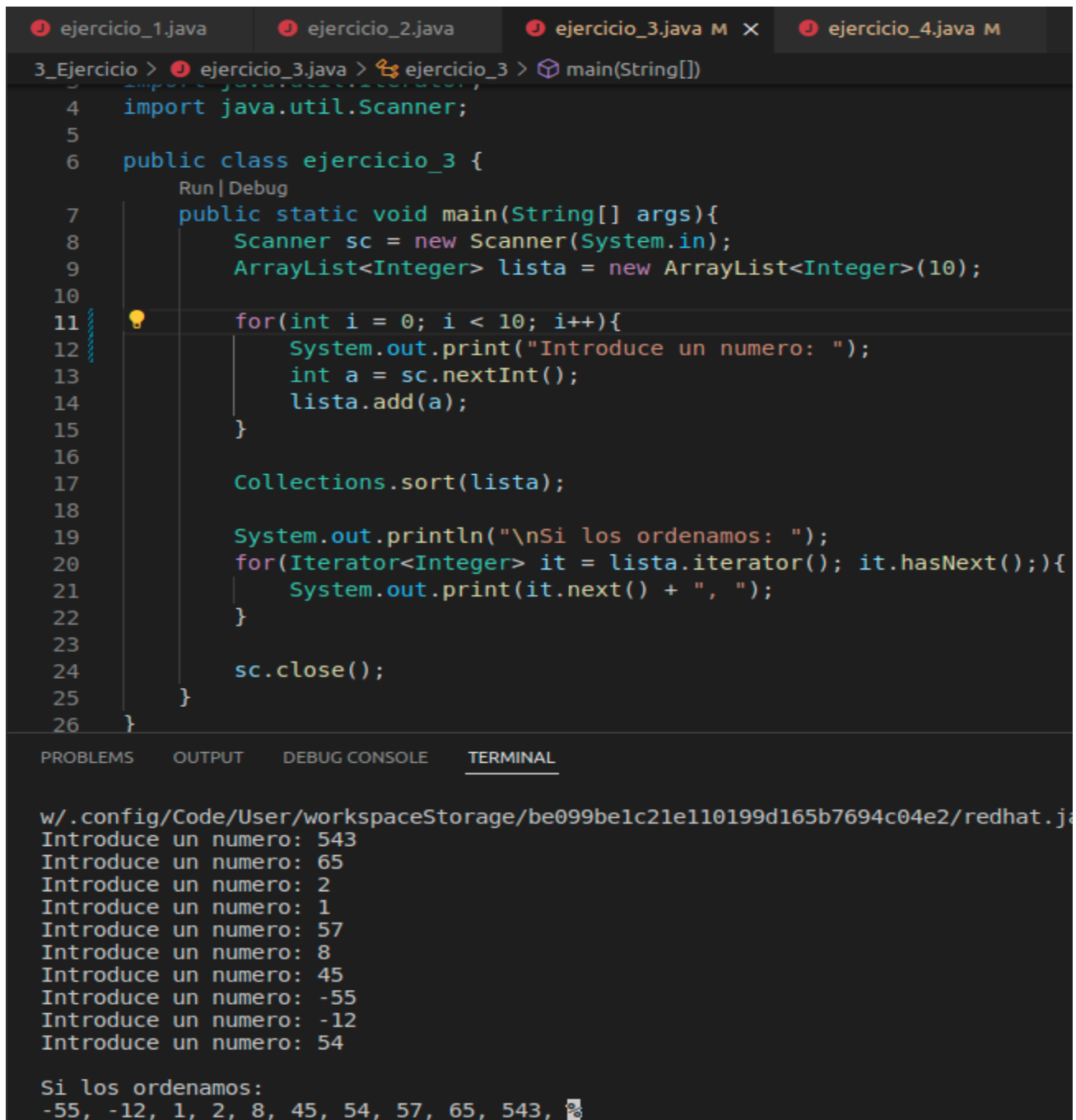
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
(~/Documents/1DAW/Programacion/U9_EstructuraDatos) (edison_1da
~/Documents/1DAW/Programacion/U9_EstructuraDatos) (edison_1daw@
(16:19:54 on main *)-> cd /home/edison_1daw/Documents/1DAW/Programacion/U9_EstructuraDatos ; /usr/bin/env /usr/lib/jvm/java-11-openjdk-amd64/bin/java -cp /home/edison_1daw/Documents/1DAW/Programacion/U9_EstructuraDatos/w/.config/Code/User/workspaceStorage/be099be1c21e110199d165b7694c04e2/redhat.java/jdt_ws/U9_EstructuraDatos_c4ac0f84/bin ejercicio_2
Sumatorio = 1086
Media = 543
Maximo: 95
Minimo = 1
```

Help us improve our support for Java
Take Short

Ejercicio 3

Escribe un programa que ordene 10 números enteros introducidos por teclado y almacenados en un objeto de la clase ArrayList .



