Apellidos:
Nombre:
Convocatoria:
DNI:

Examen PED junio 2014. GRADO
Modalidad 0

Normas:

- Tiempo para efectuar el test: **20 minutos**.
- Una pregunta mal contestada elimina una correcta.
- Las soluciones al examen se dejarán en el campus virtual.
- **Una vez empezado el examen no se puede salir del aula hasta finalizarlo.**
- En la **hoja de contestaciones** el verdadero se corresponderá con la **A**, y el falso con la **B**.

	V	F		
El tiempo requerido por un algoritmo expresado en función de la talla del problema se llama complejidad espacial del algoritmo.	❑	❑	1	F*
La complejidad temporal en su caso promedio del algoritmo de ordenación de intercambio directo (burbuja) visto en clase es $\Theta(n^2)$.	❑	❑	2	V*
Si en un árbol binario representado secuencialmente tenemos el nodo padre en la posición 5, sus hijos izquierdo y derecho se encuentran, respectivamente, en las posiciones 6 y 7.	❑	❑	3	F*
En el borrado de un elemento en un árbol 2-3-4, la altura del árbol siempre decrece cuando los punteros "q" y "r" son 2-nodo.	❑	❑	4	F*
Los árboles AVL son árboles balanceados con respecto a la altura de los subárboles.	❑	❑	5	V*
En las colas circulares enlazadas, el siguiente elemento apuntado por "fondo" es el primero a desencolar.	❑	❑	6	V*
En la inserción de un elemento en un árbol 2-3, la altura del árbol sólo aumenta cuando todos los nodos del árbol son 3-nodo.	❑	❑	7	F*
La complejidad temporal en el peor caso de la operación inserción en un árbol 2-3-4 es $O(\log_2(n+1))$.	❑	❑	8	V*
Se puede obtener un único árbol 2-3-4 a partir de su recorrido por niveles.	❑	❑	9	V*
La representación de conjuntos mediante vectores de bits tiene una complejidad espacial proporcional al tamaño del conjunto universal.	❑	❑	10	V*
La especificación algebraica de la siguiente operación indica que se devolverá el número de elementos del conjunto (C: Conjunto; x: Ítem): Operación(Crear) $\Leftrightarrow$ 0 Operación (Insertar(C, x)) $\Leftrightarrow$ 1 + Operación(C)	❑	❑	11	V*
En el TAD Diccionario con dispersión abierta, para evitar el problema del clustering secundario el tamaño de la tabla tiene que ser un número primo.	❑	❑	12	F*
Todo montículo o HEAP mínimo es un árbol binario lleno que además es árbol mínimo.	❑	❑	13	F*
Dado un grafo dirigido, siempre se cumple que Adyacencia_de_Salida(x) $\cap$ Adyacencia_de_Entrada(x) = $\emptyset$, donde x es un vértice del grafo.	❑	❑	14	F*
En un grafo no dirigido de "n" vértices pueden existir infinitas aristas.	❑	❑	15	F*

Examen PED junio 2014. GRADO

Normas:
- Tiempo para efectuar el ejercicio: **1 hora y 45 minutos**
- En la cabecera de cada hoja **Y EN ESTE ORDEN** hay que poner: ***APELLIDOS, NOMBRE***.
- Cada pregunta se escribirá en hojas diferentes.
- Las soluciones al examen se dejarán en el campus virtual.
- Se puede escribir el examen con lápiz, siempre que sea legible
- **Todas las preguntas tienen el mismo valor.**
- **Las fechas de "Publicación de notas" y "Revisión del examen teórico" se publicarán en el Campus Virtual.**

1. **a)** Utilizando exclusivamente las operaciones constructoras generadoras de lista, definir la sintaxis y la semántica de la operación *sublista* que actúa sobre una lista de números naturales y devuelve la lista original en la que se han eliminado los elementos que ocupan las posiciones pares y también los elementos iguales a un número natural especificado que ocupan las posiciones impares.

Nota: se pueden utilizar todas las operaciones definidas para números naturales.

Dado el grafo no dirigido representado por la lista de adyacencia que se muestra a continuación:

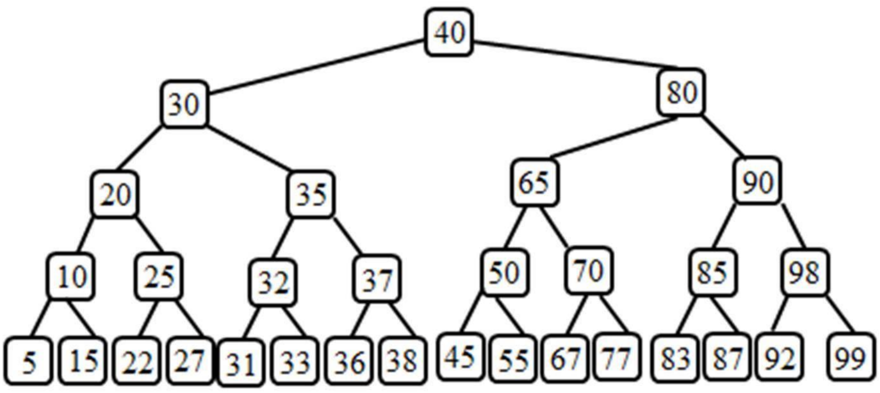
b) Obtener DFS(1), el árbol extendido en profundidad partiendo del vértice 1 y la clasificación de las aristas.

c) Define y calcula las componentes fuertemente conexas del grafo. ¿Es un grafo fuertemente conexo? Justifica tu respuesta

Nota: La lista de adyacencia de cada vértice se recorre de menor a mayor vértice para todos los casos del ejercicio. Las listas están desordenadas.

