

EVALUACION PROCESUAL HITO 3

Estudiante: Saúl Escobar Serrano

CI: 9913271

Docente: ING. Patricia Fernández Quisbert

Fecha: 17 / 10 / 2023

Resolver los siguientes ejercicios utilizando PILAS y COLAS:
Presentar código fuente y captura de pantalla de la ejecución.

PILAS EJERCICIOS.

1. Obtener el tercer elemento.

P (6, 4, 2, 12, 7, 3) => 2

Código:

```
import java.util.Scanner;
import java.util.Stack;

public class EJERCICIO_1 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        Stack<Integer> pila = new Stack<>();
        int contador = 1;

        System.out.println("Ingrese elementos a la
pila:");
        while (true) {
            System.out.print("Elemento " + contador +
": ");

            String entrada = scanner.nextLine().trim();
            if (entrada.equalsIgnoreCase("")) {
                break;
            }
            try {
                int elemento =
Integer.parseInt(entrada);
                pila.push(elemento);
                contador++;
            } catch (NumberFormatException e) {
                System.out.println("Entrada no
válida.Introduzca un número entero.");
            }
        }
        if (pila.size() >= 3) {
            pila.pop();
            pila.pop();

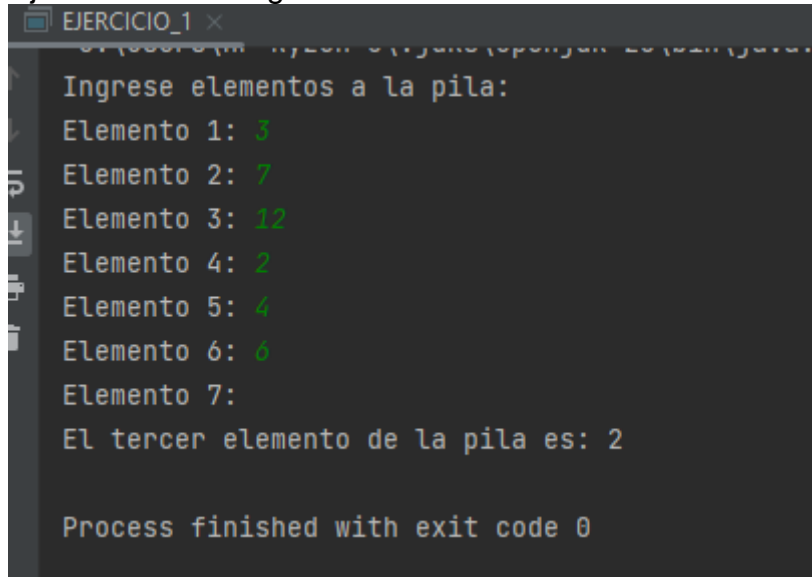
            int tercerElemento = pila.peek();
            System.out.println("El tercer elemento de
la pila es: " + tercerElemento);
        } else {
```

```

        System.out.println("La pila no tiene
suficientes elementos para obtener el tercero.");
    }
}
}

```

Ejecución del código:



```

EJERCICIO_1 x
Ingrese elementos a la pila:
Elemento 1: 3
Elemento 2: 7
Elemento 3: 12
Elemento 4: 2
Elemento 5: 4
Elemento 6: 6
Elemento 7:
El tercer elemento de la pila es: 2

Process finished with exit code 0

```

2. Inserta el elemento K antes del penúltimo elemento

P (8, 3, 7, 6, 5, 9)

K= 100 => P(8, 3, 7, 6, 100, 5, 9)

Código:

```

import java.util.Scanner;
import java.util.Stack;

public class EJERCICIO_2 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        Stack<Integer> pila = new Stack<Integer>();
        int k;

        System.out.println("Ingrese elementos:");
        while (true) {
            System.out.print("Elemento: ");
            String entrada = scanner.nextLine().trim();

            if (entrada.equalsIgnoreCase("")) {
                break;
            }

            try {
                int elemento = Integer.parseInt(entrada);
                pila.push(elemento);
            } catch (NumberFormatException e) {
                System.out.println("Entrada no válida.

```

```

    Introduzca un número entero.");
    }
}

if (pila.size() >= 2) {
    System.out.print("Ingrese el elemento K a
insertar: ");
    k = scanner.nextInt();

    Stack<Integer> pilaAux = new Stack<Integer>();

    int penultimoElemento = 0;
    boolean insertado = false;

    while (!pila.isEmpty()) {
        int elemento = pila.pop();

        if (!insertado && pila.size() == 1) {
            pilaAux.push(k);
            pilaAux.push(elemento);
            insertado = true;
        } else {
            pilaAux.push(elemento);
        }

        if (pila.size() == 1) {
            penultimoElemento = elemento;
        }
    }

    if (!insertado) {
        pilaAux.push(k);
        pilaAux.push(penultimoElemento);
    }

    System.out.println("Pila modificada:");
    while (!pilaAux.isEmpty()) {
        System.out.println(pilaAux.pop());
    }
} else {
    System.out.println("La pila no tiene
suficientes elementos para K.");
}
}
}

```

Ejecucion del codigo:

```

"C:\Users\HP-Ryzen 5\.jdk\openjdk-20\bin\java.exe
Ingrese elementos uno por uno a la pila (ingrese
Elemento: 9
Elemento: 5
Elemento: 6
Elemento: 7
Elemento: 3
Elemento: 8
Elemento:
Ingrese el elemento K a insertar: 100
Pila modificada:
9
5
100
6
7
3
8

Process finished with exit code 0

