



TAREA 7

Fecha de entrega: 21/11/2020 23:59 hrs

Problema 2

Integre la ecuación de Poisson para el potencial electrostático:

$$\nabla^2 V(x, y) = -\rho(x, y)$$

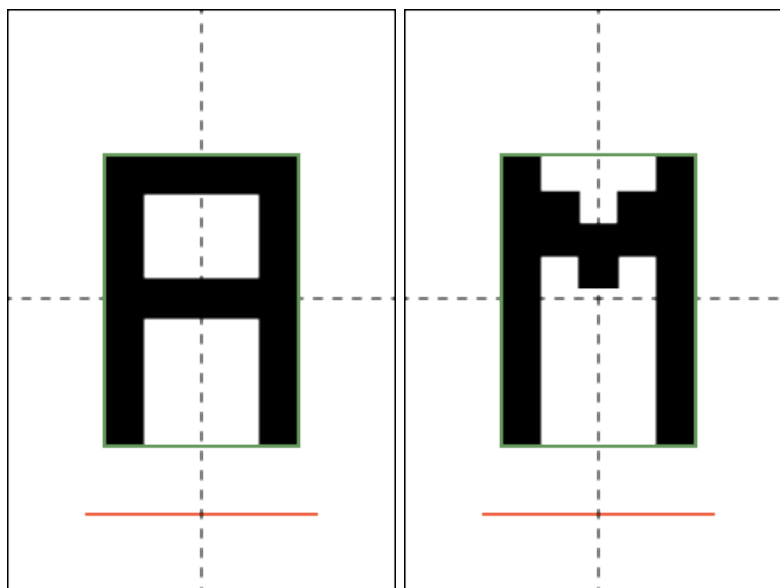
donde $\rho(x, y)$ es la densidad de carga. Integre dentro de una caja rectangular de dimensiones 10 [cm] \times 15 [cm] conectada a tierra (es decir, $V = 0$ en el perímetro). Definiremos el centro de la caja como $(x, y) = (0, 0)$. Dentro de la caja hay una línea que une los puntos $(x_0, y_0) = (-3, -5.5)$ y $(x_f, y_f) = (3, -5.5)$ que cumple con condiciones de borde derivativas:

$$\frac{\partial V}{\partial y} = -1 \text{ (sobre la línea)}$$

Además, en la caja hay una letra, la primera letra de su nombre. Dicha letra está contenida dentro del rectángulo centrado con lados 5 [cm] \times 7 [cm] en las direcciones x e y , respectivamente. Dibuje la letra como quiera, pero intente hacerla lo más simple posible y en particular evite líneas curvas y diagonales. El grosor de las líneas que dibujan su letra debe ser de 1 [cm]. La carga total dentro de la letra es:

$$Q = \int \rho(x, y) dx dy = 2.XXX [C]$$

(la unidad es el Coulomb), XXX son los 3 últimos dígitos de su RUT y la densidad de carga es constante dentro de la letra. A continuación un par de ejemplos de letras complicadas (sólo como referencia, Ud. puede dibujarlas como mejor le parezca). La línea roja es donde se aplica la condición de borde derivativa. El rectángulo verde es el que contiene a la letra. Las líneas punteadas marcan el centro de la caja.



- Use un reticulado con $h \sim 0.1$ [cm].
- Note que tendrá que derivar el algoritmo de iteración para los puntos adyacentes al segmento con condición de borde derivativa y para el segmento mismo. Debido a esto, es recomendable separar la iteración en distintos segmentos
- Use el método de sobre-relajación sucesiva con distintos w y estudie cuantas iteraciones hacen falta para converger en cada caso.
- Debe definir un criterio de convergencia, explícelo en el informe.
- Se recomienda que parta con un setup simple y vaya agregando complejidad, asegurándose primero de que los casos simples funcionan. Vaya dejando el rastro de su trabajo en `git`.
- Asegúrese de incluir gráficos que demuestren de forma efectiva la solución obtenida. Éstos pueden incluir gráficos de superficie en 2D y 3D, líneas de contorno, cortes transversales, etc. Note que el efecto de la letra es muy tenue (la carga es baja) por lo que debe probar alternativas para la escala del plot.

Instrucciones Importantes.

- Evaluaremos su uso correcto de `python`. Si define una función relativamente larga o con muchos parámetros, recuerde escribir el *docstring* que describa los parámetros que recibe la función, el output, y el detalle de qué es lo que hace la función. Recuerde que generalmente es mejor usar varias funciones cortas (que hagan una sola cosa bien) que una muy larga (que lo haga todo). Utilice nombres explicativos tanto para las funciones como para las variables de su código. El mejor nombre es aquel que permite entender qué hace la función sin tener que leer su implementación ni su *docstring*.
- Su código debe aprobar la guía sintáctica de estilo ([PEP8](#)). En [esta página](#) puede chequear si su código aprueba PEP8.
- Utilice `git` durante el desarrollo de la tarea para mantener un historial de los cambios realizados. La siguiente [cheat sheet](#) le puede ser útil. **Revisaremos el uso apropiado de la herramienta y asignaremos una fracción del puntaje a este ítem.** Realice cambios pequeños y guarde su progreso (a través de *commits*) regularmente. No guarde código que no corre o compila (si lo hace por algún motivo deje un mensaje claro que lo indique). Escriba mensajes claros que permitan hacerse una idea de lo que se agregó y/o cambió de un `commit` al siguiente.
- Para hacer un informe completo Ud. debe decidir qué es interesante y agregar las figuras correspondientes. No olvide anotar los ejes e incluir una *caption* que describa el contenido de cada figura. Tampoco olvide las unidades asociadas a las cantidades mostradas en los diferentes gráficos.
- La tarea se entrega subiendo su trabajo a github. Clone este repositorio (el que está en su propia cuenta privada), trabaje en el código y en el informe y cuando haya terminado asegúrese de hacer un último `commit` y luego un `push` para subir todo su trabajo a github.

- El informe debe ser entregado en formato pdf, este debe ser claro sin información de más ni de menos. **Esto es muy importante, no escriba de más, esto no mejorará su nota sino que al contrario.** La presente tarea probablemente no requiere informes de más de 3 o 4 páginas en total (dependiendo de cuántas figuras incluya; esto no es una regla estricta, sólo una referencia útil). Asegúrese de utilizar figuras efectivas y tablas para resumir sus resultados. **Revise su ortografía.**
- Repartición de puntaje: 50 % implementación y resolución del problema (independiente de la calidad de su código); 35 % calidad del reporte entregado: demuestra comprensión del problema y su solución, claridad del lenguaje, calidad de las figuras utilizadas; 5 % aprueba o no PEP8; 10 % diseño del código: modularidad, uso efectivo de nombres de variables y funciones, docstrings, uso de git, etc.