1 → 6→ 9 → 8 →7
2 → 3 → 6→ 5 → 7 →8 → 9
3 → 6 → 9→ 7 → 8 →5
4 → 6 → 8 → 9 → 3
5 → 6 → 1
6 → 8 → 9
7 → 8 → 9 → 10
8 → 9
9 → 10
11 → 13
12 → 11
14 → 11

2. Dado el siguiente árbol A:



a) Realiza el borrado del ítem 40 del árbol A, suponiendo que este árbol es un árbol 234, indicando número y tipo de transformaciones utilizadas.
Criterio 1: Consultar el hermano de la izquierda.
Criterio 2: Si el ítem a borrar no está en una hoja, sustituir por el mayor de la izquierda.

b) Realiza el borrado del ítem 80 del árbol A, suponiendo que este árbol es un árbol 23, indicando número y tipo de transformaciones utilizadas.
Criterio 1: Consultar el hermano de la izquierda.
Criterio 2: Si el ítem a borrar no está en una hoja, sustituir por el mayor de la izquierda.

c) Considerando el árbol A un árbol binario de búsqueda, muestra la representación secuencial en forma de vector del árbol binario A. Indica brevemente cómo realizarías el borrado del ítem 65 en este árbol ABB con esta representación secuencial.
Criterio: Si el ítem a borrar no está en una hoja, sustituir por el mayor de la izquierda.

Examen PED junio 2014. Soluciones

1.

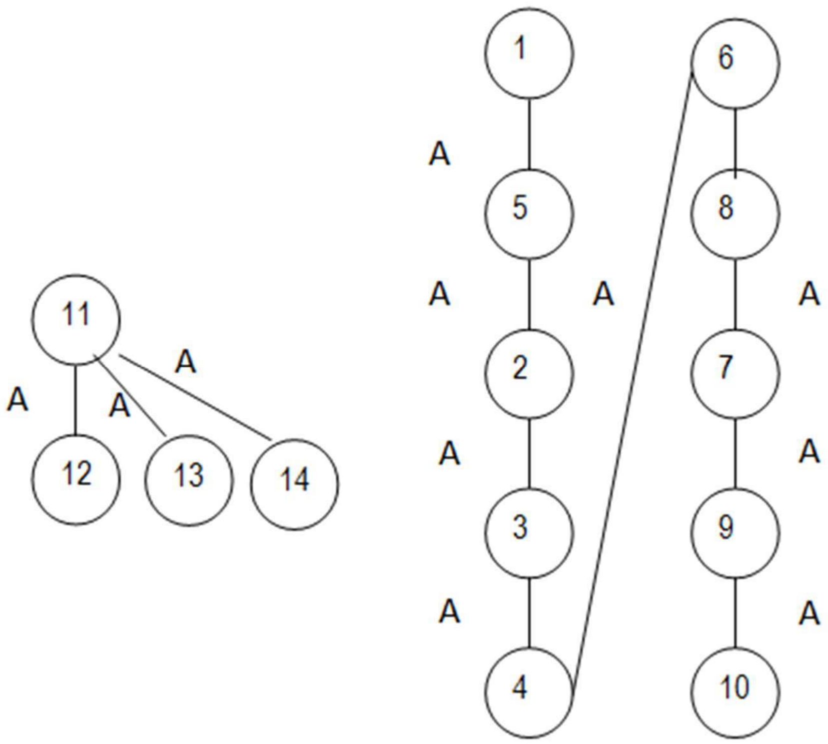
a)

sublista: lista , ítem → lista
Var l1:lista; x,y, Z:item;
sublista(crear_lista(),Z) = crear_lista()
sublista(IC(crear_lista(),x),Z) =
    si x==Z entonces crear()
      sino IC(crear_lista(),x)
sublista(IC(IC(l1,x),y),Z) =
    si y==Z entonces sublista(l1,Z)
      sino IC(sublista(l1,Z),y)

b)

DFS(1)=1,5,2,3,4,6,8,7,9,10. Se continúa por DFS(11)=11,12,13,14

Árbol extendido en profundidad. Las aristas marcadas son de árbol (A), el resto son de retroceso



c)

