

Algoritmos y Programación

Práctica 5

- 1) Implemente la clase horario que permite guardar día, hora y materia a cursar en ese horario
- 2) Implemente la clase Alumno para modelar alumnos en una Universidad. Agréguele como estado lo que considere necesario para el problema. Piense ¿cómo implementaría los horarios de cursada de un alumno? (debe usar la clase del ejercicio anterior)
Como comportamiento implemente el método agregarMateria(string dia, string hora, string materia) que agrega a los horarios del alumno la materia en ese día y hora, si es que tiene libre el horario.
Piense qué otros métodos podría definir.
En una aplicación arme una lista de alumnos con sus horarios de cursada. Defina un menú que permita inscribir alumnos en una materia, imprimir los alumnos inscriptos en una materia dada, informar cuántas materias cursa un alumno e imprimir la lista de todos los alumnos.
Observación: la lista de alumnos puede ser un Array o un ArrayList (no hay que definir otra clase nueva)
- 3) Implemente la clase Avión suponiendo que se conoce el nro de vuelo, la empresa aérea, la cantidad de asientos, el origen y destino del vuelo, la hora de salida, cantidad de escalas y la duración del vuelo. Defina la clase Aeropuerto que tiene nombre, localidad y la lista de aviones que salen de dicho aeropuerto. En una aplicación (Main) crear el aeropuerto “Chapelco” y cargarle varios aviones. Luego simular la venta de pasajes controlando que haya asientos disponibles. De cada venta se ingresa nro de vuelo y cantidad de asientos solicitados. Al final del proceso imprimir la cantidad de pasajes vendidos, qué porcentaje de ellos corresponde a vuelos nocturnos (horario de salida posterior a las 20h y anterior a las 6AM) y cuál es la duración promedio de los vuelos con escalas cuyo destino es Río de Janeiro.
Observación: la simulación de venta solo controla que haya asientos disponibles, no asigna nro de asiento ni los ocupa.
- 4) En un supermercado se venden distintos productos. De cada producto se conoce el código, nombre, marca, stock, nombre del proveedor y precio. Definir las clases Supermercado y Producto completas (variables de instancia, constructores, propiedades). En una aplicación crear el supermercado “Sol de Oro” y cargarle varios productos con un bucle. Luego simular la atención a clientes. Cada cliente puede comprar uno o varios productos. De cada producto se ingresa por teclado el código y la cantidad de unidades compradas. Se debe imprimir el ticket de compra y el monto final a abonar por cada cliente. Debe decrementarse el stock de cada producto comprado y en caso de que quede nulo, se debe imprimir el nombre del proveedor junto con el código del producto a reponer. Al finalizar la simulación imprimir el monto total recaudado por el supermercado.
Observación: en la simulación de atención de clientes no se hace control de stock ya que se supone que cuando el cliente llega a la caja, tiene los productos en su carro.
- 5) Implemente la clase Pila de expedientes (usando la interfaz vista en la teoría. No usar la clase Stack definida en C#). Luego resuelva el siguiente problema: En una oficina un abogado debe clasificar una pila de expedientes. Cada expediente cuenta con un número, titular, trámite y fecha de presentación. Se desea:
 - a) Generar otra pila con los expedientes correspondientes al mes de febrero.
 - b) Verificar si la señora Gabriela Turco tuvo una audiencia en el mes de marzo. Si es así, imprimir todos los trámites realizados por ella en el mes de febrero.
- 6) Implemente la clase Cola de clientes (usando la interfaz vista en la teoría. No usar la clase Queue definida en C#). Desarrolle un programa de aplicación que permita manejar colas de servicios para una compañía de seguros. Un cliente llega y según lo que necesite le corresponde un número dentro de una de las colas. El

personal de atención va llamando a cada cliente eligiendo la cola que desea atender.

Este sistema deberá permitir simular la llegada de un cliente ingresando por consola la letra C seguida de un número de servicio. El sistema deberá entregar un número de atención (el que corresponda según la cola actual del servicio).

También deberá permitir atender a un cliente ingresando por consola la letra A seguida del número de servicio. El sistema deberá mostrar en consola el número que se está llamando.