

Tarea 3: Programación Intermedia y OOP (7%)

Marlon Brenes*

El propósito de esta tarea es practicar los conceptos de OOP y C++. La tarea involucra entregar código fuente de C++ con su respectivo reporte detallando el análisis. La tarea se puede realizar en parejas, pero **cada estudiante debe entregar sus propios archivos**. Usted debe entregar los siguientes archivos para ser evaluados:

- vector.cpp, vector.hpp y main.cpp
- rk4.cpp
- reporte3.pdf/reporte3.doc/reporte3.docx/reporte3.ipynb (dependiendo de cuál formato escoja para su reporte)

PARTE I

Su tarea es construir una clase que puede ser utilizada para trabajar con vectores. La clase maneja un recurso de memoria dinámico asignado en el heap, de manera tal que los elementos del vector son contiguos en memoria.

1. Genere tres archivos:

- vector.hpp: este archivo contiene la declaración de la clase
- vector.cpp: este archivo contiene la implementación de la clase
- vector.cpp: este archivo contiene el main() con el cual pondremos en uso la funcionalidad de la clase

2. Declare una clase llamada `Vector` que contiene dos atributos: `int n_elems` para el número de elementos y `double *data` que contiene los elementos del vector. Decida si los miembros deben ser públicos o privados. Si los miembros son privados, debe implementar métodos para acceder estos miembros sin ser modificados [0.5%]

3. Decida que hacer con el constructor por defecto y escriba su implementación para no generar un estado inconsistente de los objetos de tipo `Vector` [0.5%]

4. Genere un constructor personalizado que asigna la memoria del vector con el número de elementos como argumento [0.5%]

5. Genere un constructor copia apropiado (deep copy) [0.5%]

6. Genere un operador de asignación apropiado (dos vectores son iguales si y solo si sus elementos y dimensiones son iguales) [0.5%]

7. Genere un destructor apropiado [0.5%]

8. Escriba un método para imprimir el vector en stdout [0.5%]

9. Sobrecarga de operadores: Sobrecargue los siguientes operadores y explique porqué en general esto es una buena práctica en su reporte [1.0%]:

- Adición +
- Sustracción -
- Multiplicación *
- Ostream << (se sobrecarga este operador para imprimir en terminal objetos de tipo `Vector`)

10. En todos los métodos que lo requieran, utilice manejo de excepciones para cerciorar que `n_elems` satisface las condiciones necesarias para la operación requerida [0.5%]

11. Escriba una prueba de cada método y operador en el main. Las operaciones deben dar resultados correctos. La inicialización (i.e., los valores que el vector contiene) puede hacerse en el main. [1.0%]

* marlon.brenes@utoronto.ca

PARTE II

Utilice el método RK4 para evaluar la solución a la ecuación

$$\frac{dy}{dx} = 2(1 - y) - e^{-4x}, \quad (1)$$

con la condición inicial $y(0) = 1$. Note que el lado derecho de la ecuación depende de la variable independiente. Escriba la solución en C++ y su reporte en el formato que guste. Para graficar el resultado, debe escribir la solución en un archivo que contiene en una columna el valor de x y en la otra columna el valor de y . Con este archivo, puede utilizar su graficador preferido para realizar el gráfico (e.g., gnuplot, matplotlib, etc). Grafique el resultado de la integración numérica en el rango $x \in [0, 2]$ utilizando distintos valores de h [4.0%].