```
ejercicio_1.java  ejercicio_2.java  ejercicio_3.java M X  ejercicio_4.java M
3_Ejercicio >  ejercicio_3.java >  ejercicio_3 >  main(String[])
 1  import java.util.ArrayList;
 2
 3  import java.util.Scanner;
 4
 5
 6  public class ejercicio_3 {
 7      Run | Debug
 8      public static void main(String[] args){
 9          Scanner sc = new Scanner(System.in);
10          ArrayList<Integer> lista = new ArrayList<Integer>(10);
11
12          for(int i = 0; i < 10; i++){
13              System.out.print("Introduce un numero: ");
14              int a = sc.nextInt();
15              lista.add(a);
16          }
17
18          Collections.sort(lista);
19
20          System.out.println("\nSi los ordenamos: ");
21          for(Iterator<Integer> it = lista.iterator(); it.hasNext();){
22              System.out.print(it.next() + ", ");
23          }
24
25          sc.close();
26      }
27  }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
w/.config/Code/User/workspaceStorage/be099be1c21e110199d165b7694c04e2/redhat.ja
Introduce un numero: 543
Introduce un numero: 65
Introduce un numero: 2
Introduce un numero: 1
Introduce un numero: 57
Introduce un numero: 8
Introduce un numero: 45
Introduce un numero: -55
Introduce un numero: -12
Introduce un numero: 54

Si los ordenamos:
-55, -12, 1, 2, 8, 45, 54, 57, 65, 543, 
```

Ejercicio 4

Realiza un programa equivalente al anterior pero en esta ocasión, el programa debe ordenar palabras en lugar de números

```
ejercicio_1.java  ejercicio_2.java  ejercicio_3.java M  ejercicio_4.java M X
4_Ejercicio >  ejercicio_4.java >  ejercicio_4 >  main(String[])
import java.util.Scanner;
4  import java.util.Scanner;
5
6  public class ejercicio_4 {
    Run | Debug
7      public static void main(String[] args){
8          Scanner sc = new Scanner(System.in);
9          ArrayList<String> lista = new ArrayList<String>(10);
10
11          System.out.println("Introduce 10 palabras : ");
12          for(int i = 0; i < 10; i++){
13              System.out.print((i + 1) + ": ");
14              String a = sc.nextLine();
15              lista.add(a);
16          }
17
18          Collections.sort(lista);
19
20          System.out.println("\nOrdenados: ");
21          for(Iterator<String> it = lista.iterator(); it.hasNext();){
22              System.out.print(it.next() + ", ");
23          }
24
25          sc.close();
26      }
}

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

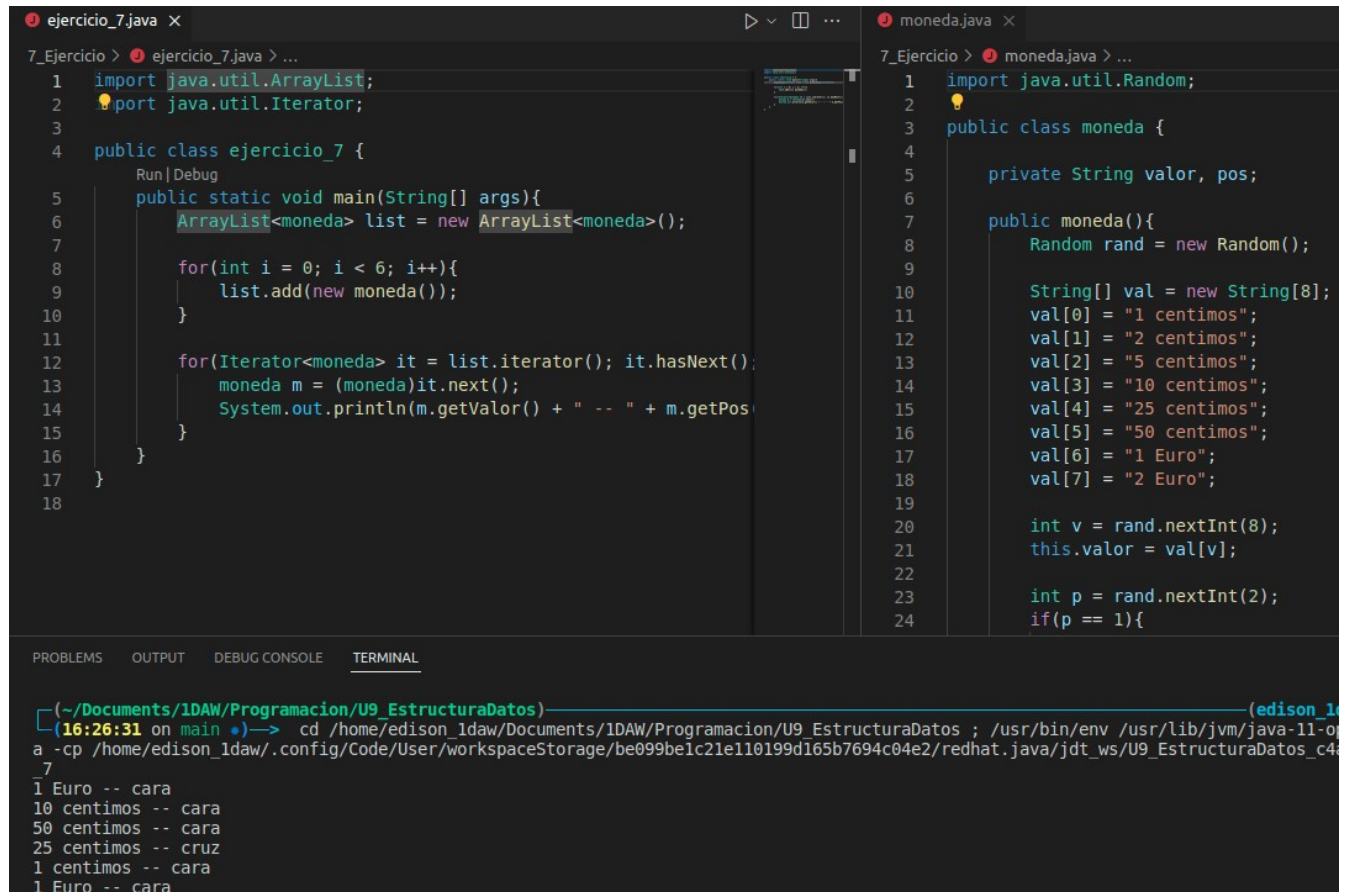
Introduce 10 palabras :
1: Montaña
2: Cotxe
3: Moto
4: Playa
5: Zebra
6: Cojin
7: Imposible
8: Ecologico
9: Haver
10: Ave

