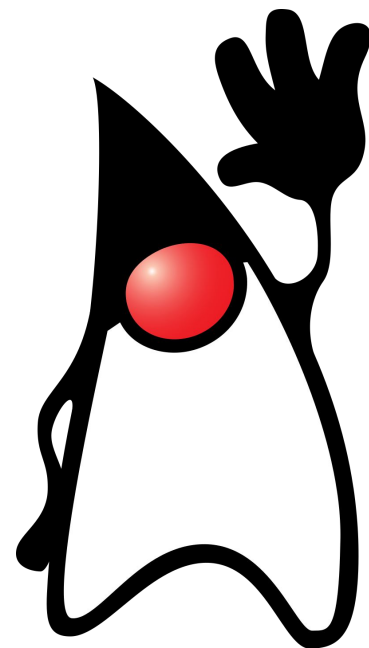




Boris Molina Igor
Taller de programación II
bmolintai@ing.ucsc.cl



Imports

```
import java.util.Scanner;  
import java.util.List;  
import java.util.ArrayList;  
import java.util.Collections;  
import java.util.Arrays;  
import java.util.Stack;
```

Stack

- Push
- Peek
- Pop
- Search
- Empty

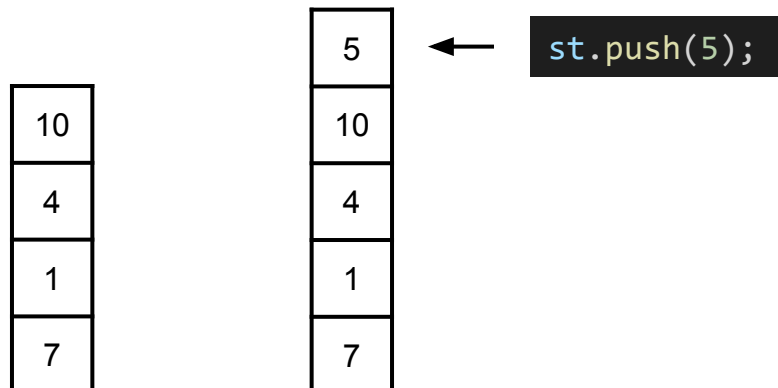
API Java Stack: <https://docs.oracle.com/javase/8/docs/api/java/util/Stack.html>

Stack métodos

```
Stack<Integer> st = new Stack<>();
st.push(7);
st.push(1);
st.push(4);
st.push(10);
```

Tipo	Método	Descripción
Object	<code>st.push(Object o);</code>	Coloca un objeto en la parte superior de esta pila.

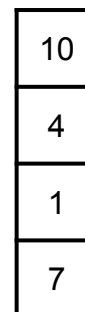
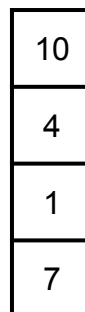
`st.push(5); //Inserta un objeto a la pila, en este caso un 5 [7, 1, 4, 10, 5]`



Stack métodos

```
Stack<Integer> st = new Stack<>();
st.push(7);
st.push(1);
st.push(4);
st.push(10);
```

Tipo	Método	Descripción
Object	<code>st.peek();</code>	Mira el objeto superior de esta pila pero sin quitarlo.
<code>st.peek(); //Retorna el objeto superior de esta pila, en este caso el 10</code>		

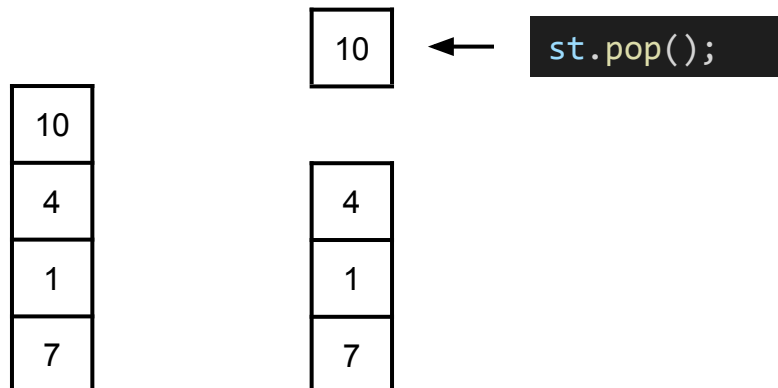


`st.peek();`

Stack métodos

```
Stack<Integer> st = new Stack<>();
st.push(7);
st.push(1);
st.push(4);
st.push(10);
```

Tipo	Método	Descripción
Object	<code>st.pop();</code>	Elimina el objeto superior de esta pila y devuelve ese objeto.
<code>st.pop();</code> //Retorna el objeto superior de esta pila y lo elimina, en este caso el 10		



Stack métodos

```
Stack<Integer> st = new Stack<>();
st.push(7);
st.push(1);
st.push(4);
st.push(10);
```

Tipo	Método	Descripción
int	<code>st.search(Object o);</code>	Devuelve la posición Base 1 donde un objeto está en esta pila.
<code>st.search(1); //Retorna el posición del objeto de esta pila, en este caso retorna 3</code>		

1	10
2	4
3	1
4	7

Stack métodos

```
Stack<Integer> st = new Stack<>();  
st.push(7);  
st.push(1);  
st.push(4);  
st.push(10);
```

Tipo	Método	Descripción
boolean	<code>st.empty();</code>	Comprueba si esta pila está vacía.
<code>st.empty(); //Retorna true si está vacía, falso caso contrario, en este caso retorna falso</code>		

Competencia 3

Indicaciones Competencia 3 (Evaluada)

1. Entrar a vjudge.net.
2. Registrarse en la página con un “***Username***” y “***Nickname***” descriptivos.
 - a. En el campo ***School*** rellenen con “**UCSC**”
3. Luego entren a la siguiente competencia [Competencia 3](#).
4. **Password:** ucsc

Reglas Competencia 3 (Evaluada)

1. Con 0 problema resuelto = 10.
2. Con 1 problema resueltos = 40.
3. Con 2 problemas resueltos = 70.

Cómo subir una solución

1. Hacer clic en “**Submit**”.
2. En **Problem**, seleccionar el problema del cual tiene la solución programada.
3. En **Language**, seleccionar **Java 1.8.0_241** ó **Java 1.8.0**.
4. En **Share**, seleccionar **NO**.
5. En **Solution**,
 - a. Copiar su código SIN el package.
 - b. Incluir las bibliotecas (import).
 - c. El nombre de la clase principal debe de ser **Main**.

Ejemplo

Submit

Problem:

D - Problema D

Language:

Java 1.8.0_241

Share:

Yes

No

Solution:

```
import java.util.ArrayList;
import java.util.List;
import java.util.Collections;
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        List<Integer> arr = new ArrayList<>();

        int n = sc.nextInt();

        for (int i = 0; i < n; i++) {
            arr.add(sc.nextInt());
        }

        Collections.sort(arr);

        for (int i = 0; i < arr.size(); i++) {
            System.out.print(arr.get(i) + " ");
        }
    }
}
```

Cancel

Submit