

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

ALGORITMIA

Examen 2

(Segundo semestre 2015)

Indicaciones generales:

- Duración: 3 horas.
- Materiales a utilizar: Apuntes y diapositivas de clase. **No se podrá usar código impreso.**
- Al inicio de cada programa, el alumno deberá incluir, a modo de comentario, la estrategia que utilizará para resolver el problema. De no incluirse dicho comentario o si la implementación es significativamente diferente a la estrategia indicada, el alumno no obtendrá el puntaje completo en dicha pregunta.
- Un programa que no muestre resultados coherentes y/o útiles será corregido sobre el 60 % del puntaje asignado a dicha pregunta.
- Debe utilizar comentarios para explicar la lógica seguida en el programa elaborado.
- El orden será considerado dentro de la evaluación.
- Cada programa debe ser guardado en un archivo con el nombre *preg#codigo_de_alumno.c* y subido a PAIDEIA en el espacio indicado por los Jefes de Práctica.
- La presentación, la ortografía y la gramática influirán en la calificación.

Puntaje total: 20 puntos

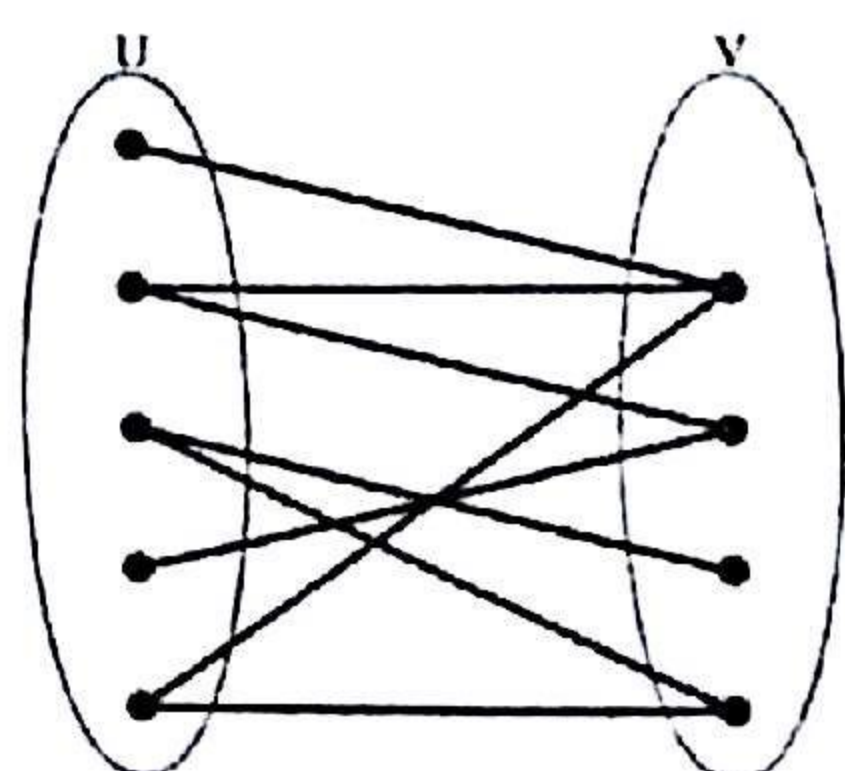
Cuestionario:

Pregunta 1 (6 puntos) Aplicación de TADs

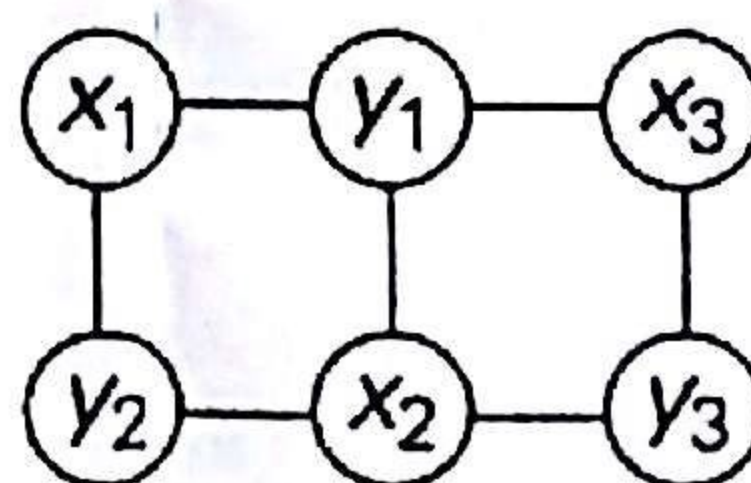
Indique el nombre o número de equipo en el que realizó el segundo trabajo grupal.