Ordenados:
Ave, Cojin, Cotxe, Ecologico, Haver, Imposible, Montaña, Moto, Playa, Zebra, 🗑
```

Ejercicio 7

La máquina Eurocoin genera una moneda de curso legal cada vez que se pulsa un botón siguiendo la siguiente pauta: o bien coincide el valor con la moneda

anteriormente generada – 1 céntimo, 2 céntimos, 5 céntimos, 10 céntimos, 25 céntimos, 50 céntimos, 1 euro o 2 euros – o bien coincide la posición – cara o cruz. Simula, mediante un programa, la generación de 6 monedas aleatorias siguiendo la pauta correcta. Cada moneda generada debe ser una instancia de la clase Moneda y la secuencia se debe ir almacenando en una lista.



```
ejercicio_7.java
1 import java.util.ArrayList;
2 import java.util.Iterator;
3
4 public class ejercicio_7 {
5     public static void main(String[] args){
6         ArrayList<moneda> list = new ArrayList<moneda>();
7
8         for(int i = 0; i < 6; i++){
9             list.add(new moneda());
10        }
11
12        for(Iterator<moneda> it = list.iterator(); it.hasNext();){
13            moneda m = (moneda)it.next();
14            System.out.println(m.getValor() + " -- " + m.getPos);
15        }
16    }
17 }
18

moneda.java
1 import java.util.Random;
2
3 public class moneda {
4     private String valor, pos;
5
6     public moneda(){
7         Random rand = new Random();
8
9         String[] val = new String[8];
10        val[0] = "1 centimos";
11        val[1] = "2 centimos";
12        val[2] = "5 centimos";
13        val[3] = "10 centimos";
14        val[4] = "25 centimos";
15        val[5] = "50 centimos";
16        val[6] = "1 Euro";
17        val[7] = "2 Euro";
18
19        int v = rand.nextInt(8);
20        this.valor = val[v];
21
22        int p = rand.nextInt(2);
23        if(p == 1){
24            pos = "cruz";
25        } else {
26            pos = "cara";
27        }
28    }
29
30    String getValor(){
31        return valor;
32    }
33
34    String getPos(){
35        return pos;
36    }
37 }
```

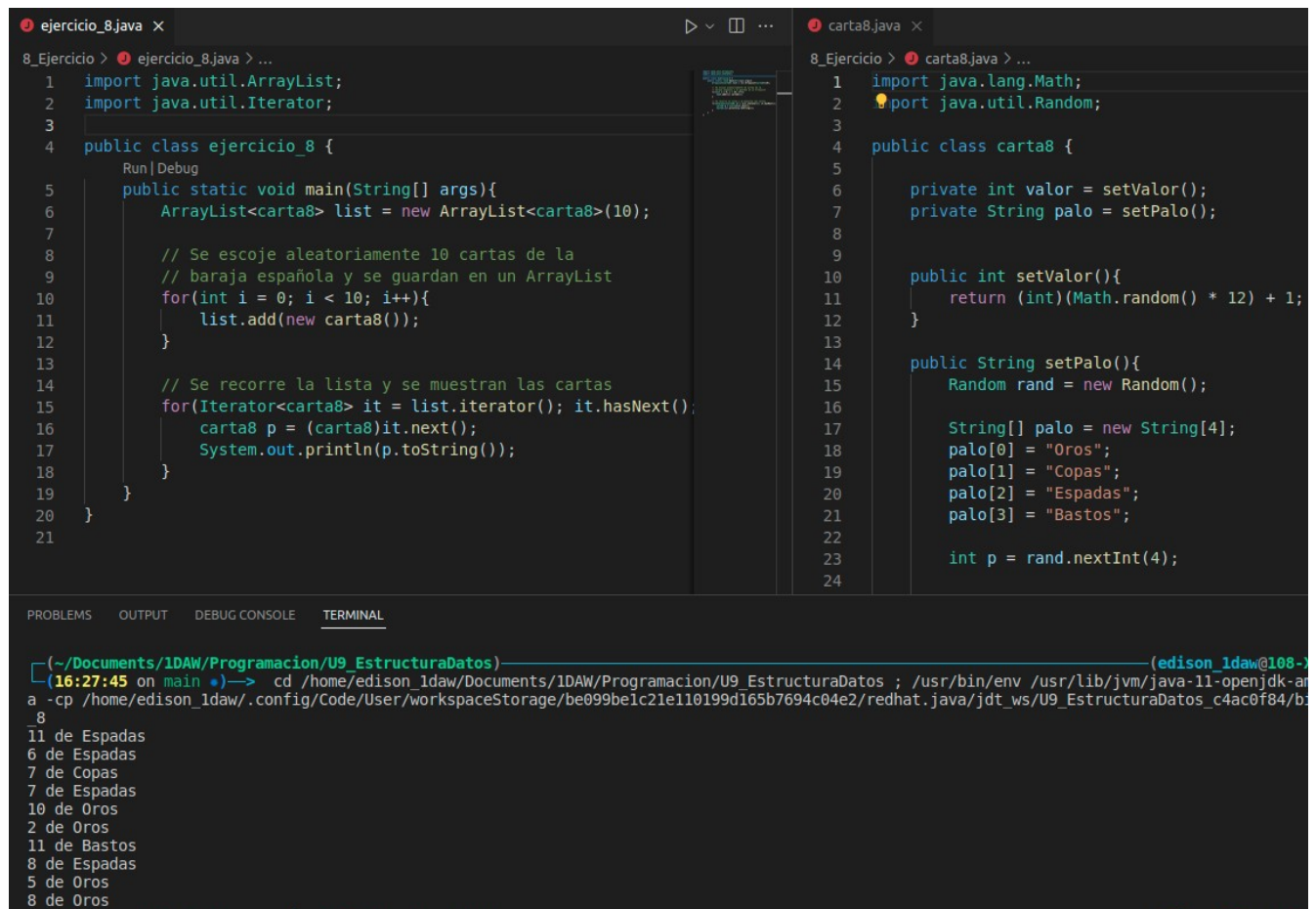
```

