

1. **(Eliminación de duplicados) 25P.** Use un arreglo unidimensional para resolver el siguiente problema: escriba una aplicación que reciba como entrada cinco números, cada uno de los cuales debe estar entre 10 y 100, inclusive. A medida que se lea cada número, muéstrelo solamente si no es un duplicado de un número que ya se haya leído. Prepárese para el “peor caso”, en el que los cinco números son diferentes. Use el arreglo más pequeño que sea posible para resolver este problema. Muestre el conjunto completo de valores únicos introducidos, después de que el usuario introduzca cada nuevo valor.

2. **(Sistema de reservaciones de una aerolínea). 25P.** Una pequeña aerolínea acaba de comprar una computadora para su nuevo sistema de reservaciones automatizado. Se le ha pedido a usted que desarrolle el nuevo sistema. Usted escribirá una aplicación para asignar asientos en cada vuelo del único avión de la aerolínea (capacidad: 10 asientos). Su aplicación debe mostrar las siguientes alternativas: Por favor escriba 1 para Primera Clase y Por favor escriba 2 para Económico. Si el usuario escribe 1, su aplicación debe asignarle un asiento en la sección de primera clase (asientos 1 a 5). Si el usuario escribe 2, su aplicación debe asignarle un asiento en la sección económica (asientos 6 a 10). Su aplicación deberá entonces imprimir un pase de abordar, indicando el número de asiento de la persona y si se encuentra en la sección de primera clase o clase económica. Use un arreglo unidimensional del tipo primitivo boolean para representar la tabla de asientos del avión. Inicialice todos los elementos del arreglo con false para indicar que todos los asientos están vacíos. A medida que se asigne cada asiento, establezca el elemento correspondiente del arreglo en true para indicar que ese asiento ya no está disponible. Su aplicación nunca deberá asignar un asiento que ya haya sido asignado. Cuando esté llena la sección económica, su programa deberá preguntar a la persona si acepta ser colocada en la sección de primera clase (y viceversa). Si la persona acepta, haga la asignación de asiento apropiada. Si no, imprima el mensaje “El próximo vuelo sale en 3 horas”.

3. **(Ventas totales) 25P.** Use un arreglo bidimensional para resolver el siguiente problema: una compañía tiene cuatro vendedores (1 a 4) que venden cinco productos distintos (1 a 5). Una vez al día, cada vendedor pasa una nota por cada tipo de producto vendido. Cada nota contiene lo siguiente:

- El número del vendedor
- El número del producto
- El valor total en dólares de ese producto vendido en ese día

Así, cada vendedor pasa entre 0 y 5 notas de venta por día. Suponga que está disponible la información sobre todas las notas del mes pasado. Escriba una aplicación que lea toda esta información para las ventas del último mes y que resuma las ventas totales por vendedor, y por producto. Todos los totales deben guardarse en el arreglo bidimensional ventas. Después de procesar toda la información del mes pasado, muestre los resultados en formato tabular, en donde cada columna represente a un vendedor específico y cada fila simbolice un producto. Saque el total de cada fila para obtener las ventas totales de cada producto durante el último mes. Calcule el total de cada columna para sacar las ventas totales de cada vendedor durante el último mes. Su impresión tabular debe incluir estos totales cruzados a la derecha de las filas totalizadas, y en la parte inferior de las columnas totalizadas.

4. **(Serie de Fibonacci) 25P.** La serie de Fibonacci

0,1,1,2,3,5,8,13,21,...

Empieza con los términos 0 y 1, y tiene la propiedad de que cada término sucesivo es la suma de los dos términos anteriores.

- Escriba un método llamado fibonacci(n) que calcule el enésimo número de Fibonacci. Incorpore este método en una aplicación que permita al usuario introducir el valor de n.
- Determine el número de Fibonacci más grande que puede imprimirse en su sistema.
- Modifique la aplicación que escribió en la parte (a), de manera que utilice double en vez de int para calcular y devolver números de Fibonacci, y utilice esta aplicación modificada para repetir el inciso (b)