

Programación y Métodos Numéricos Avanzado

Tarea 4

Entrega: 09 Mayo hasta 23:59 hrs.
29 de abril de 2019

Profesor: Benjamín Toledo
Ayudante: Víctor Gaete

- Comprima sus respuestas en un único archivo con el formato `nombre_apellido_tareaN.tar.bz2`
- En el archivo comprimido debe existir una carpeta por cada problema de su tarea con el nombre `problemaN`, donde almacenará los programas necesarios para sus respuestas.
- Envíe el archivo `tar.bz2` al correo `victor.gaete@ug.uchile.cl`. Cualquier tarea enviada fuera de plazo y sin el formato pedido, no será revisada.
- Para cada problema, usted debe generar un único texto en \LaTeX , en el que explique de manera **clara** y **breve** el método utilizado para desarrollar su programa, modo de uso y los resultados del mismo (output de programas, gráficos, ecuaciones, tablas, etc.) si es necesario.
- Se sugiere comentar partes de su código explicando de forma **breve** lo que realiza. Evite incluir código innecesario, es decir, que no se utiliza al correr su programa. Se sugiere ser ordenado, recuerde que otra persona debe entender lo que usted quiso hacer.
- En caso de recibir códigos copiados o tareas **similares** se calificará con la nota mínima.

Problema 1

Los polinomios de Legendre, $P_l(x)$, pueden ser encontrados de varias maneras. Una de ellas es mediante el proceso de ortogonalización de Gram-Schmidt. Considere,

$$\begin{aligned}P_0(x) &= 1, \\P_1(x) &= x, \\ \|P_l\|^2 &= \frac{2}{2l+1}, \\ \langle f(x), g(x) \rangle &= \int_{-1}^1 f(x)g(x)dx, \end{aligned} \tag{1}$$

donde (1) corresponde al producto interno entre dos funciones de x que pertenecen a L^2 . Utilizando estas definiciones cree un algoritmo que permita encontrar $P_l(x)$, para ello **debe** crear una clase de template de polinomios con grado entero, determine sus resultados implementando sus polinomios sobre variables tipo `double`, y estime cuanto difiere su valor para cada $0 \leq l \leq 20$, con respecto a las funciones conocidas (tabuladas).

Problema 2

Un ejemplo de autómatas celulares es ‘El Juego de la Vida’, que se caracteriza por ser un juego de 0 jugadores altamente dependiente de sus condiciones iniciales.

El juego consta de una grilla (en este caso de 32×32) cuyas casillas pueden tomar los valores 0 (muerto) ó 1 (vivo). Para nacer una casilla debe estar rodeada de cierta cantidad de células vivas (uno o varios valores enteros entre 0 y 8), y para mantenerse con vida también debe estar rodeada de cierta cantidad de células vivas.

Genere un programa capaz de leer un archivo llamado `input.dat` el cual contiene las condiciones iniciales para su partida y una función que avance una generación su grilla la cual debe aceptar al menos dos valores enteros para mantenerse con vida y dos valores enteros para que nazca. Cree un archivo para cada generación n de su grilla con $0 \leq n \leq 700$, y haga un vídeo de 2 minutos de duración con la evolución de su sistema. Considere que una célula se mantiene con vida si tiene 2 ó 3 vecinos vivos, y nace si tiene 3 ó 6 vecinos vivos, además su grilla debe ser periódica.