~/Documents/1DAW/Programacion/U9_EstructuraDatos (edison_1
(16:26:31 on main) → cd /home/edison_ldaw/Documents/1DAW/Programacion/U9_EstructuraDatos ; /usr/bin/env /usr/lib/jvm/java-11-o
a -cp /home/edison_ldaw/.config/Code/User/workspaceStorage/be099be1c21e110199d165b7694c04e2/redhat.java/jdt_ws/U9_EstructuraDatos_c4
7
1 Euro -- cara
10 centimos -- cara
50 centimos -- cara
25 centimos -- cruz
1 centimos -- cara
1 Euro -- cara
```

Ejercicio 8

Realiza un programa que escoja al azar 10 cartas de la baraja española (10 objetos de la clase Carta). Emplea un objeto de la clase ArrayList para

almacenarlas y asegúrate de que no se repite ninguna.



```
ejercicio_8.java x
8_Ejercicio > ejercicio_8.java > ...
1 import java.util.ArrayList;
2 import java.util.Iterator;
3
4 public class ejercicio_8 {
5     public static void main(String[] args){
6         ArrayList<carta8> list = new ArrayList<carta8>(10);
7
8         // Se elige aleatoriamente 10 cartas de la
9         // baraja española y se guardan en un ArrayList
10        for(int i = 0; i < 10; i++){
11            list.add(new carta8());
12        }
13
14        // Se recorre la lista y se muestran las cartas
15        for(Iterator<carta8> it = list.iterator(); it.hasNext();
16            carta8 p = (carta8)it.next();
17            System.out.println(p.toString());
18        }
19    }
20 }
21

