



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor: TISTA GARCÍA EDGAR

Asignatura: ESTRUCTURA DE DATOS Y ALGORITMOS I

Grupo: 1

No de Práctica(s): 1

Integrante(s): GÓMEZ LUNA ALEJANDRO

*No. de Equipo de
cómputo empleado* 34

Semestre: 2019-2

Fecha de entrega: 11 de Febrero de 2019

Obervaciones:

CALIFICACIÓN: _____

- **Objetivo de la práctica**

Utilizar arreglos unidimensionales y multidimensionales para dar solución a problemas computacionales.

- **Desarrollo**

Análisis del programa propuesto en la guía de laboratorio “Escítala Espartana”:

El programa sirve para cifrar y descifrar mensajes mediante una escítala espartana, el cual fue uno de los primeros métodos criptográficos usado por los espartanos para poder mandar mensajes que solo pudieran ser descifrados por generales durante la guerra.

En el momento en que el programa se inicia, podemos observar que nos aparece un menú en donde nosotros podemos seleccionar que queremos seleccionar, ingresando el número correspondiente a la acción que queramos hacer.

La opción número 1 nos permite crear un mensaje para cifrarlo, pidiéndonos el número de renglones y de columnas que usaremos para la encriptación. Posteriormente se nos solicita ingresar el texto que queremos cifrar. El programa guarda en un arreglo bidimensional, de manera ordenada, letra por letra del texto que nosotros ingresamos, haciéndolo primero por columnas y posteriormente por renglones, para finalmente mostrarnos como quedaría la “tira” del mensaje cifrado. Para lograrlo, el programa obtiene letra por letra del arreglo bidimensional, primero por renglones y posteriormente por columnas.

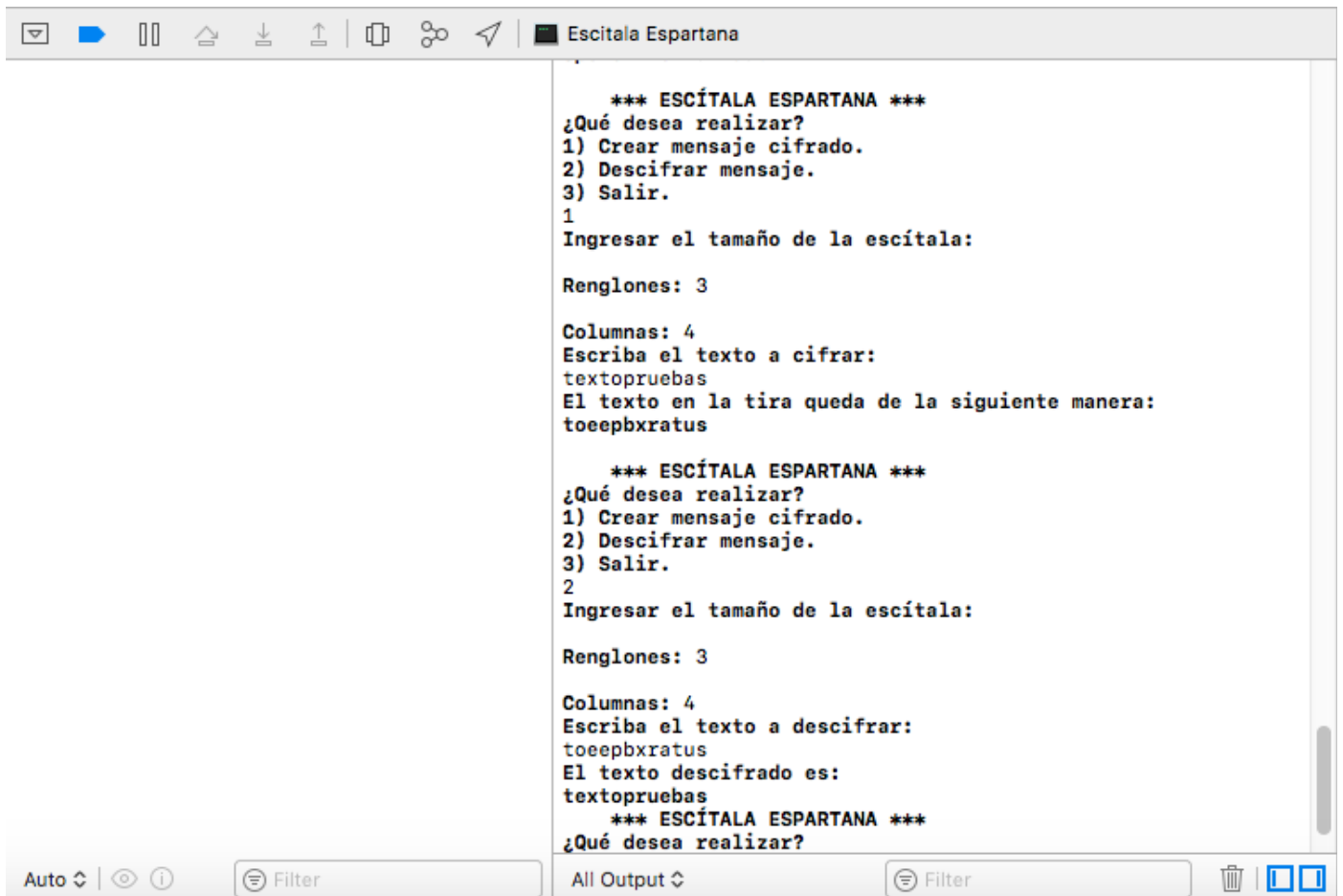
La opción número 2 nos permite descifrar un mensaje, pidiéndonos que especifiquemos la cantidad de columnas y renglones que se utilizarán, así como el texto tal cual viene en la “tira”. El programa guarda en un arreglo bidimensional, de manera ordenada, letra por letra del texto que nosotros ingresamos, haciéndolo primero por renglones y posteriormente por columnas, para finalmente mostrarnos como quedaría el texto descifrado. Para lograrlo, el programa obtiene letra por letra del arreglo bidimensional, primero por columnas y posteriormente por renglones.

