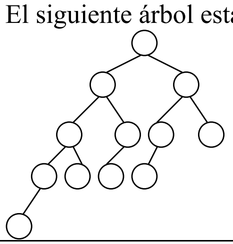
Apellidos:
Nombre:
Convocatoria:
DNI:

Examen PED junio 2012. Grado en Informática
Modalidad 0

Normas:
- La entrega del test no corre convocatoria.
- 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		
En C++, la instrucción "int &a = 1 ; " daría error de compilación	☑	☐	1	V
En C++, cuando se emplea layering, desde la clase A que contiene un objeto de la clase B siempre se puede acceder a la parte privada del objeto contenido de la clase B	☐	☑	2	F
Paso de programa es una secuencia de operaciones con contenido semántico cuyo coste es dependiente de la talla del problema	☐	☐	3	F
La semántica de la operación cima del tipo pila vista en clase es la siguiente: VAR p: pila, e: item; cima(crear( )) = error( ) cima(apilar(p, e)) = cima(p)	☐	☑	4	F
Dado un único recorrido de un árbol binario lleno, es posible reconstruir dicho árbol	☑	☐	5	V
En el borrado de un elemento que se encuentre en un nodo con dos hijos no vacíos en un árbol binario de búsqueda, tenemos que intercambiar el elemento a borrar por el mayor del subárbol de la izquierda o por el menor del subarbol de la derecha	☑	☐	6	V
El siguiente árbol está balanceado con respecto a la altura 	☑	☐	7	V
El borrado de un elemento en un árbol 2-3 se realiza en las hojas. Se pueden producir reestructuraciones del árbol aplicando el algoritmo descendente que empieza en la raíz del árbol y finaliza en las hojas.	☐	☑	8	F
En un árbol 2-3-4 el máximo número elementos del nivel N es 3*2N-1	☐	☑	9	F
La especificación algebraica de la siguiente operación eliminaría todas las claves repetidas de un determinado ítem (C: ConjuntoConClavesRepetidas; x, y: Ítem): Eliminar(Crear, x) ⇔ Crear Eliminar(Insertar(C, x), y) ⇔ si (x == y) entonces C sino Insertar(Eliminar(C, y), x)	☐	☑	10	F
En el TAD Diccionario con dispersión cerrada, los elementos se almacenan en una tabla de tamaño fijo.	☑	☐	11	V
Todo Heap Mínimo cumple las condiciones de ser un árbol binario y un árbol mínimo	☑	☐	12	V
En un multigrafo pueden existir infinitas aristas para un número "n" de vértices.	☑	☐	13	V
En un grafo dirigido con K arcos (el número máximo de arcos en el grafo) y N vértices, una complejidad de O(K) es equivalente a la complejidad de $O(N^2)$.	☑	☐	14	V

Examen PED junio 2012

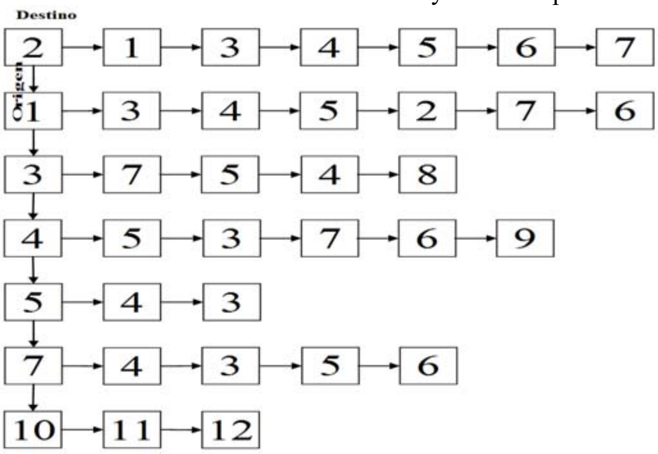
<u>**Normas:**</u>

- Tiempo para efectuar el ejercicio: **2 horas**
- En la cabecera de cada hoja **Y EN ESTE ORDEN** hay que poner: ***APELLIDOS, NOMBRE***.
- Cada pregunta se escribirá en hojas diferentes.
- Se dispone de 20 minutos para abandonar el examen sin que corra convocatoria.
- 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.**

