



## TAREA 6. PSEUDOCÓDIGO PARA RESOLVER SERIES DE TAYLOR EN C.

Adolfo Hernández Ramírez (427560)

Correo: [a.hernandezramirez3@ugto.mx](mailto:a.hernandezramirez3@ugto.mx).

Licenciatura Ingeniera Química Sustentable. Universidad de Guanajuato. División de Ciencias e Ingenierías. Campus León. Loma del Bosque 103, Lomas del Campestre. León, Gto, México.

Pseudocódigo.

1. Definir las siguientes funciones en c.

- Sen(x)
- Cos(x)
- $\text{Exp}^{-x}$
- $\text{Exp}^{2x}$ .
- Arctan(x).

2. Definir como funciones las 10 derivadas de cada función.

3. Definir una función factorial para almacenar el valor de las factoriales para la serie de Taylor. Se usará un ciclo for donde dentro del ciclo este la instrucción `for(int = 1, i<=n, i++)` y dentro del ciclo la operación `f*=i`.

4. Dentro del int main, declarar la variable entera repetir, que será necesaria para el do-while.

5. Dentro del do, declarar las variables enteras opcion y n\_lim, y las variables doubles: x, x\_i, h.

6. Imprimir en la pantalla las 5 funciones que contiene el programa. Y especificar un número para cada función.

7. Registrar la respuesta del usuario con un scanf.

8. Pedir y registrar los valores de x y x\_i con scanf.

9. Declarar que  $h=x-x_i$ .

10. Pedir y registrar el valor de n\_lim que el usuario desee con scanf.

11. Colocar un condicional `if(n_lim>=10)` imprimir que no existe la 11 derivada y que los términos de la serie serán hasta  $n=9$ .

12. Declarar un arreglo que contenga 11 elementos y que reciba para cada uno la derivada n-esima de la función según la función.

13. Declarar un arreglo llamado funcion que apuntara a una de las 5 funciones según sea el caso con el switch.

14. Establecer un switch(opcion) donde en cada caso se definiran en los elementos del arreglo cada derivada para la función.

$df[j] = fm\_dj.$

funcion = fm.

Donde j es el elemento del arreglo desde 0 hasta 10 y que cada uno corresponde a la derivada de la función. m es el número de la función.

15. Colocar en el default un printf donde se imprima que la opción que colocó el usuario es inválida.

16. Declarar una variable double fx que recibirá el valor del arreglo función.

17. Declarar una variable double llamada termino\_n = 0, que corresponde al valor de la f(x) usando la serie de Taylor.

18. Dentro de un ciclo `for(n = 0; n < 10; n++)`, calcular la suma de los terminos de la serie de Taylor de acuerdo al siguiente procedimiento: Declaramos una nueva variable double

$\text{termino\_i} = df[n](x\_i) * \text{pow}(h, n) / \text{factorial}.$

$\text{termino\_n} += \text{termino\_i}.$

19. Tambien dentro del ciclo for, calculamos el residuo declarando la variable como una variable double.

$\text{termino\_i} = df[n+1](x\_e) * \text{pow}(h, n+1) / \text{factorial}(n+1)$

Donde  $x_e$  es el valor medio entre  $x$  y  $x_i$ .

20. Dentro del ciclo for, declaramos la variable double de error y la calculamos como:

Error = fabs(fx-termino\_n).

21. Imprimir una tabla de resultados que contiene: n, fx, termino\_n, residuo y error.

22. Colocar un condicional donde si el error es menor que  $1e-6$ , se coloque un break y el ciclo for acabe hasta donde se cumpla la condición.

23. Preguntar al usuario si desea hacer otra ejecución del programa donde 1=si y 0=no, y registrar su respuesta con un scanf.

24. Colocar la condicional del do-while, while(repetir==1)

25. Finalizar el programa si el usuario colocó un valor igual a 0 o diferente de 1.