

Análisis y Diseño de Algoritmos (620448)
Tarea # 2 *
(Máximo dos integrantes por grupo).

Profesor: Pedro A. Rodríguez M.

December 2, 2018

1 Enunciado.

Dado un conjunto de puntos en el espacio bidimensional, con coordenadas (x,y), se pide encontrar el par de puntos más cercano. Para lograr este objetivo, usted deberá implementar dos algoritmos:

1. Un algoritmo de fuerza bruta (visto en clases).
2. Un algoritmo diseñado con la técnica de dividir y conquistar.

En conjuntos de puntos muy grande, es posible que haya más de un par de puntos con la misma distancia, por lo tanto si el conjunto tiene más de un par de puntos con la misma distancia más corta, entonces debe encontrarlos.

2 Se pide que usted:

1. Programe estos algoritmo en lenguaje C.
2. Implemente un solo programa en lenguaje C, el cual permita al usuario elegir el algoritmo.
3. Los datos de entrada del programa deberán ser proporcionados por consola, como por ejemplo, la elección del algoritmo y el nombre del archivo que contiene los puntos.
4. El conjunto de puntos debe ser leído desde un archivo.
5. Use varios archivos fuente para escribir su programa. Por ejemplo, cada algoritmo de búsqueda del par de puntos debe ir en archivos fuente separados.
6. Use el comando **make** para compilar el programa completo (los comandos para la compilación deben estar escritos en un archivo Makefile).
7. Use un depurador (tal como **gdb**) para encontrar errores en tiempo de ejecución.
8. Use archivos cabeceras (headers) creados por usted para definir, por ejemplo, la estructura de los puntos.
9. El programa debe imprimir finalmente el o los pares de puntos más cercanos y su distancia.
10. Tomar el tiempo de ejecución de los algoritmos. Use la función `time(time_t *tloc)` para este propósito. Deberá ejecutar los algoritmos con arreglos de puntos de tamaño: 1.000.000, y 5.000.000.

*Máximo dos integrantes por grupo

11. Debe proporcionar un informe que tenga la siguiente estructura:
- (a) Portada (título, asignatura, nombre integrantes, nombre profesor, nombre de los alumnos ayudantes.)
 - (b) Los algoritmos en pseudo lenguaje.
 - (c) Haga el análisis temporal para cada algoritmo en el peor caso, e indique el orden de crecimiento del tiempo de ejecución. Muestre que el algoritmo ingenuo o por fuerza bruta es $O(n^2)$ y obtenga un algoritmo diseñado con la técnica de dividir y conquistar que tenga complejidad $O(n \log n)$.
 - (d) Tabule los tiempos de ejecución de los algoritmos. Construya una sola tabla para los dos algoritmos, con los tiempos medidos durante la ejecución.
 - (e) Grafique estos tiempos de ejecución. Construya un solo gráfico número de puntos versus tiempo de ejecución. En el mismo gráfico deben ir las curvas de tiempo de los dos algoritmos.
 - (f) Conclusiones.
12. Programe en ambiente Linux (use herramientas como **gcc**, **make**, **gdb**, etc).

A ustedes se les entregará dos archivos de texto con las listas de puntos a procesar. Este archivo contiene en la primera línea el número de puntos y a continuación la lista de puntos, de la forma: **x y**.

3 Plazo de entrega de la Tarea.

Para el día día viernes 14 de diciembre de 2018 hasta las 23:59 hrs. La tarea debe ser enviada a mi correo electrónico prodrigu@ubiobio.cl, con copia a los alumnos ayudantes: Xavier Canales (xavier.canales1501@alumnos.ubiobio.cl) y Carlos Faundes (carlos.faundez1501@alumnos.ubiobio.cl). El correo debe ir con el nombre de los alumnos. El subject o asunto del correo debe decir "Entrega Tarea 2 de ADA".