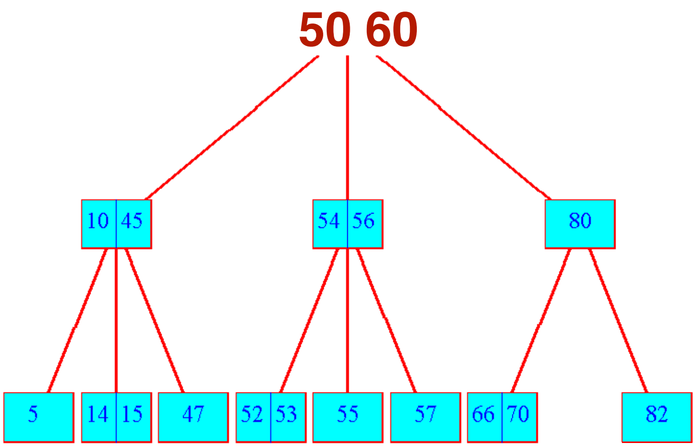
1. a) Sean las siguientes variables: G: Grafo; x, y, a, b:Vértice; p, q: Ítem; Completa en esta misma hoja la semántica de la operación BorrarOcurrencias (BO) que recibe un multigrafo y devuelve un multigrafo en el que se han eliminado todas las ocurrencias del arco <a, b> pasado como parámetro.

BO(CrearGrafo() , a, b) ↔ CrearGrafo()
BO(. . . . . , a, b) ↔ . . . .
. . . .
. . . .

b) Dado el grafo dirigido representado por la lista de listas que se muestra a continuación, calcular el recorrido DFS partiendo del vértice 4 y el árbol extendido en profundidad. Clasifica los arcos respondiendo en esta misma hoja. Nota: La lista de adyacencia de cada vértice se recorre de menor a mayor vértice para todos los casos del ejercicio.



2. Sobre el árbol 2-3 de la figura realiza las siguientes operaciones de forma consecutiva: Insertar 13, Borrar 50, Borrar 56, Borrar 14. Criterios: Si el nodo tiene dos hijos sustituir por el mayor de la izquierda, si el nodo tiene dos hermanos consultar el hermano de la derecha.



3. Sea un montículo doble (DEAP) inicialmente vacío.

a) Inserta los siguientes elementos: 3, 6, 10, 7, 12, 20, 5, 31, 36, 2, 4 y 1.
b) ¿Cuál es la complejidad temporal en el peor caso de la inserción en un montículo doble? Justifica tu respuesta.

4.

a) Insertar en una tabla de dispersión cerrada de tamaño B=7 inicialmente vacía, con función de dispersión H(x)= x MOD B, y con estrategia de redispersión "segunda función hash", los siguientes elementos: 23, 14, 9, 6, 30, 12, 18. Hay que detallar cada intento de inserción realizado, devolviendo el número total de intentos
b) ¿Qué coste tendría el insertar en la tabla resultante anterior el elemento 25? Justificar la respuesta
c) Si en lugar de utilizar un vector de tamaño B=7, utilizásemos una lista de tamaño 7, ¿qué coste temporal en su peor caso tendría la operación de inserción de una clave? Justificar la respuesta

d) Indica el número de intentos necesarios para realizar la inserción de las claves 3, 5, 1 en una tabla de dispersión cerrada de tamaño B=4 inicialmente vacía, con función de dispersión H(x)= x MOD B, y con estrategia de redispersión “segunda función hash”. Justifica tu respuesta.

