

## Problema Práctico 2. Algoritmos de grafos

Título	Implementación de algoritmos de grafos
<b>Aprendizaje esperado (objetivo)</b>	El alumno aprenderá a utilizar dos de las bibliotecas existentes para la programación de grafos y la utilización de los algoritmos de recorridos en profundidad y en amplitud, obtención de un árbol de expansión mínimo mediante Kruskal y Prim, así como también los algoritmos de caminos mínimos Dijkstra y Floyd-Warshall.
<b>Instrucciones</b>	<p>Busque en Internet las bibliotecas para la implementación de grafos (<a href="#">Boost Graph Library</a> y <a href="#">SNAP</a>) e implemente un programa que haciendo uso de cada biblioteca permita:</p> <ul style="list-style-type: none"> <li>• Insertar vértices en el grafo</li> <li>• Insertar arista en el grafo</li> <li>• Eliminar vértices del grafo</li> <li>• Eliminar aristas del grafo</li> <li>• Realizar un recorrido en profundidad (DFS)</li> <li>• Realizar un recorrido en amplitud (BFS)</li> <li>• Obtener el árbol de recubrimiento mínimo correspondiente utilizando el algoritmo de Prim</li> <li>• Obtener el árbol de recubrimiento mínimo correspondiente utilizando el algoritmo de Kruskal</li> <li>• Determinar la ruta mínima para llegar de un vértice origen a todos los demás vértices del grafo (Dijkstra)</li> <li>• Determinar la ruta mínima para llegar de cualquier vértice origen a todos los demás vértices del grafo (Floyd-Warshall)</li> </ul> <p>Posteriormente modele en cada programa el grafo que aparece a continuación y mida los tiempos de ejecución de la implementación de cada algoritmo (de los anteriores) utilizando cada una de las bibliotecas tanto en su laptop como en una RPi o BeagleBone. Para cada uno de los algoritmos utilizados, analice su complejidad temporal y espacial.</p> <p>Finalmente, genere un reporte como resultado de la investigación donde analice los resultados obtenidos y en caso de haber diferencias entre los tiempos de ejecución utilizando una biblioteca u otra, realice una reflexión personal donde mencione con sus palabras cuáles son los factores que ocasionan dichas diferencias de tiempo.</p> <p>Como parte de la respuesta de este ejercicio debe incluir en el reporte de investigación (Pages) lo siguiente:</p> <ul style="list-style-type: none"> <li>• Tabla con los tiempos de ejecución de cada algoritmo utilizado de cada biblioteca, tanto en la BB / RPi como en la laptop / escritorio.</li> <li>• Para cada algoritmo, una gráfica comparativa con sus tiempos de ejecución en la BB / RPi y en la laptop / escritorio para cada biblioteca.</li> <li>• Análisis e interpretación de los resultados (con sus palabras).</li> </ul> <p>Suba a la plataforma el archivo Pages, Word, Google Docs o PDF (nombrado según la siguiente nomenclatura <i>TC2017_PP2_Matricula1_Matricula2</i>) e incluya en el mismo la liga al repositorio de GitHub que contenga todos los códigos programados.</p> <p>No se aceptan trabajos fuera de fecha ni por correo electrónico. En ambos casos la calificación de la tarea será 0 puntos.</p>

<b>Lugar en que se llevará a cabo</b>	Casa
<b>Forma de trabajo</b>	En equipos de 2 estudiantes (según se asignaron al inicio del semestre)
<b>Recursos</b>	Notas de clases Foros de información en Internet Wikipedia ( <a href="http://www.wikipedia.org">http://www.wikipedia.org</a> ) Códigos de grafos vistos en la materia Estructura de Datos Computadora
<b>Tiempo estimado</b>	10 horas
<b>Criterios de evaluación</b>	La evaluación se realizará de la siguiente manera:  Programar cada algoritmo (12) correctamente (60 puntos, 5 puntos por cada algoritmo) Implementación correcta del grafo (4 puntos) Programar el ejemplo que demuestre el uso de cada algoritmo (12 puntos) Obtener la complejidad de cada algoritmo (12 puntos). Obtener la técnica de diseño de cada algoritmo (12 puntos).
<b>Valor de la actividad</b>	20% de la calificación del segundo parcial

### **Límite de entrega de la tarea**

La tarea estará activa en la plataforma hasta el 23 de octubre de 2015 a las 23:55 horas.

### **Grafo a modelar**