Pregunta 2 (5 puntos) Grafo Bipartito

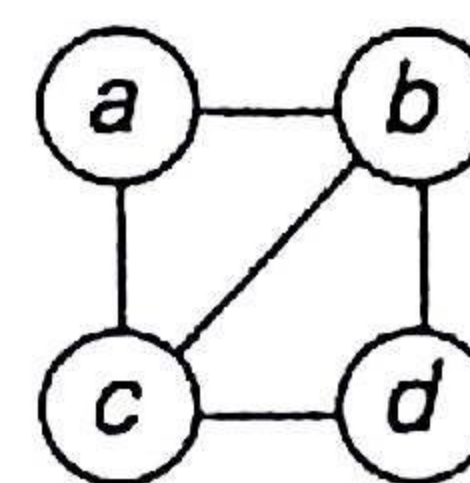
Un grafo es bipartito si sus vértices pueden ser divididos en dos conjuntos independientes, U y V , de tal forma que cada arista (u, v) o conecta un vértice de U con uno de V o un vértice de V como uno de U . En otras palabras, por cada arista (u, v) , o u pertenece a U y v a V , o u pertenece a V y v a U . También se puede decir que no existe alguna arista que conecte vértices del mismo conjunto. En la siguiente figura, los grafos I y II son bipartitos pero el grafo III no lo es.



(I)



(II)

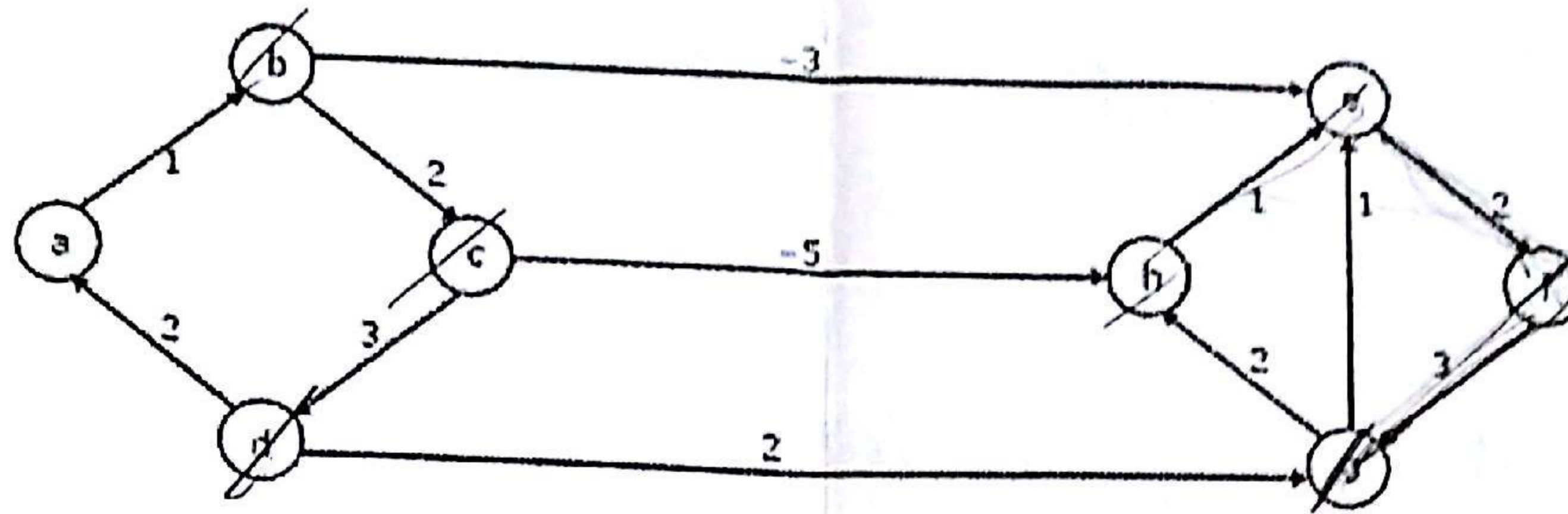


(III)

Implemente un programa en C que permita leer un grafo (usted establezca el formato de la entrada) y que determine si es bipartito o no. SUGERENCIA: ¿Qué sucede si en el grafo II a X_1 lo pintase de rojo, luego a sus vértices adyacentes los pinto de azul, luego a los adyacentes de estos los pinto de rojo, y así sucesivamente? ¿Qué sucede si realizo el mismo proceso con el grafo III?

Pregunta 3 (2 puntos) Caminos Más Cortos

Si se ejecuta el algoritmo de Dijkstra desde el vértice *a* del siguiente grafo, se logra calcular los caminos de distancias más cortas para: (A) solo el vértice *a*; (B) solo los vértices *a*, *c*, *f*, *g*, *h*; (C) solo los vértices *a*, *b*, *c*, *d*; (D) todos los vértices. Justifique su respuesta (ejecute el algoritmo si lo considera necesario).



Pregunta 4 (3 puntos) Hojas de Árbol

Implemente una función que reciba como parámetro un árbol binario e imprima sólo los nodos “no-internos” con un recorrido Post-Orden.

