

Universidad Nacional del Litoral
Facultad de Ingeniería y Ciencias Hídricas
Departamento de Informática



Ingeniería Informática

FUNDAMENTOS DE PROGRAMACIÓN

UNIDAD 9 **Arrays y Structs**

Guía de trabajos prácticos
2017

UNIDAD 9

Arrays y Structs

Ejercicio 9.1

Leer las calificaciones de un grupo de 30 alumnos que asistieron a una evaluación de programación. Mostrar las notas de los alumnos que se encuentren por encima del promedio.

Ejercicio 9.2

Generar aleatoriamente un arreglo lineal de 20 elementos numéricos, con enteros aleatorios entre 100 y 150 y mostrarlo en pantalla. Luego ingresar dos valores en las variables m y p . El valor m debe ser insertado en la posición p del arreglo. Mostrar el vector modificado.

Ejercicio 9.3

Se lee un arreglo ordenado en forma creciente de 100 enteros y se desea insertar un nuevo elemento de forma que se preserve el orden. Codifique dos funciones llamadas *busca_pos()* e *inserta()*. La primera debe localizar y devolver la posición en la que debería insertarse un elemento para mantener el arreglo ordenado, mientras que la segunda debe recibir el elemento y la posición e insertarlo en el arreglo.

Ejercicio 9.4

Leer los nombres y calificaciones de un grupo de alumnos que asistieron a una evaluación de programación. Los datos finalizan con la calificación 0. Generar un vector con los nombres de los alumnos aprobados (calificación ≥ 6) y otro con los nombres de los no aprobados. Mostrar luego ambos vectores.

Ejercicio 9.5

Leer en un arreglo lineal una lista de N datos numéricos. Eliminar del arreglo el valor 523. Si este dato aparece más de una vez eliminar todas las ocurrencias y mostrar el modificado. Para hacerlo, codifique y utilice una función llamada *buscar()* que devuelva la posición que un determinado elemento ocupa dentro del arreglo, y otra función *eliminar()* que borre de un arreglo el elemento ubicado en una posición especificada.

Ejercicio 9.6

Una asociación cooperadora escolar recibe aportes de dinero variable de los estudiantes asociados. Se leen sin orden alguno los montos aportados durante un año, y la fecha correspondiente (día y mes). Estos datos terminan con el valor de monto cero. Informe: a) el total recaudado por mes. b) El mes de menor aporte.

Ejercicio 9.7

En una olimpiada estudiantil compiten N alumnos en 3 pruebas (1: matemáticas, 2: física y 3: computación). Se ingresan por cada inscripto el DNI y su número asignado para la competencia (entre 1 y N). Luego, sin orden alguno, se van ingresando ternas con los puntajes de cada prueba: número de participante, código de prueba, y puntaje en la actividad. Escriba un algoritmo que determine: a) El DNI del ganador de la competencia y el puntaje total obtenido. b) El DNI del estudiante que ocupó el 2do lugar y su puntaje. c) ¿Qué puntaje obtuvo en Computación el ganador de la competencia?

Ejercicio 9.8

Una empresa distribuidora comercializa 25 artículos. Posee 4 sucursales y desea analizar el desempeño de las mismas. Para ello se ingresan los datos correspondientes a las ventas efectuadas en cierto período: código sucursal (1...4), código artículo (1...25), cantidad unidades vendidas. Determine e informe:

- Las cantidades de unidades vendidas por la empresa de cada artículo.
- El total de unidades vendidas por la sucursal 3, sumando todos los artículos.
- La cantidad vendida por la sucursal 1 del artículo 6.
- La sucursal que vendió más unidades del artículo 8.

Ejercicio 9.9

Considere el mismo enunciado del problema anterior. Incorpore -además- los datos de los precios de los 25 artículos que comercializa la empresa. Determine e informe:

- La recaudación de cada sucursal.
- La recaudación de la empresa.
- La sucursal que obtuvo mayor recaudación.