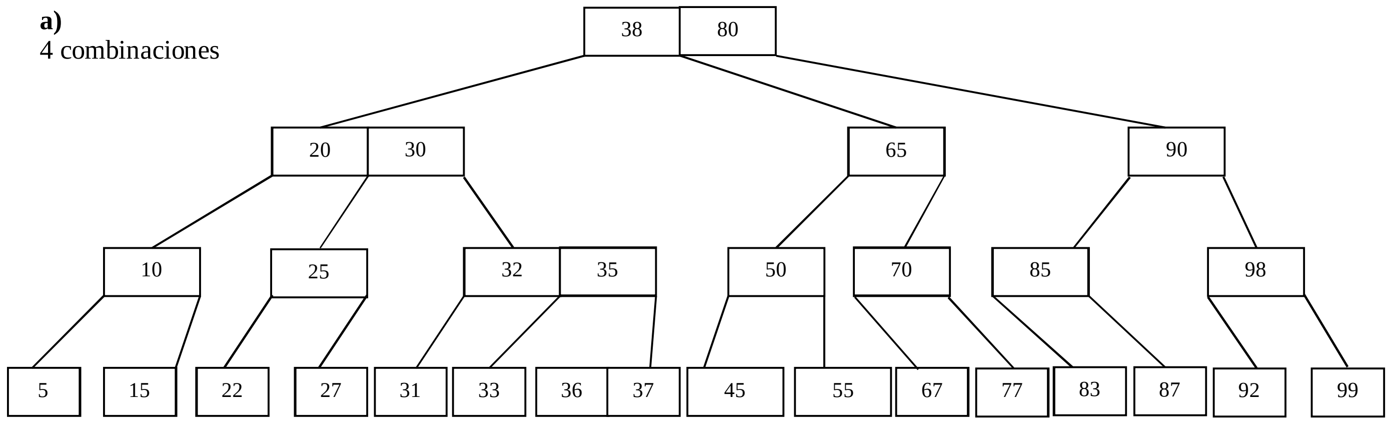
Componente Fuertemente conexa de un grafo: Conjunto maximal de vértices en el cual existe un camino que va desde cualquier vértice del conjunto hasta cualquier otro vértice también del conjunto.

GRAFO FUERTEMENTE CONEXO: aquel que tiene un sólo componente fuerte.

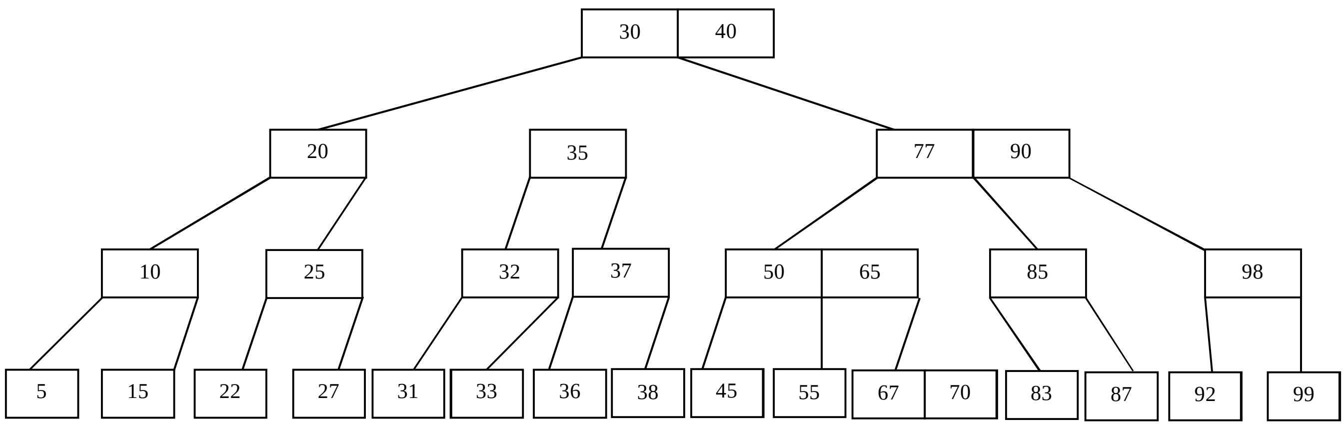
C1={1,2,3,4,5,6,7,8,9,10} y C2={11,12,13,14}. No es un grafo fuertemente conexo

2.

a)
4 combinaciones



b)
4 combinaciones



c)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
40	30	80	20	35	65	90	10	25	32	37	50	70	85	98	5	15	22	27	31	33	36	38	45	55	67	77	83	87	92	99

Para los movimientos descendentes, dada una casilla i del vector, se desplazará al hijo izquierda accediendo a la casilla 2i, y al hijo derecho accediendo a 2i+1. Para los movimientos ascendentes se accederá mediante la división entera por 2.

Para realizar el borrado del ítem 65, se comenzará en la raíz del ABB que está en la casilla 1 del vector y se seguirá el proceso de búsqueda habitual del ABB utilizando las fórmulas anteriormente descritas, hasta llegar al ítem 65. Puesto que dicha clave se encuentra en un nodo interior del árbol, se realizará la búsqueda del ítem mayor de su subárbol izquierdo y se procederá a su intercambio y posterior borrado, quedando el vector del siguiente modo:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
40	30	80	20	35	55	90	10	25	32	37	50	70	85	98	5	15	22	27	31	33	36	38	45	--	67	77	83	87	92	99