carta8.java x
8_Ejercicio > carta8.java > ...
1 import java.lang.Math;
2 import java.util.Random;
3
4 public class carta8 {
5
6     private int valor = setValor();
7     private String palo = setPalo();
8
9
10    public int setValor(){
11        return (int)(Math.random() * 12) + 1;
12    }
13
14    public String setPalo(){
15        Random rand = new Random();
16
17        String[] palo = new String[4];
18        palo[0] = "Oros";
19        palo[1] = "Copas";
20        palo[2] = "Espadas";
21        palo[3] = "Bastos";
22
23        int p = rand.nextInt(4);
24    }
25 }
```

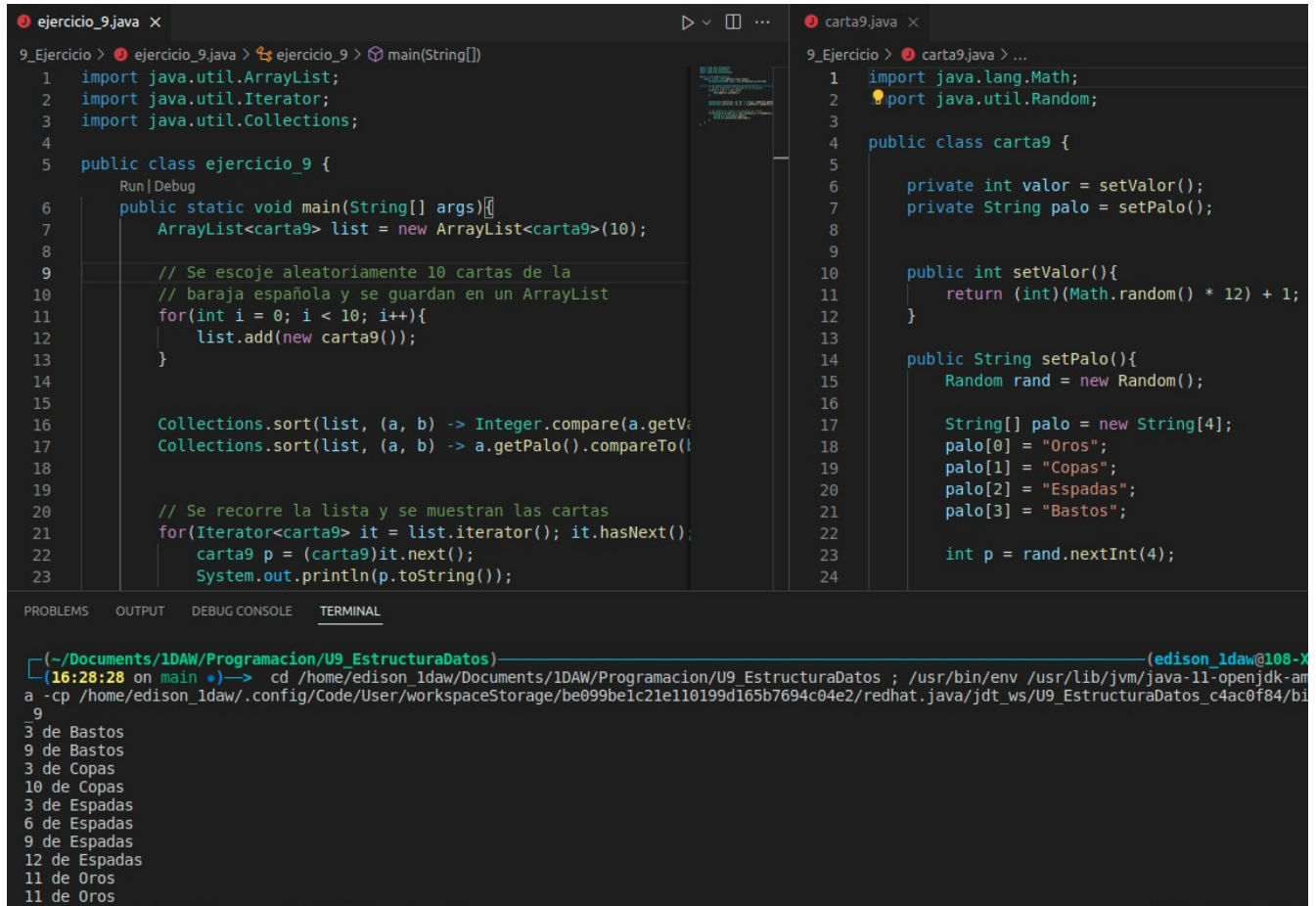
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
(~/Documents/1DAW/Programacion/U9_EstructuraDatos) (edison_1daw@108-...)
(16:27:45 on main *) -> cd /home/edison_1daw/Documents/1DAW/Programacion/U9_EstructuraDatos ; /usr/bin/env /usr/lib/jvm/java-11-openjdk-ar
a -cp /home/edison_1daw/.config/Code/User/workspaceStorage/be099be1c21e110199d165b7694c04e2/redhat.java/jdt_ws/U9_EstructuraDatos_c4ac0f84/b
8
11 de Espadas
6 de Espadas
7 de Copas
7 de Espadas
10 de Oros
2 de Oros
11 de Bastos
8 de Espadas
5 de Oros
8 de Oros
```

Ejercicio 9

Modifica el programa anterior de tal forma que las cartas se muestren orde-
nadas. Primero se ordenarán por palo: bastos, copas, espadas, oros. Cuando

coincida el palo, se ordenará por número: as, 2, 3, 4, 5, 6, 7, sota, caballo, rey.



```
ejercicio_9.java X
9_Ejercicio > ejercicio_9.java > ejercicio_9 > main(String[])
1 import java.util.ArrayList;
2 import java.util.Iterator;
3 import java.util.Collections;
4
5 public class ejercicio_9 {
6     public static void main(String[] args){
7         ArrayList<carta9> list = new ArrayList<carta9>(10);
8
9         // Se elige aleatoriamente 10 cartas de la
10        // baraja española y se guardan en un ArrayList
11        for(int i = 0; i < 10; i++){
12            list.add(new carta9());
13        }
14
15        Collections.sort(list, (a, b) -> Integer.compare(a.getValor(), b.getValor()));
16        Collections.sort(list, (a, b) -> a.getPalo().compareTo(b.getPalo()));
17
18        // Se recorre la lista y se muestran las cartas
19        for(Iterator<carta9> it = list.iterator(); it.hasNext();){
20            carta9 p = (carta9)it.next();
21            System.out.println(p.toString());
22        }
23    }
24}
```

