Tarea 2 - Problema 1 - Programación avanzada

Nicolás Aylwin

El primer desafío resulta en encontrar el valor de las funciones de Bessel de primer orden ($J_n(x)$). Sabiendo que estas cumplen la recurrencia:

$$J_{n+1}(x) = \frac{2n}{x} J_n(x) - J_{n-1}(x)$$

Pero que esta recurrencia diverge de los valores reales para n creciente, se utiliza el algoritmo de Miller, en la que se utiliza la recurrencia decreciente. Matemáticamente:

$$J_n(x) = \frac{2(n+1)}{x} J_{n+1}(x) - J_{n+2}(x)$$

Se programó un ciclo 'for' que realizara esta recurrencia para un n cualquiera, comenzando desde $N \approx n + \sqrt{10n} + 1$ (solo busqué más precisión de la sugerida). De obtener un N menor a 20, se ajustó este a 20.

Durante el ciclo, se acumula la suma de $2J_i$ con i par, y J_0. También se almacena en una variable el J_n con el n pedido en la llamada de la función.

Luego de terminar el ciclo, se divide la suma mencionada al valor de J_n almacenado, y se devuelve este último como valor de $J_n(x)$ para x y n dado. Esta normalización viene de la condición que cumplen las funciones de Bessel:

$$1 = J_0(x) + \sum_{i=1}^{\infty} 2J_{2i}(x)$$

No son necesarias 'tantas' funciones de Bessel para lograr un valor muy cercano pues a mayor n en J_n, la función evaluada se acerca a cero rápidamente. Por esto se logra una buena aproximación con pocos valores.

Luego se busca graficar $f(x) = J_0(x) + 2\sum_{i=1}(-1)^i J_{2i}(x)$ y $g(x) = 2\sum_{i=0}(-1)^i J_{2i+1}(x)$. Para esto se crea una función que entregue un vector conteniendo todas las $J_n(x)$ con $0 \leq n \leq N$ para un x y N dado. Se crea también una función que separa de este último vector, los J_n pares e impares, los almacena en vectores separados para así entregarlos a f y g respectivamente.

Estas últimas se programaron para recibir los vectores mencionados y retornar un valor, correspondiente a la función evaluada en algun x.

Usando todo esto, se establece un ciclo 'for' en el 'main' que evalua en 100 puntos entre 0 y 2π el valor de f y g, guardando en cada iteración el valor de x y su evaluación en 'grafico_f.dat' y 'grafico_g.dat' respectivamente. El programa se ejecuta, hace todo, y luego de crear estos dos archivos, lanza un mensaje a la terminal indicando el éxito del programa.

Luego con estos archivos se crean gráficos a través de Gnuplot, almacenados en esta misma carpeta. En particular, $f(x)$ se plotea junto a $\cos(x)$, y $g(x)$ con $\sin(x)$, mostrándose así que al menos en este rango son la misma función.