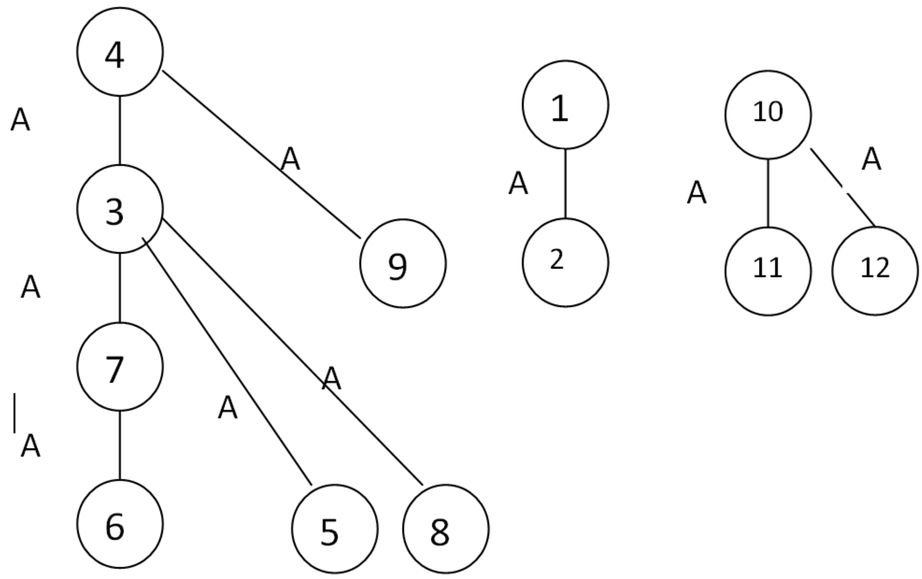
Examen PED junio 2012. Soluciones

1.

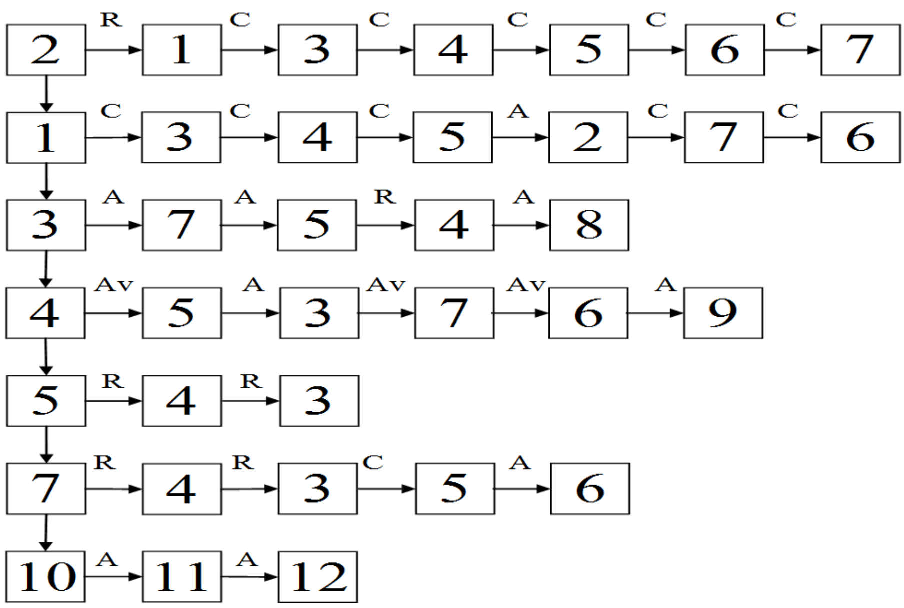
a) BO(C r e a r G r a f o () , a, b) ↔ CrearGrafo()

BO(InsertarArista(G, x, y, p), a, b) ↔ si (x == a) y (y == b)
entonces BO(G, a, b)
sino InsertarArista(BO(G, a, b), x, y, p)

