

Monitoria *Sesión*2.0

Algoritmos y Estructuras de Datos

Juan Pablo Sierra Useche

Universidad del Rosario

15.08.2020

Plan de Acción

1 Antes de empezar con lo bueno. . .

2 Stacks

3 Queue

4 Mi quiz del semestre pasado.

5 Ñapa. . .

└─ Antes de empezar con lo bueno. . .

TODO:

└─ Antes de empezar con lo bueno. . .

TODO:

- 1) Leer bien el problema.

└─ Antes de empezar con lo bueno. . .

TODO:

- 1) Leer bien el problema.
- 2) Separar las instrucciones.

TODO:

- 1) Leer bien el problema.
- 2) Separar las instrucciones.
- 3) Desarrollarlas por separado.

TODO:

- 1) Leer bien el problema.
- 2) Separar las instrucciones.
- 3) Desarrollarlas por separado.
- 4) Únificar.

TODO:

- 1) Leer bien el problema.
- 2) Separar las instrucciones.
- 3) Desarrollarlas por separado.
- 4) Únificar.
- 5) Probar y corregir errores.

TODO:

- 1) Leer bien el problema.
- 2) Separar las instrucciones.
- 3) Desarrollarlas por separado.
- 4) Únificar.
- 5) Probar y corregir errores.
- 6) **Disfrutarlo**. . . Eres lo máximo por haber solucionado el problema.

Plan de Acción

1 Antes de empezar con lo bueno...

2 Stacks

3 Queue

4 Mi quiz del semestre pasado.

5 Ñapa...

Ejercicio 1

instrucciones:

Escriba un programa que lea una secuencia de caracteres y los imprima en orden inverso. Para esto use un **stack** de caracteres.

Ejercicio 1

instrucciones:

Escriba un programa que lea una secuencia de caracteres y los imprima en orden inverso. Para esto use un **stack** de caracteres.

Ejercicio 1

instrucciones:

Escriba un programa que lea una secuencia de caracteres y los imprima en orden inverso. Para esto use un **stack** de caracteres.

Ejercicio 1

PA' LOS Hardcore

Hacer un programa que **lea un archivo** y utilice el programa anterior, para imprimir cada palabra del texto al revés **de forma individual**.

Ejercicio 2

instrucciones:

escriba un programa que reciba un **stack** de numeros enteros arbitrarios y que retorne el mismo stack pero donde solo los elementos que son menores a los elementos anteriores permanezcan en el **stack**.

$[10, 8, 11, 4, 7, 2, 1] \rightarrow [10, 8, 4, 2, 1]$

Ejercicio 2

instrucciones:

Escriba un programa que reciba un **stack** de numeros enteros arbitrarios y que retorne el mismo stack pero donde solo los elementos que son menores a los elementos anteriores permanezcan en el **stack**.

$[10, 8, 11, 4, 7, 2, 1] \rightarrow [10, 8, 4, 2, 1]$

Plan de Acción

1 Antes de empezar con lo bueno...

2 Stacks

3 Queue

4 Mi quiz del semestre pasado.

5 Ñapa...

Ejercicio 3

instrucciones:

Usando las operaciones estándar de un **queue**, escriba un programa que elimina el elemento medio de una cola de números de doble precisión.

Ejercicio 3

instrucciones:

Usando las operaciones estándar de un **queue**, escriba un programa que **elimina el elemento medio de una cola de números de doble precisión**.

Ejercicio 3

instrucciones:

Usando las operaciones estándar de un **queue**, escriba un programa que elimina el elemento medio de una cola de números de doble precisión.

Ejercicio 3

instrucciones:

Usando las operaciones estándar de un **queue**, escriba un programa que elimina el elemento medio de una cola de números de doble precisión.

Ejercicio 3

PÆ' L05 QU3 T€NG4N G@Na5 D +

Modificar el programa para que retorne un **queue** con el elemento de la mitad en la primera posición.

Ejercicio 4

instrucciones:

En el banco de Chía ubicado en el centro comercial más concurrido del municipio tienen el problema de que la gente suele colarse muy seguido, para solucionar este problema licitaron con varios grupos de expertos en estructuras de datos; de todos ellos escogieron dos: el primero debe entregar un algoritmo que al conectarse con las cámaras clasifique con un número impar a aquellos que se hayan colado, y el segundo grupo debe encargarse de reconocer la posición de aquellos que se colaron (para que así el guardia los detecte) y además los debe sacar de la estructura de datos. Como estamos en pandemia y la Universidad no nos puede prestar las cámaras para probar el primer algoritmo se nos encargo hacer el segundo.

Plan de Acción

1 Antes de empezar con lo bueno...

2 Stacks

3 Queue

4 Mi quiz del semestre pasado.

5 Ñapa...

Ejercicio 5

instrucciones:

Escriba la función:

`int` howMany(`list`<`int`> `L`, `int` `k`)

que recibe una lista de enteros **L** y un entero **k**. Mediante un ciclo **while**, la función debe encontrar el número de veces que se encuentra **k** en **L** y retornar esta cantidad. Puede modificar la lista si lo considera necesario.

Ejercicio 5

instrucciones:

Escriba la función:

`int` howMany(`list`<`int`> `L`, `int` `k`)

que recibe una lista de enteros `L` y un entero `k`. Mediante un ciclo **while**, la función debe encontrar el número de veces que se encuentra `k` en `L` y retornar esta cantidad. Puede modificar la lista si lo considera necesario.

Ejercicio 5

instrucciones:

Escriba la función:

`int` howMany(`list`<`int`> `L`, `int` `k`)

que recibe una lista de enteros `L` y un entero `k`. Mediante un ciclo `while`, la función debe encontrar el número de veces que se encuentra `k` en `L` y retornar esta cantidad. Puede modificar la lista si lo considera necesario.

Ejercicio 5

instrucciones:

Escriba la función:

```
int howMany(list<int> L,int k)
```

que recibe una lista de enteros **L** y un entero **k**. Mediante un ciclo **while**, la función debe encontrar el número de veces que se encuentra **k** en **L** y retornar esta cantidad. Puede modificar la lista si lo considera necesario.

Ejercicio 5

instrucciones:

Escriba la función:

```
int howMany(list<int> L,int k)
```

que recibe una lista de enteros **L** y un entero **k**. Mediante un ciclo **while**, la función debe encontrar el número de veces que se encuentra **k** en **L** y retornar esta cantidad. Puede modificar la lista si lo considera necesario.

Plan de Acción

1 Antes de empezar con lo bueno...

2 Stacks

3 Queue

4 Mi quiz del semestre pasado.

5 Ñapa...

■ ■ ■

< algo muuuuuus? >

.....

$$\begin{array}{c} \wedge \quad \wedge \\ \backslash \quad (oo) \backslash \text{-----} \\ (\text{---}) \backslash \quad) \backslash \backslash \\ \quad \quad | \quad | \text{----} w \quad | \\ \quad \quad | \quad | \quad \quad | \quad | \end{array}$$