Ejercicio 9.10

Se leen los datos de una matriz o tabla de 5x12 que contiene los valores en mm de lluvias producidas en 5 departamentos de la provincia durante los 12 meses de 2005. Se sabe que en algunos departamentos tienen datos faltantes por lo que aparecen valores negativos (-1) en cada uno de esos casos. Escriba un programa

C++ que: a) complete los datos de lluvia faltantes con el promedio anual de precipitación en el departamento correspondiente. Cada promedio debe obtenerse con los datos reales existentes (no contar los -1). b) Amplíe la matriz agregando una nueva columna con los totales de mm caídos en todo el año en cada departamento. Informe: 1) la matriz original de datos, 2) la matriz modificada y 3) los totales anuales de lluvia por departamento.

Ejercicio 9.11

Declare un tipo struct con el nombre `ficha_alumno` indicando al menos 10 campos de tipo simple o string correspondientes a los datos de un alumno de la UNL. Luego declare 3 variables de ese tipo.

Ejercicio 9.12

Considere el struct `ficha_alumno` del ejercicio anterior y realice lo siguiente:

- Agregue al tipo `ficha_alumno` un campo llamado `materias` el cual debe almacenar las calificaciones (de 1 a 10) de las asignaturas aprobadas por cada alumno en su carrera. Considere que la carrera consta de 32 materias.
- Indique la sintaxis para asignar las calificaciones 8, 7 y 10 a las materias 6, 11 y 12 respectivamente a los miembros correspondientes de una variable de tipo `ficha_alumno`.

Ejercicio 9.13

Declare una variable `x` con la cual pueda representar en un programa C++ a una lista de hasta 200 alumnos, cuyos datos se organizan en la estructura tipo `ficha_alumno` que creó antes.

Ejercicio 9.14

Un número complejo tiene la forma $a+bi$ donde a y b son números reales. Las operaciones básicas con complejos son: Suma, Resta, Producto y Cociente: Escriba un programa C++ que permita ingresar las componentes de 2 números complejos y a través de un menú el usuario seleccione la operación a realizar. Proponga funciones para cada operación de complejos, que permitan devolver el complejo resultante. Proponga una struct llamado complejo de 2 miembros: `parte_real` y `parte_imag` para representar y operar los números complejos.

Ejercicio 9.15

Considere el struct complejo del ejercicio anterior. Escriba una función C++ llamada `opera_complex()` cuyo prototipo se indica abajo:

```
complejo opera_complex(complejo c1, complejo c2, char op);
```

Su objetivo es recibir 2 complejos como parámetros y un carácter que indique la operación: '+', '-', '*' . Debe devolver el resultado de la operación.

Cuestionario

1. ¿Qué tipos de datos se admiten como índices de una estructura de tipo array en C++?
2. ¿Cómo se indica en C++ la dimensión de un arreglo al pasarlo como parámetro en la llamada de una función?
3. ¿Si un arreglo es pasado como parámetro a una función, y es modificado dentro de esta. ¿Qué ocurre con el arreglo empleado en la llamada de la función?
4. ¿Cuántos bytes ocupan en memoria los siguientes arreglos declarados como sigue?. Nota: los datos int y float ocupan 4 bytes.
 - a) `int x[]={12,56,78};`
 - b) `float nuevo[10]={2.34, 4.51, 9.02};`
 - c) `char a[];`
5. Es posible organizar un arreglo bidimensional en un arreglo lineal. Si responde por sí ¿cómo ubica a cada elemento?
6. ¿Los arreglos ocupan siempre la misma cantidad de memoria durante la ejecución de un programa ?
7. ¿Qué ocurre si accede a un elemento de un arreglo que no fue inicializado todavía?
8. ¿Es posible declarar una instancia de un tipo struct dentro de otro struct? Si responde por sí proponga un ejemplo.
9. ¿Qué diferencia encuentra entre un struct y un array? ¿Cuándo conviene emplear cada uno?
10. ¿Puede una función retornar un struct? ¿Y retornar un arreglo? Si responde por sí proponga un ejemplo en cada caso.
11. ¿Es posible pasar por valor un arreglo a una función? ¿Y pasar un struct por referencia? Explique.