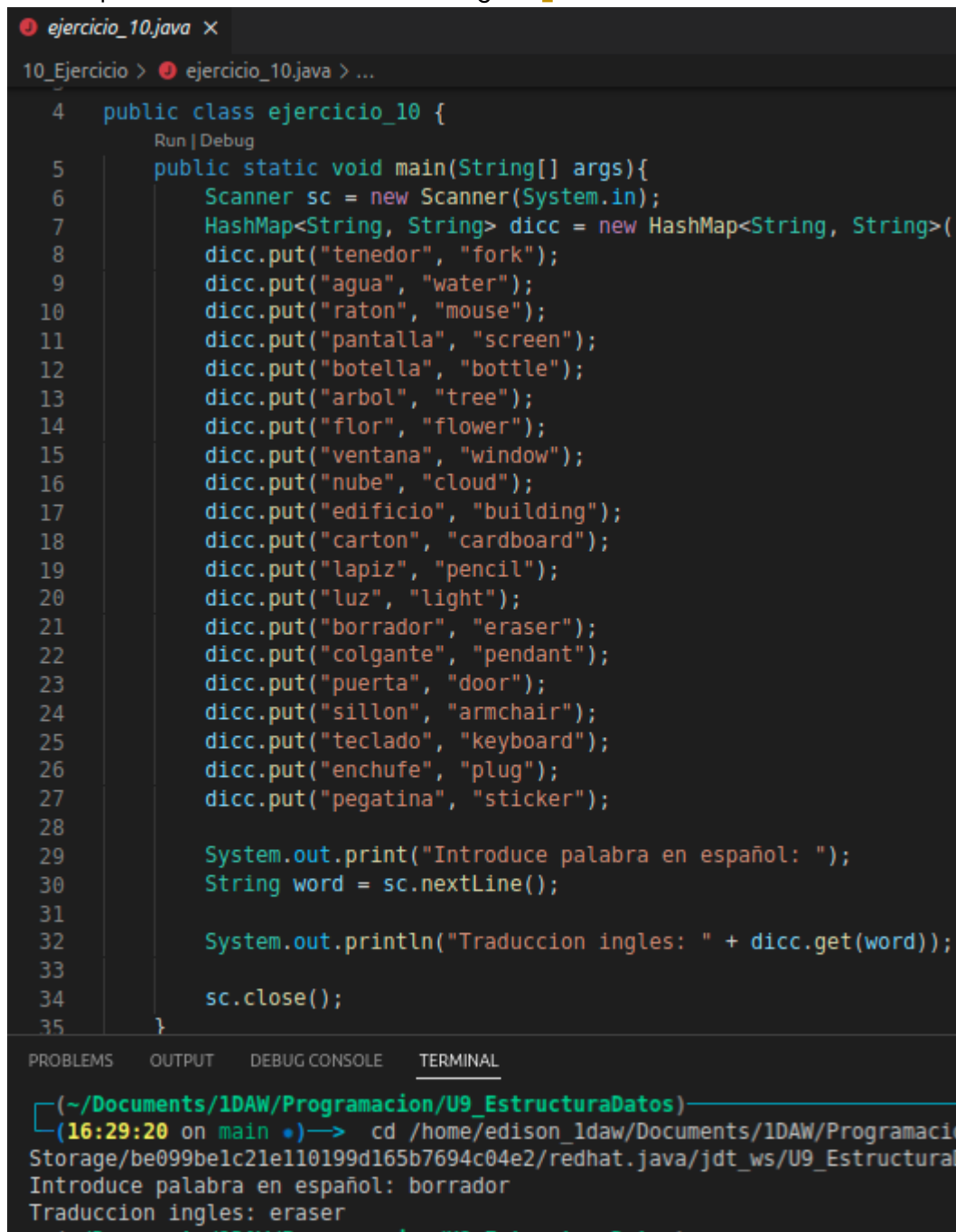
```
carta9.java X
9_Ejercicio > carta9.java > ...
1 import java.lang.Math;
2 import java.util.Random;
3
4 public class carta9 {
5
6     private int valor = setValor();
7     private String palo = setPalo();
8
9
10    public int setValor(){
11        return (int)(Math.random() * 12) + 1;
12    }
13
14    public String setPalo(){
15        Random rand = new Random();
16
17        String[] palo = new String[4];
18        palo[0] = "Oros";
19        palo[1] = "Copas";
20        palo[2] = "Espadas";
21        palo[3] = "Bastos";
22
23        int p = rand.nextInt(4);
24    }
25}
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
~/Documents/1DAW/Programacion/U9_EstructuraDatos (edison_1daw@108-X)
16:28:28 on main * -> cd /home/edison_1daw/Documents/1DAW/Programacion/U9_EstructuraDatos ; /usr/bin/env /usr/lib/jvm/java-11-openjdk-amd64/bin/java -cp /home/edison_1daw/.config/Code/User/workspaceStorage/be099belc21e110199d165b7694c04e2/redhat.java/jdt_ws/U9_EstructuraDatos_c4ac0f84/bi
9
3 de Bastos
9 de Bastos
3 de Copas
10 de Copas
3 de Espadas
6 de Espadas
9 de Espadas
12 de Espadas
11 de Oros
11 de Oros
```

Ejercicio 10

Crea un mini-diccionario español-inglés que contenga, al menos, 20 palabras (con su correspondiente traducción). Utiliza un objeto de la clase HashMap para

