

Taller #2 de Métodos Computacionales

FISI 2028, Semestre 2014 - 20

Profesor: Jaime Forero

Viernes 14 de Agosto, 2014

Importante

- Los tres archivos de código fuente de solución de esta tarea deben subirse a través de sicua-plus antes de las medio día del viernes 29 de Agosto como un único archivo zip con el nombre `NombreApellidos_hw2.zip`, por ejemplo yo debería subir un archivo llamado `JaimeForero_hw2.zip`
- La nota máxima de este taller es de 100 puntos. Se otorgan 1/3 de los puntos si el código fuente es razonable, 1/3 si se puede compilar/ejecutar y 1/3 si da los resultados correctos.
- **Para las personas que entreguen la tarea antes del medio día del viernes 22 de Agosto cada programa se calificará sobre 35 puntos en lugar de 30. Es decir, pueden sacar 115 puntos en lugar de 100.**

1. Jugando con π

En el Repositorio `ComputationalMethodsData` se encuentra el archivo `homework/hw_1/Pi_2500000.txt` que contiene cerca de 2.5 millones de dígitos del número π en su representación decimal. Escriba un programa en C (el código fuente se debe llamar `suma_digitos_pi.c`) que sume los primeros n dígitos de la representación decimal de π . Después de compilado el programa debe poder ejecutarse como

```
./a.out n
```

Donde n representa el número de dígitos decimales para sumar.

- (5 puntos) El programa debe parar su ejecución y dar un mensaje de error si $n < 0$ o si $n > 2500000$.
- (30 puntos) El programa debe funcionar de manera correcta con valores hasta de $n = 2500000$. El valor final se debe imprimir en pantalla.

2. Volando de una ciudad a otra

Los aviones que vuelan largas distancias internacionales buscan minimizar su consumo de combustible. Para esto la ruta de vuelo debe ser la más corta entre los puntos de partida y llegada. El objetivo de este ejercicio es escribir un programa que calcule las coordenadas de esa trayectoria para dos puntos arbitrarios sobre la Tierra.

Para simplificar el problema supondremos que la Tierra es perfectamente esférica. Además vamos a utilizar los ángulos θ (polar) y φ (ecuatorial) para describir las coordenadas sobre un punto de la esfera.

Escriba un programa (código fuente `trayectoria_esfera.c`) que dados dos puntos arbitrarios sobre la esfera (parametrizados por θ y φ) calcula la trayectoria más corta que entre esos dos puntos. Después de compilado el programa debe ejecutarse como

```
./a.out theta_1 phi_1 theta_2 phi_2
```

Donde `theta_1`, `phi_1` corresponden a las coordenadas del primer punto y `theta_2`, `phi_2` a las del segundo. Ambos valores deben darse en grados.

- (5 puntos) El programa debe parar su ejecución y dar un mensaje de error si los ángulos se encuentran por fuera del rango permitido.

- (30 puntos) El programa debe escribir en pantalla una serie de 100 pares de valores de θ y φ que marquen la trayectoria. Estos valores deben corresponder a puntos equidistantes sobre la esfera.

3. Marcha Aleatoria 3D

En clase hemos ganado experiencia haciendo marchas aleatorias. En este punto vamos a obtener estadísticas para una marcha aleatoria en tres dimensiones.

Las condiciones y definiciones son las siguientes:

- La marcha se hace en coordenadas cartesianas y empieza en el origen $x_0 = 0, y_0 = 0, z_0 = 0$.
- Cada paso que toma la marcha siempre tiene longitud igual a uno, $r = \sqrt{d_{ix}^2 + d_{iy}^2 + d_{iz}^2} \equiv 1$ donde los d_i indican los desplazamientos del paso i en cada una de las coordenadas.
- La dirección en que se hace el desplazamiento es aleatoria y corresponde a una distribución plana de probabilidad sobre la esfera. Es decir, no puede haber ninguna dirección privilegiada.
- Luego de N pasos la marcha se encuentra en la posición x_N, y_N, z_N a una distancia $r_N = \sqrt{x_N^2 + y_N^2 + z_N^2}$ del centro.

Esta marcha es difusiva, es decir, que luego de un N_D suficientemente grande puedo esperar que la marcha se encuentre alejada del origen una distancia arbitraria D . El objetivo de este ejercicio es estudiar la relación estadística entre N_D y D .

Escriba un programa (código fuente `marcha_3D.c`) que calcule M marchas aleatorias. El programa debe calcular el valor medio y la dispersión de los M valores de N_D , cuando un D determinado se fija.

En este caso nos interesa hacerlo para una serie de 50 valores diferentes $D = 10, 20, \dots, 490, 500$.

Después de compilado el programa se debe ejecutar como

```
./a.out M
```

donde $M > 2$ es el número total de marchas aleatorias.

- (30 puntos) El programa debe funcionar de manera correcta para valores arbitrarios de M . Al final de su ejecución el programa debe crear un archivo de nombre `final_stats_3D_M.dat` (donde M corresponde al número de marchas) que contiene dos columnas y 50 filas. La primera columna tiene los valores de $D = 10, 20, \dots, 490, 500$ y la segunda tiene el valor medio $\langle N_D \rangle$ para el D correspondiente.
- Escriba en los comentarios del código fuente si logró encontrar alguna relación entre D y $\langle N_D \rangle$ con los datos obtenidos por su programa.

4. Three Wise Men

The story of the three wise men got me wondering: What if you did walk towards a star at a fixed speed? What path would you trace on the Earth? Does it converge to a fixed cycle? —N. Murdoch

Answer: <https://what-if.xkcd.com/25/>