

## PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

### FACULTAD DE CIENCIAS E INGENIERÍA

#### ALGORITMIA

#### 5ta. práctica (tipo B) (Primer Semestre 2020)

##### Indicaciones Generales:

- Duración: 2h 50 min.
- 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, el alumno perderá el derecho a reclamo en esa pregunta.
- Los programas deben ser desarrollados en Ansi C. Si la implementación es diferente a la estrategia indicada o no la incluye, la pregunta no será corregida.
- Un programa que no muestre resultados coherentes y/o útiles será corregido sobre el 50% del puntaje asignado a dicha pregunta.
- Debe utilizar comentarios para explicar la lógica seguida en el programa elaborado.
- El orden será parte de la evaluación.
- Se utilizarán herramientas para la detección de plagios, por tal motivo si se encuentran soluciones similares, se anulará la evaluación a todos los implicados y se procederá con las medidas disciplinarias dispuestas por la FCI.
- Su trabajo deberá ser subido a PAIDEIA en el espacio indicado por los jefes de práctica.

##### **Pregunta 1 (10 puntos)**

Una empresa inmobiliaria dedicada a la construcción de edificios de departamentos, condominios y locales comerciales, anualmente realiza la selección de los proyectos que ejecutara durante este periodo. Cada proyecto tiene un costo de ejecución  $c_i$ . Se conoce que al año se proponen  $N$  proyectos. Para la selección de proyectos la principal limitante es que la suma de los costos de los proyectos elegidos debe tener un monto igual al presupuesto  $P$ , asignado por el área Financiera para todo el año. También se debe considerar que algunos proyectos son dependientes de otro para ser ejecutados. Por tal motivo no pueden elegirse si no se ha seleccionado su predecesor. Se sabe que como máximo un proyecto puede tener un predecesor y como mínimo pueden tener cero predecesores. A continuación mostramos un ejemplo del ingreso de 8 proyectos, las cantidades están expresadas en Millones de dólares:

$N = 8$

$P = 10$

Proyecto	Costo (c)	Predecesores
0	7	
1	1	
2	3	1
3	2	
4	4	2
5	5	4
6	1	
7	5	6

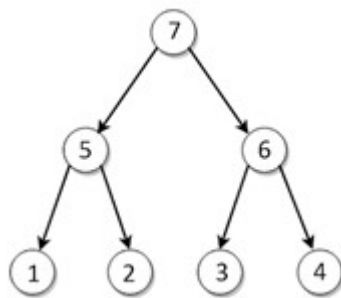
##### **Resultado:**

Hay soluciones: P0, P1, P3  
 P0, P3, P6  
 P1, P2, P3, P4

- a) Desarrolle las sentencias necesarias para el ingreso de datos al programa. Puede ser vía teclado o mediante archivos (1.0 punto).
- b) Desarrolle una función **seleccionauna** que utilizando backtracking, obtenga una solución al problema propuesto, si en caso no existen soluciones debe mostrarse en pantalla (4.0 puntos).
- c) Desarrolle una función **seleccionamultiple** que utilizando backtracking, obtenga todas las soluciones posibles al problema propuesto, si en caso no existen soluciones debe mostrarse en pantalla (5 puntos).

## Pregunta 2 (10 puntos)

Una empresa de desarrollo contrató a un programador para construir un sistema de base de datos desde cero. Uno de los desarrollos consistía en crear un sistema de búsqueda clave-valor dentro del sistema de base de datos, para lo cual la idea era que el programador cree un árbol de búsqueda binario que almacene las claves de la base de datos. El programador quiso emplear el método básico para crear un ABB desde datos existentes: ordenar la secuencia de claves y después insertarlas en el ABB. Desafortunadamente, el programador implementó mal el algoritmo, de tal forma que el árbol le quedó así:



Como puede verse, las claves están ordenadas por nivel y de abajo hacia arriba. La empresa ahora quiere que un nuevo desarrollador solucione el problema de este árbol. Como la empresa no quiere volver a crear el árbol desde cero, lo que se necesita es crear un recorrido que pueda extraer las claves en orden. Sin embargo, el nuevo algoritmo de recorrido debería preservar la complejidad computacional del algoritmo de recorrido in-orden como si el árbol estuviera correcto. Es decir, el algoritmo de recorrido a crear debe tener complejidad  $O(n)$ . Para llevar a cabo este algoritmo se le recomienda emplear TADs auxiliares.

Nota: el algoritmo completo para recorrer todo el árbol debe ser  $O(n)$ . Además para el proceso de creación e inserción puede usar el archivo **Lab5.c** que se encuentra en la tarea.

Profesores del curso: Johan Baldeon

Rony Cueva

Ivan Sipiran

San Miguel, 06 de julio del 2020