```

3. Eliminar el primer elemento par.

P (9, 7, 4, 5, 6, 2) => P (9,7,5,6,2)

Código:

```

import java.util.Scanner;
import java.util.Stack;

public class EJERCICIO_3 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        Stack<Integer> pila = new Stack<Integer>();

        System.out.println("Ingrese elementos:");
        while (true) {
            System.out.print("Elemento: ");
            String entrada = scanner.nextLine().trim();

            if (entrada.equalsIgnoreCase("")) {
                break;
            }
            try {
                int elemento = Integer.parseInt(entrada);
                pila.push(elemento);
            } catch (NumberFormatException e) {
                System.out.println("Entrada no válida.
Introduzca un número entero.");
            }
        }
    }
}

```

```

    }
}
Stack<Integer> pilaModificada = new
Stack<Integer>();
boolean primerElementoParEliminado = false;

while (!pila.isEmpty()) {
    int elemento = pila.pop();
    if (!primerElementoParEliminado && elemento % 2
== 0) {
        primerElementoParEliminado = true;
    } else {
        pilaModificada.push(elemento);
    }
}

if (primerElementoParEliminado) {
    System.out.println("Pila modificada:");
    while (!pilaModificada.isEmpty()) {
        System.out.println(pilaModificada.pop());
    }
} else {
    System.out.println("No se encontraron elementos
pares en la pila.");
}
}
}

```

Ejecucion del código:

```

EJERCICIO_3 x
"C:\Users\HP-Ryzen 5\.jdk\openjdk-20\bin\java.exe"
Ingrese elementos:
Elemento: 2
Elemento: 6
Elemento: 5
Elemento: 4
Elemento: 7
Elemento: 9
Elemento:
Pila modificada:
2
6
5
7
9
Process finished with exit code 0

```

COLAS EJERCICIOS.

4. Sumar elementos impares divisibles entre 3

C (11, 6, 3, 4, 15, 2) ==> sum=3+15=18

Código:

```
import java.util.LinkedList;
import java.util.Queue;
import java.util.Scanner;

public class EJERCICIO_4 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        Queue<Integer> cola = new LinkedList<>();

        System.out.println("Ingrese elementos:");
        while (true) {
            System.out.print("Elemento: ");
            String entrada = scanner.nextLine().trim();

            if (entrada.equalsIgnoreCase("")) {
                break;
            }

            try {
                int elemento = Integer.parseInt(entrada);
                cola.add(elemento);
            } catch (NumberFormatException e) {
                System.out.println("Entrada no válida.
Introduzca un número entero.");
            }
        }

        int suma = 0;

        while (!cola.isEmpty()) {
            int elemento = cola.poll();

            if (elemento % 2 != 0 && elemento % 3 == 0) {
                suma += elemento;
            }
        }

        System.out.println("Suma de elementos impares
divisibles entre 3: " + suma);
    }
}
```

Ejecución del código:

```
EJERCICIO_4 x
"C:\Users\HP-Ryzen 5\.jdk\openjdk-20\bin\java.exe"
Ingrese elementos:
Elemento: 11
Elemento: 6
Elemento: 3
Elemento: 4
Elemento: 15
Elemento: 2
Elemento:
Suma de elementos impares divisibles entre 3: 18

Process finished with exit code 0
```

5. Duplicar el elemento K.

C (4, 7, 1, 4, 6) K= 7 => C (4, 14, 1, 4, 6)

C (7, 11, 5, 3) K=10 => "No se encontró"

Código:

```
import java.util.LinkedList;
import java.util.Queue;
import java.util.Scanner;

public class EJERCICIO_COLAS_5 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        Queue<Integer> cola = new LinkedList<>();

        System.out.println("Ingrese elementos:");
        while (true) {
            System.out.print("Elemento: ");
            String entrada = scanner.nextLine().trim();

            if (entrada.equalsIgnoreCase("")) {
                break;
            }

            try {
                int elemento = Integer.parseInt(entrada);
                cola.add(elemento);
            } catch (NumberFormatException e) {
                System.out.println("Entrada no válida.
Introduzca un número entero.");
            }
        }

        System.out.print("Ingrese el elemento K a duplicar:
```

```

");
    int k = scanner.nextInt();

    Queue<Integer> colaModificada = new LinkedList<>();

    boolean encontrado = false; // Variable para
    verificar si se encontró el elemento K

    for (int elemento : cola) {
        if (elemento == k) {
            colaModificada.add(elemento * 2);
            encontrado = true; // Se encontró el
elemento K
        } else {
            colaModificada.add(elemento);
        }
    }
    if (encontrado) {
        System.out.println("Cola modificada:");
        for (int elemento : colaModificada) {
            System.out.println(elemento);
        }
    } else {
        System.out.println("No se encontró el elemento
K en la cola.");
    }
}
}

```

Ejecución del código:

```

EJERCICIO_COLAS_5 x
"C:\Users\HP-Ryzen 5\.jdk\openjdk-20\bin\java
Ingrese elementos:
Elemento: 7
Elemento: 11
Elemento: 5
Elemento: 3
Elemento:
Ingrese el elemento K a duplicar: 10
No se encontró el elemento K en la cola.

Process finished with exit code 0

```



```
"C:\Users\HP-Ryzen 5\.jdk\openjdk-20\bin\j
Ingrese elementos:
Elemento: 4
Elemento: 7
Elemento: 1
Elemento: 4
Elemento: 6
Elemento:
Ingrese el elemento K a duplicar: 7
Cola modificada:
4
14
1
4
6

Process finished with exit code 0
```