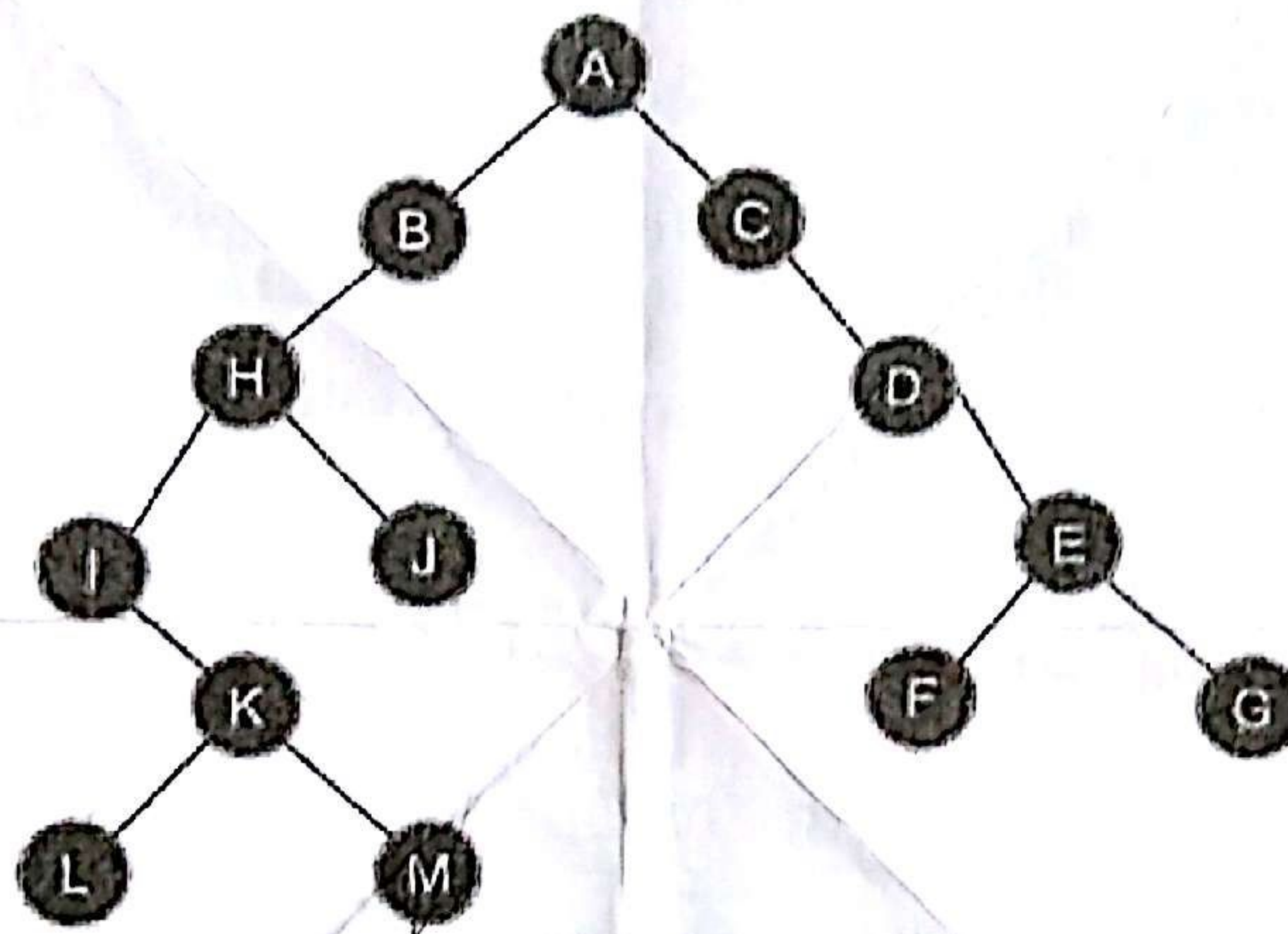


Figura 1: Los nodos “~~no~~” a imprimir son: L M J F G A
No internos.

Nota: Deberá implementar todos los TADs necesarios, así como la función main que verifique el correcto funcionamiento de la función solicitada.

Pregunta 5 (4 puntos) El Directorio de Casimiro

Casimiro desea crear un directorio telefónico que implemente una función `searchContact` (que dado el nombre de un abonado nos devuelva su número de teléfono) y una función `addContact` (que añade un nuevo contacto al directorio telefónico).

Casimiro decide implementar el Directorio Telefónico utilizando *Open Hashing* como método de resolución de colisiones. Además la función Hash a implementar debe ser de tipo *Mid-Square*. Implementar un programa en C que resuelva el problema descrito.

Nota: Deberá implementar todos los TADs necesarios, así como la función main que verifique el correcto funcionamiento de la función solicitada.

Profesores del curso: Fernando Alva
Robert Ormeño

Pando, 05 de diciembre de 2015