



## Programación Declarativa

Ingeniería Informática  
Especialidad de Computación  
Cuarto curso. Primer cuatrimestre.



Escuela Politécnica Superior de Córdoba  
Universidad de Córdoba

Curso académico: 2015 - 2016

---

### Práctica número 5: tipos abstractos de datos y lectura y escritura

1. Codifica las funciones de **creación, acceso, consulta y modificación** del tipo abstracto de datos **estudiante**, que posee los campos **nombre, titulación y curso**.
  - Versión con vectores
  - Versión con listas simples
  - Versión con listas de asociación
2. Tipo abstracto de datos "**lista-estudiantes**"
  - **Utiliza** las funciones del tipo abstracto **estudiante** para codificar las siguientes funciones del tipo abstracto "**lista-estudiantes**":
    - **crear-lista-estudiantes-vacía**: crea una lista vacía de estudiantes.
    - **lista-estudiantes-vacía?**: comprueba si una lista de estudiantes está vacía.
    - **buscar-estudiante!**: busca un estudiante en una lista usando su nombre.
    - **insertar-estudiante!**: inserta un estudiante en una lista.
    - **borrar-estudiante!**: borra un estudiante de una lista.
    - **modificar-estudiante!**: modifica un estudiante de una lista.
    - **primer-estudiante**: recibe una lista de estudiantes y devuelve el primer estudiante.
    - **resto-de-estudiantes**: recibe una lista de estudiantes y devuelve todos los estudiantes excepto el primero.
    - **mostrar-estudiantes**: recibe una lista de estudiantes y muestra por pantalla todos sus datos.
    - **mostrar-estudiantes-titulación**: recibe una lista de estudiantes y una titulación y muestra los estudiantes de dicha titulación.
    - **grabar-lista-estudiantes**: graba en un fichero los datos de los estudiantes.
    - **cargar-fichero-lista-estudiantes**: lee los datos de los estudiantes desde un fichero y devuelve una lista con los

estudiantes leídos.

3. Codifica un programa que permita la gestión de una “**lista de estudiantes**”

- Crear una lista de estudiantes.
- Añadir un estudiante.
- Buscar un estudiante.
- Consultar un estudiante.
- Insertar un estudiante.
- Editar un estudiante.
- Borrar un estudiante.
- Mostrar los estudiantes.
- Mostrar los estudiantes de una titulación.
- Grabar los estudiantes en un fichero.
- Cargar los estudiantes de un fichero.
- Etc.

4. Tipo abstracto de datos “**cola**”

- Este tipo abstracto se caracteriza por estar compuesto por una colección lineal de elementos, de forma que el primer elemento que entra en la cola es el primero que sale (*First in; first out:FIFO*).
- Codifica las siguientes funciones para el tipo abstracto de datos “cola”:
  - **cabeza**: muestra el elemento que está en la primera posición de la cola, pero no lo extrae de ella.
  - **extraer!**: extrae la cabeza de la cola
  - **introducir!**: coloca un elemento al final de la cola
  - **vacía?**: comprueba si la cola está vacía
- **Observación**
  - Se debe elegir la representación interna de la “cola”: vector o lista.

5. Codifica un programa que permita la **gestión de una cola**

- Introducir un elemento.
- Extraer un elemento.
- Comprobar si está vacía o no.
- Mostrar el primer elemento de la cola.
- Mostrar el contenido de toda la cola.
- Grabar el contenido de una cola en un fichero.
- Cargar el contenido de una cola desde un fichero.
- Etc.

6. Tipo abstracto de datos “**polinomio**”

$$P(X) = a_0 + a_1 X + a_2 X^2 + \dots + a_n X^n$$

- Se deben codificar las siguientes funciones del tipo abstracto
  - **crear\_polinomio**: crea un polinomio a partir de un número variable de coeficientes.

- **grado**: indica el grado del polinomio.
- **coeficiente**: indica el valor del coeficiente "i" del polinomio.
- **modificar\_coeficiente!**: modificar el coeficiente "i" de un polinomio.
- **valor\_polinomio**: calcula el valor del polinomio para un dato particular  $x_0$

$$P(x_0) = a_0 + a_1 x_0 + a_2 x_0^2 + \dots + a_n x_0^n$$

- **Observación:**
  - Se debe elegir la representación interna que se desee: vector o lista.
  - Se valorará la **eficiencia** en la representación.
    - Por ejemplo, ¿cómo se podría representar de forma eficiente el polinomio  $P(X) = X^5 - 1$ ?

## 7. Codifica un programa que permita la **gestión de un polinomio**

- Crear un polinomio.
- Escribir el coeficiente del polinomio.
- Escribir el polinomio completo.
- Modificar el coeficiente de un polinomio.
- Ver el grado de un polinomio.
- Aplicar el polinomio a un dato particular.
- Grabar un polinomio en un fichero.
- Cargar un polinomio desde un fichero.
- Etc.