La opción número 3 nos permite salir del programa.

Escribiendo cualquier otro carácter nos mostrará un mensaje de opción no válida.

El programa es conversacional, por lo que siempre se estará repitiendo a menos que nosotros seleccionemos la opción de salir.

El programa funciona bastante bien hasta que se ingresan textos que incluyen más de una palabra, en donde no se realiza correctamente el cifrado ni el descifrado del mensaje.



```
*** ESCÍTALA ESPARTANA ***
¿Qué desea realizar?
1) Crear mensaje cifrado.
2) Descifrar mensaje.
3) Salir.
1
Ingresar el tamaño de la escítala:

Renglones: 3

Columnas: 4
Escriba el texto a cifrar:
textopruebas
El texto en la tira queda de la siguiente manera:
toeepbxratus

*** ESCÍTALA ESPARTANA ***
¿Qué desea realizar?
1) Crear mensaje cifrado.
2) Descifrar mensaje.
3) Salir.
2
Ingresar el tamaño de la escítala:

Renglones: 3

Columnas: 4
Escriba el texto a descifrar:
toeepbxratus
El texto descifrado es:
textopruebas
*** ESCÍTALA ESPARTANA ***
¿Qué desea realizar?
```

Ejercicios propuestos

Ejercicio 1: El programa nos muestra los datos almacenados en un arreglo tridimensional, en donde los datos son proporcionados desde un inicio. El único error con el código es que al momento de mostrar los resultados, estos se muestran en un orden incorrecto, debido a que el ciclo más anidado se repite 6 veces, cuando se debería de repetir solamente 3, y el ciclo más exterior se repite 3 veces, cuando se debería repetir 6 veces.

Ejercicio 2: Se encontraron varios errores de sintaxis, principalmente hacían falta ; al final de algunas sentencias, en la declaración del ciclo for se usaba , en vez de ; , la variable "i" no estaba bien declarada, había una instrucción que tenía que ser "printf" en vez de solamente "print", para imprimir el resultado se estaba utilizando un signo "+" en vez de usar una coma para y no imprime el resultado de lista2. Además, existe un error de coherencia al momento de asignar valores de lista 1 a lista 2, ya que al asignarle el último valor a lista 2, lista 1 ya no tiene más elementos y se está haciendo referencia a una posición que queda fuera del arreglo de lista 1, debido a la instrucción "i+1".

Cuando se imprimen los valores de lista 2, aparecen los mismos elementos de lista 1 pero multiplicados por dos, es decir, los valores de lista 2 son el doble de los valores de lista 1.

Por último, cree otro arreglo en donde solamente estuvieran los números que son divisibles entre 4 y posteriormente sume los elementos de dicho arreglo

Ejercicio 3:

Evidencia de la implementación

- Ejercicio 1:

```
Arreglo[0][0][0]: 3
Arreglo[0][0][1]: 5
Arreglo[0][0][2]: 7
Arreglo[0][1][0]: 9
Arreglo[0][1][1]: 11
Arreglo[0][1][2]: 13
Arreglo[1][0][0]: 15
Arreglo[1][0][1]: 17
Arreglo[1][0][2]: 19
Arreglo[1][1][0]: 21
Arreglo[1][1][1]: 23
Arreglo[1][1][2]: 25
Arreglo[2][0][0]: 27
Arreglo[2][0][1]: 29
Arreglo[2][0][2]: 31
Arreglo[2][1][0]: 33
Arreglo[2][1][1]: 35
Arreglo[2][1][2]: 37
Arreglo[3][0][0]: 39
Arreglo[3][0][1]: 41
Arreglo[3][0][2]: 43
Arreglo[3][1][0]: 45
Arreglo[3][1][1]: 47
Arreglo[3][1][2]: 49
Arreglo[4][0][0]: 51
Arreglo[4][0][1]: 53
Arreglo[4][0][2]: 55
Arreglo[4][1][0]: 57
Arreglo[4][1][1]: 59
Arreglo[4][1][2]: 61
Arreglo[5][0][0]: 63
Arreglo[5][0][1]: 65
Arreglo[5][0][2]: 67
Arreglo[5][1][0]: 69
Arreglo[5][1][1]: 71
Arreglo[5][1][2]: 73
Program ended with exit code: 0
```

- Ejercicio 2:

```
10,20,30,40,50,60,70,80,90,100,
20,40,60,80,100,120,140,160,180,200,
El valor es: 550
El valor de los elementos divisibles entre 4 es: 300
Program ended with exit code: 54
```

All Output ⌵

Filter



- Ejercicio 3:
- Conclusiones de la práctica