



Pontificia Universidad Javeriana

Departamento de Ingeniería de Sistemas

Estructuras de Datos

Taller 2: Estructuras lineales, 2024-10

1 Objetivo

Completar la implementación de una aplicación de manipulación de polinomios. En especial, se desea evaluar las habilidades del estudiante en el desarrollo y uso de las operaciones de recorrido en secuencias.

2 Recordatorio: compilación con g++

La compilación con g++ (compilador estándar que será usado en este curso para evaluar y calificar las entregas) se realiza con los siguientes pasos:

1. **Compilación:** de todo el código fuente compilable (**ÚNICAMENTE LOS ARCHIVOS CON EXTENSIONES** *.c, *.cpp, *.cxx)
`g++ -std=c++11 -c *.c *.cxx *.cpp`

2. **Encadenamiento:** de todo el código de bajo nivel en el archivo ejecutable
`g++ -std=c++11 -o nombre_de_mi_programa *.o`

Nota: Estos dos pasos (compilación y encadenamiento) pueden abreviarse en un sólo comando:

`g++ -std=c++11 -o nombre_de_mi_programa *.c *.cxx *.cpp`

3. **Ejecución:** del programa ejecutable anteriormente generado
`./nombre_de_mi_programa`

ATENCIÓN: Los archivos de encabezados (*.h, *.hpp, *.hxx) **NO SE COMPILAN**, se incluyen en otros archivos (encabezados o código). Así mismo, los archivos de código fuente (*.c, *.cpp, *.cxx) **NO SE INCLUYEN**, se compilan. Si el programa entregado como respuesta a este Taller no atiende estas recomendaciones, automáticamente se calificará la entrega sobre un 25% menos de la calificación máxima.

3 Desarrollo del taller

El desarrollo del taller consistirá en completar el código de una aplicación de manipulación de polinomios aritméticos, cuyos archivos fuente se entregan en un comprimido adjunto. En particular, se debe completar la implementación de los métodos contenidos en el archivo "Polinomio.hxx" y, adicionalmente, generar un reporte (en formato PDF) donde se respondan algunas preguntas de reflexión.

Un polinomio es una expresión que se puede representar como:

$$P_A(x) = \sum_{j=0}^{n-1} a_j x^j$$

donde el conjunto $A = \{a_j \in \mathbb{R}\}$ es una secuencia de coeficientes que caracterizan el polinomio. El exponente más alto asociado a un coeficiente no nulo se conoce como el grado del polinomio $\text{grado}(A)$ (en el caso del ejemplo es $n - 1$).

1. **(10%)** En el reporte, describa paso a paso el procedimiento principal (función "main") en el archivo main.cxx. Sea conciso y describa cada paso en 15 palabras o menos.

Estudie los archivos "Polinomio.h" y "Polinomio.hxx". ¿Es claro y se entiende todo el código que se presenta allí? si hay algo que no conoce o no entiende, intente hacer una búsqueda en internet, o no dude en preguntarle al profesor o al monitor.

2. **(5%)** Intente compilar el código. En el reporte, haga un inventario conciso de los errores y advertencias que aparecen (no copiar y pegar tal cual), y proponga formas de solucionarlos. Aplique estas soluciones en el código fuente. Después de esto, el código debería compilar sin errores ni advertencias.

Para que pueda ejecutar el programa compilado, se provee un ejemplo de archivo de entrada (polinomios_00.in) y su correspondiente salida (polinomios_00.out).

3. **(10%)** En el reporte, describa el método "Polinomio<S>::FijarCoeficiente()". Sea conciso en su descripción (15 palabras o menos).

Identifique en el archivo "Polinomio.hxx" las secciones marcadas con el comentario `/** TODO #n */`, éstas son las secciones donde se debe completar el código del algoritmo, como se indica a continuación:

4. **(15%)** TODO #1: escriba el código necesario para calcular la suma de dos polinomios, teniendo en cuenta las siguientes definiciones:

$$P_C(x) = P_A(x) + P_B(x)$$

donde

$$C = \{a_j + b_j; a_j, b_j \in \mathbb{R} \wedge j \in [0, \max(\text{grado}(A), \text{grado}(B))]\}$$

En el reporte, describa un plan de pruebas para esta operación (suma de polinomios).

5. **(35%)** TODO #2: escriba el código necesario para calcular el producto de dos polinomios, teniendo en cuenta las siguientes definiciones:

$$P_C(x) = P_A(x) P_B(x)$$

donde

$$C = \left\{ \sum_{k=0}^j a_k b_{j-k}; a_j, b_j \in \mathbb{R} \wedge j \in [0, 2n-2] \right\}$$

y

$$n = \max(\text{grado}(A), \text{grado}(B))$$

En el reporte, describa un plan de pruebas para esta operación (producto de polinomios).

6. **(25%)** TODO #3: escriba el código necesario para evaluar el polinomio, dado un valor de x .

4 Evaluación

La entrega se hará a través de la correspondiente asignación de BrightSpace, antes de la medianoche del martes 20 de febrero. Se debe entregar un único archivo comprimido (único formato aceptado: .zip), nombrado con los apellidos de los integrantes del grupo. Este archivo comprimido debe contener, dentro de un mismo directorio (sin estructura de carpetas interna), el documento del reporte (.pdf) y el código fuente modificado (.h, .hxx, .cxx, .cpp). Si la entrega contiene archivos en cualquier otro formato, será descartada y no será evaluada, es decir, la nota definitiva de la entrega será de 0 (cero) sobre 5 (cinco).

La evaluación del taller tendrá la siguiente escala para cada una de las preguntas o secciones de código a completar:

- **Excelente (5.0/5.0):** El código es correcto, compila sin advertencias y/o la respuesta es correcta y concisa.
- **Bueno (3.8/5.0):** El código es correcto, pero compila con advertencias y/o la respuesta es correcta pero no es concisa.
- **No fue un trabajo formal de ingeniería (3.0/5.0):** El código es parcialmente correcto y/o compila con advertencias y/o la respuesta es aproximada y/o no es concisa.
- **Necesita mejoras sustanciales (2.0/5.0):** El código no es correcto y/o la respuesta no es clara.
- **Malo (1.0/5.0):** El código entregado por el estudiante no compila en el compilador g++ (mínimo versión número 4.5).
- **No entregó (0.0/5.0).**