almacenar las parejas de palabras. El programa pedirá una palabra en español y dará la correspondiente traducción en inglés.



```
ejercicio_10.java X
10_Ejercicio > ejercicio_10.java > ...

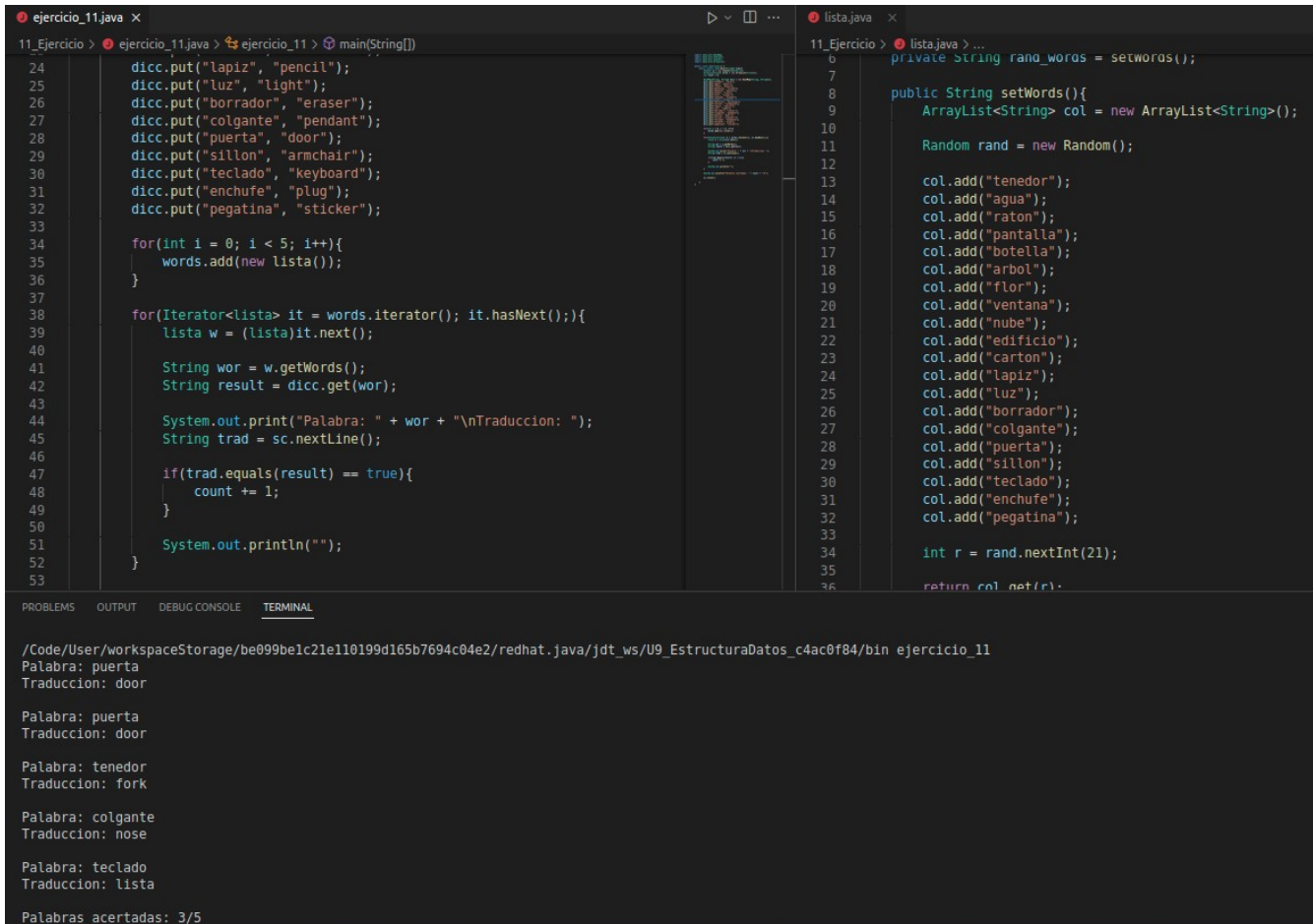
4 public class ejercicio_10 {
5     public static void main(String[] args){
6         Scanner sc = new Scanner(System.in);
7         HashMap<String, String> dicc = new HashMap<String, String>();
8         dicc.put("tenedor", "fork");
9         dicc.put("agua", "water");
10        dicc.put("raton", "mouse");
11        dicc.put("pantalla", "screen");
12        dicc.put("botella", "bottle");
13        dicc.put("arbol", "tree");
14        dicc.put("flor", "flower");
15        dicc.put("ventana", "window");
16        dicc.put("nube", "cloud");
17        dicc.put("edificio", "building");
18        dicc.put("carton", "cardboard");
19        dicc.put("lapiz", "pencil");
20        dicc.put("luz", "light");
21        dicc.put("borrador", "eraser");
22        dicc.put("colgante", "pendant");
23        dicc.put("puerta", "door");
24        dicc.put("sillon", "armchair");
25        dicc.put("teclado", "keyboard");
26        dicc.put("enchufe", "plug");
27        dicc.put("pegatina", "sticker");
28
29        System.out.print("Introduce palabra en español: ");
30        String word = sc.nextLine();
31
32        System.out.println("Traduccion ingles: " + dicc.get(word));
33
34        sc.close();
35    }
}

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
[~/Documents/1DAW/Programacion/U9_EstructuraDatos]
(16:29:20 on main •)→ cd /home/edison_ldaw/Documents/1DAW/Programacion/U9_EstructuraDatos
Introduce palabra en español: borrador
Traduccion ingles: eraser
```

