



Estructuras de Datos

Grado en Informática
Segundo Curso, segundo cuatrimestre
Escuela Politécnica Superior de Córdoba
Universidad de Córdoba



Curso académico 2015-2016

Práctica 1. Polinomios de una variable

- **Objetivo**

- Usar estructuras de datos lineales para la implementación de polinomios de una variable.
- Un polinomio es una suma algebraica de monomios.
 - Ejemplo:
$$p(x) = x^3 - 2x + 1$$
- Cada monomio está compuesto por el coeficiente y el grado:
$$\text{coeficiente } x^{\text{grado}}$$

- **Primera parte: implementación de Monomio**

- Codifica una clase abstracta, denominada **MonomioInterfaz**, que especifique los métodos “virtuales puros” para acceder y modificar los atributos de un Monomio: coeficiente y grado.
- Codifica una clase, denominada **Monomio**,
 - que herede de forma pública de la clase **MonomioInterfaz**
 - que defina los atributos de la clase Monomio:
 - coeficiente, dato de tipo real.
 - grado, dato de tipo entero.
 - y que codifique los siguientes métodos de la clase Monomio
 - Constructor con valores por defecto.
 - Constructor de copia.
 - Funciones de acceso y modificación de los atributos de la clase Monomio.
 - Función para leer desde el teclado un Monomio.
 - Función para escribir por pantalla un Monomio.
 - Sobrecarga del operador “=” para la asignación de monomios.
 - Sobrecarga del operador “*” para el producto de monomios.
 - Valor de un monomio para un dato concreto de X.
 - Además, también se deben codificar las funciones **amigas** para sobrecargar los operadores “>>” y “<<”
 - **friend** istream &operator>>(istream &stream, Monomio &m);
 - **friend** ostream &operator<<(ostream &stream, Monomio const &m);

- **Segunda parte: programa de prueba de la clase Monomio**

- Se debe codificar un programa para comprobar el funcionamiento de la clase **Monomio**.
- El programa debe
 - declarar tres monomios.

- Se debe mostrar el uso del constructor con valores por defecto y del constructor de copia.
 - y permitir el uso de un menú con, al menos, las siguientes operaciones:
 1. Valor de un monomio para un dato concreto de X.
 2. Multiplicación de dos monomios y almacenar el resultado en el tercero.
 3. Fin del programa
- **Tercera parte: implementación de Polinomio**
 - Codifica una clase abstracta, denominada **PolinomioInterfaz**, que especifique los métodos “virtuales puros” para
 - acceder y modificar los atributos de un Polinomio: grado, número de monomios y monomios
 - y para comprobar si un Polinomio es nulo o vacío.
 - Codifica una clase, denominada **Polinomio**,
 - que herede de forma pública de la clase **PolinomioInterfaz**
 - que defina los atributos de la clase **Polinomio**
 - grado, de tipo entero.
 - número de monomios o términos, de tipo entero.
 - monomios: se utilizará un vector o una lista de la biblioteca de plantillas STL de C++.
 - y que codifique los siguientes métodos de la clase Polinomio
 - Constructor con valores por defecto.
 - Constructor de copia.
 - Funciones de acceso y modificación de los atributos de la clase Polinomio.
 - Función para leer desde el teclado un Polinomio.
 - Función para escribir por pantalla un Polinomio.
 - Sobrecarga del operador “=” para la asignación de polinomios.
 - Sobrecarga del operador “+” para la suma de polinomios.
 - Sobrecarga del operador “*” para el producto de polinomios.
 - Valor de un polinomio para un dato concreto de X.
 - Además, también se deben codificar las funciones **amigas** para sobrecargar los operadores “>>” y “<<”
 - **friend** istream &operator>>(istream &stream, **Polinomio** &p);
 - **friend** ostream &operator<<(ostream &stream, **Polinomio** const &p);
- **Cuarta parte: programa de prueba de la clase Polinomio**
 - Se debe codificar un programa para comprobar el funcionamiento de la clase **Polinomio**
 - El programa debe
 - declarar tres polinomios.
 - Se debe mostrar el uso del constructor con valores por defecto y del constructor de copia.
 - y permitir el uso de un menú que con, al menos, las siguientes operaciones:
 1. Valor de un polinomio para un dato concreto de X.
 2. Suma de dos monomios y almacenar el resultado en el tercero.
 3. Multiplicación de dos monomios y almacenar el resultado en el tercero.
 4. Lectura y escritura de un monomio haciendo uso de los los operadores “>>” y “<<”

5. Fin del programa

- **Observaciones**

- Se debe usar el espacio de nombres de la asignatura: **ed**
- Se debe documentar el código con **doxygen**
- Se debe crear un fichero **makefile**
- **Opcional**
 - Clase Monomio
 - Sobrecarga del operador “+” para la suma de monomios con el mismo grado: el resultado es otro monomio.
 - Clase Polinomio
 - Sobrecarga del operador “=” para la asignación de un monomio a un polinomio.
 - Sobrecarga del operador “+” para la suma de un polinomio y un monomio: el resultado es un polinomio.
- **Tiempo de realización:**
 - Comienzo de la práctica: semana del 22 al 26 de febrero.
 - Entrega de la práctica:
 - Grupos 1 y 2: hasta las 9:00 horas del lunes 14 de marzo de 2016.
 - Grupos 3, 5 y 6: hasta las 9:00 horas del martes 8 de marzo de 2016.
 - Grupo 4: hasta las 9:00 horas del miércoles 9 de marzo de 2016.