



Universidad Simón Bolívar  
Departamento de Computación y Tecnología de la Información  
Computación II (CI-2126)

## QUIZ 2 (guía de soluciones)

1.- Considere las siguientes funciones del *TDA Cola*:

- \*.- cola **Crear\_cola** ( )
- \*.- lógico **Borrar\_cola** ( $\diamond c$ : cola)
- \*.- lógico **Es\_vacia** (c: cola)
- \*.- lógico **Esta\_llena** (c: cola)
- \*.- cola **Encolar** ( $\diamond c$ : cola, e: Entero)
- \*.- entero **Desencolar** ( $\diamond c$ : cola)
- \*.- entero **Tamano\_cola** (c: cola)
- \*.- cola **Vaciar\_cola** ( $\diamond c$ : cola)
- \*.- cola **Copiar\_cola** ( $\diamond c1$ : cola, c2: cola)
- \*.- **Imprimir\_cola** (c: cola)

donde “ $\diamond$ ” indica que el objeto puede ser modificado en su contenido

Se desea que desarrolle un *Algoritmo* que procese dos instancias del *TDA Cola* y produzca otro como salida. Para ello, su respuesta inicialmente deberá recibir como entrada los objetos Cola 1 (C1) y Cola 2 (C2) de números enteros, y luego deberá producir un nuevo objeto, Cola 3 (C3), que estará formada por la intercalación de los elementos de C1 y C2, en ese orden. Considere además que el tamaño de C1 no necesariamente es igual al de C2, en ese caso, la intercalación deberá realizarse hasta donde sea posible. Cuide además, de que las colas originales C1 y C2, no reciban ninguna alteración en sus contenidos y que ello pueda verificarse a través de su solución al problema. (3 puntos)

### **Respuesta:**

Comienzo

```
c1 ← Crear_cola ( )
c2 ← Crear_cola ( )
c3 ← Crear_cola ( )
c1 ← Llenar_cola (c1)
c2 ← Llenar_cola (c2)
Imprimir_cola (c1)
Imprimir_cola (c2)
caux ← Crear_cola ( )
caux2 ← Crear_cola ( )
caux ← Copiar_cola (caux, c1)
caux2 ← Copiar_cola (caux, c2)
Si (Tamano_cola (caux) = Tamano_cola (caux2)) entonces
    Mientras (Esta_llena (caux))
```

c3 ← Encolar (c3, Desencolar (caux))  
c3 ← Encolar (c3, Desencolar (caux2))

Fmientras

Sino

Si (TamanoCola (caux) > TamanoCola (caux2)) entonces

Mientras (Esta\_llena (caux2))

c3 ← Encolar (c3, Desencolar (caux))

c3 ← Encolar (c3, Desencolar (caux2))

Fmientras

Mientras (Esta\_llena (caux))

c3 ← Encolar (c3, Desencolar (caux))

Fmientras

Sino

Mientras (Esta\_llena (caux))

c3 ← Encolar (c3, Desencolar (caux))

c3 ← Encolar (c3, Desencolar (caux2))

Fmientras

Mientras (Esta\_llena (caux2))

c3 ← Encolar (c3, Desencolar (caux))

Fmientras

Fsi

Fsi

Si (¬BorrarCola (caux)) entonces

Mensaje\_de\_que\_algo\_salió\_mal ← Imprimir\_en\_pantalla

Fsi

Si (¬BorrarCola (caux2)) entonces

Mensaje\_de\_que\_algo\_salió\_mal ← Imprimir\_en\_pantalla

Fsi

ImprimirCola (c1)

ImprimirCola (c2)

ImprimirCola (c3)

Si (¬BorrarCola (c1)) entonces

Mensaje\_de\_que\_algo\_salió\_mal ← Imprimir\_en\_pantalla

Fsi

Si (¬BorrarCola (c2)) entonces

Mensaje\_de\_que\_algo\_salió\_mal ← Imprimir\_en\_pantalla

Fsi

Si (¬BorrarCola (c3)) entonces

Mensaje\_de\_que\_algo\_salió\_mal ← Imprimir\_en\_pantalla

Fsi

Fin

cola Función LlenarCola (∅c: cola)

Mientras (¬Esta\_llena (c))

Mensaje\_solicitando\_valor ← Imprimir\_en\_pantalla

dato ← Leer\_entero\_del\_teclado

c ← Encolar (c, dato)

Fmientras

Si (Es\_vacia (c)) entonces

Mensaje\_de\_que\_algo\_salió\_mal ← Imprimir\_en\_pantalla

Abortar\_programa

sino

Devolver (c)

Fsi

Efunción LlenarCola

Nota1: “¬” indica el operador de negación

Nota2: La verificación extraordinaria en la función “LlenarCola” se hace por si se produjeron entradas inválidas

2.- Considere el TDA Pila de cadenas de caracteres, con las operaciones “Crear\_pila”, “Borrar\_pila”, “Tope\_pila” -indica la posición del último elemento ocupado de la pila-, “Base\_pila” -indica la posición del fondo de la pila-, “Apilar”, “Des-empilar”, “Longitud\_pila”, “Es\_vacia\_pila”, “Imprimir\_pila”.

2.1.- Especifique sintácticamente el TDA Pila de cadena de caracteres. (1 punto)

2.2.- Luego, se desea que elabore un Algoritmo que construya una pila de cadena de caracteres y luego la invierta. Para facilitar su respuesta debe considerar que las cadenas serán del mismo tamaño. (1 punto)

**Solución:**

2.1- Especificación Sintáctica del TDA Pila [Cadena de caracteres]

Crear\_pila:  $\rightarrow$  Pila  
Borrar\_pila: Pila  $\rightarrow$   
Es\_vacia\_pila: Pila  $\rightarrow$  lógico  
Apilar: Pila  $\times$  Cadena de caracteres  $\rightarrow$  Pila  
Des-empilar: Pila  $\rightarrow$  Pila  
Tope\_pila: Pila  $\rightarrow$  dirección de memoria  
Base\_pila: Pila  $\rightarrow$  dirección de memoria  
Longitud\_pila: Pila  $\rightarrow$  entero  
Imprimir\_pila: Pila  $\rightarrow$

2.2.- Algoritmo:

Comienzo

p1  $\leftarrow$  Crear\_pila ( $\emptyset$ )  
p1  $\leftarrow$  Llenar\_pila (p1)  
Imprimir\_pila (p1)  
  
p2  $\leftarrow$  Crear\_pila ( $\emptyset$ )  
Mientras ( $\neg$ Esta\_llena (p1))  
    p2  $\leftarrow$  Apilar (p2, Des-empilar (p1))  
Fmientras  
Imprimir\_pila (p2)  
  
Borrar\_pila (p1)  
Borrar\_pila (p2)

Fin

pila Función Llenar\_pila ( $\emptyset p$ : pila)

Mientras ( $\neg$ Es\_vacia\_pila (p))  
    Mensaje solicitando cadena  $\leftarrow$  Imprimir\_en\_pantalla  
    cadena  $\leftarrow$  Leer\_cadena\_de\_caracteres\_del\_teclado  $\square$   
    p  $\leftarrow$  Apilar (p, cadena)  
Fmientras  
Devolver (p)

Efunción Llenar\_pila

Nota1: “ $\neg$ ” indica el operador de negación