Ejercicio 11

Realiza un programa que escoja al azar 5 palabras en español del mini-diccionario del ejercicio anterior. El programa irá pidiendo que el usuario teclee

la traducción al inglés de cada una de las palabras y comprobará si son correctas. Al final, el programa deberá mostrar cuántas respuestas son válidas y cuántas erróneas.



```
ejercicio_11.java x
11_Ejercicio > ejercicio_11.java > main(String[])
24     dicc.put("lapiz", "pencil");
25     dicc.put("luz", "light");
26     dicc.put("borrador", "eraser");
27     dicc.put("colgante", "pendant");
28     dicc.put("puerta", "door");
29     dicc.put("sillon", "armchair");
30     dicc.put("teclado", "keyboard");
31     dicc.put("enchufe", "plug");
32     dicc.put("pegatina", "sticker");
33
34     for(int i = 0; i < 5; i++){
35         words.add(new lista());
36     }
37
38     for(Iterator<lista> it = words.iterator(); it.hasNext();){
39         lista w = (lista)it.next();
40
41         String wor = w.getWords();
42         String result = dicc.get(wor);
43
44         System.out.print("Palabra: " + wor + "\nTraduccion: ");
45         String trad = sc.nextLine();
46
47         if(trad.equals(result) == true){
48             count += 1;
49         }
50
51         System.out.println("");
52     }
53
lista.java x
11_Ejercicio > lista.java > ...
6     private String rand_wordos = setwordos();
7
8     public String setWords(){
9         ArrayList<String> col = new ArrayList<String>();
10
11         Random rand = new Random();
12
13         col.add("tenedor");
14         col.add("agua");
15         col.add("raton");
16         col.add("pantalla");
17         col.add("botella");
18         col.add("arbol");
19         col.add("flor");
20         col.add("ventana");
21         col.add("nube");
22         col.add("edificio");
23         col.add("carton");
24         col.add("lapiz");
25         col.add("luz");
26         col.add("borrador");
27         col.add("colgante");
28         col.add("puerta");
29         col.add("sillon");
30         col.add("teclado");
31         col.add("enchufe");
32         col.add("pegatina");
33
34         int r = rand.nextInt(21);
35
36         return col.get(r);

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

/Code/User/workspaceStorage/be099belc21e110199d165b7694c04e2/redhat.java/jdt_ws/U9_EstructuraDatos_c4ac0f84/bin ejercicio_11

Palabra: puerta
Traduccion: door

Palabra: puerta
Traduccion: door

Palabra: tenedor
Traduccion: fork

Palabra: colgante
Traduccion: nose

Palabra: teclado
Traduccion: lista

Palabras acertadas: 3/5

Ejercicio 12

Escribe un programa que genere una secuencia de 5 cartas de la baraja española y que sume los puntos según el juego de la brisca. El valor de las

cartas se debe guardar en una estructura HashMap que debe contener parejas (figura, valor), por ejemplo ("caballo", 3). La secuencia de cartas debe ser una estructura de la clase ArrayList que contiene objetos de la clase Carta. El valor de las cartas es el siguiente: as → 11, tres → 10, sota → 2, caballo → 3, rey → 4; el resto de cartas no vale nada.

```
ejercicio_12.java x
12_Ejercicio > ejercicio_12.java > ejercicio_12 > main(String[])
4
5 public class ejercicio_12 {
6     public static void main(String[] args){
7         ArrayList<carta12> showCards = new ArrayList<carta12>();
8         HashMap<String, Integer> mapCards = new HashMap<String, Integer>();
9         int cont = 0;
10
11         mapCards.put("as", 11);
12         mapCards.put("tres", 10);
13         mapCards.put("sota", 2);
14         mapCards.put("caballo", 3);
15         mapCards.put("rey", 4);
16
17         for(int i = 0; i < 5; i++){
18             showCards.add(new carta12());
19         }
20
21         for(Iterator<carta12> it = showCards.iterator(); it.hasNext();){
22             carta12 c = (carta12)it.next();
23             System.out.println(c.getCarta());
24
25             if(mapCards.containsKey(c.getValor())){
26                 cont += mapCards.get(c.getValor());
27             }
28         }
29
30         System.out.println("Tienes " + cont + " puntos");
31
32     }
33 }

carta12.java x
12_Ejercicio > carta12.java > ...
4 public class carta12 {
5
6     private String valor = setValor();
7     private String palo = setPalo();
8
9     public String setPalo(){
10         Random rand = new Random();
11         String[] palo = new String[4];
12         palo[0] = "Oros";
13         palo[1] = "Copas";
14         palo[2] = "Espadas";
15         palo[3] = "Bastos";
16
17         int p = rand.nextInt(4);
18
19         return palo[p];
20     }
21
22     public String setValor(){
23         ArrayList<String> cartas = new ArrayList<String>();
24         Random rand = new Random();
25         cartas.add("as");
26         cartas.add("2");
27         cartas.add("3");
28         cartas.add("4");
29         cartas.add("5");
30         cartas.add("6");
31         cartas.add("7");
32         cartas.add("8");
33         cartas.add("9");
34     }
35 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
(~/Documents/1DAW/Programacion/U9_EstructuraDatos) (edison_1daw@)
(16:31:28 on main *) -> cd /home/edison_1daw/Documents/1DAW/Programacion/U9_EstructuraDatos ; /usr/bin/env /usr/lib/jvm/java-11-openjdk-amd64/bin/java -cp /home/edi
/Code/User/workspaceStorage/be099belc21e110199d165b7694c04e2/redhat.java/jdt_ws/U9_EstructuraDatos_c4ac0f84/bin ejercicio_12
6 de Oros
5 de Copas
as de Oros
rey de Copas
caballo de Bastos
Tienes 18 puntos
```