b) DFS(4)= 4,3,5,7,6,8,9

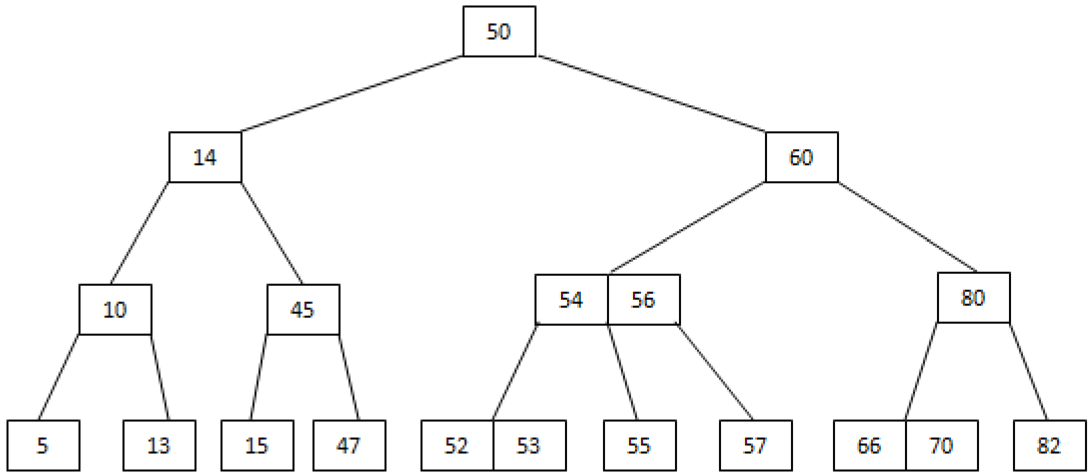


Clasificación de arcos

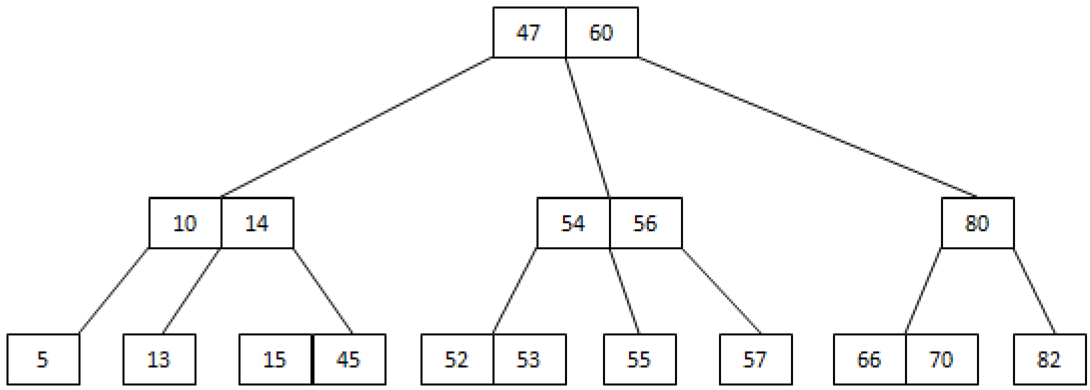


2.

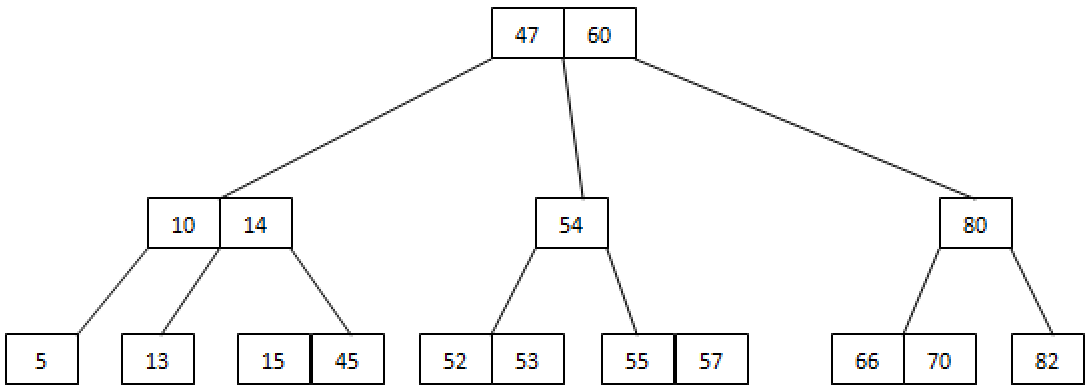
Insertar 13:



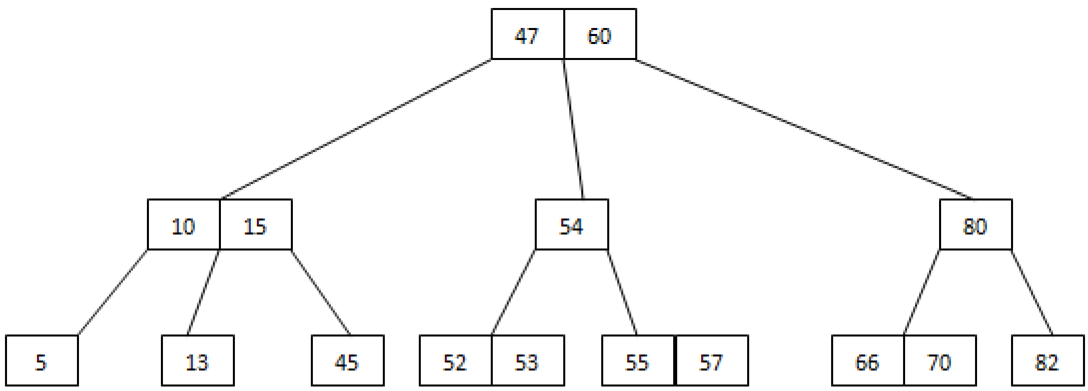
Borrar 50: Comb, comb, comb



Borrar 56: Comb

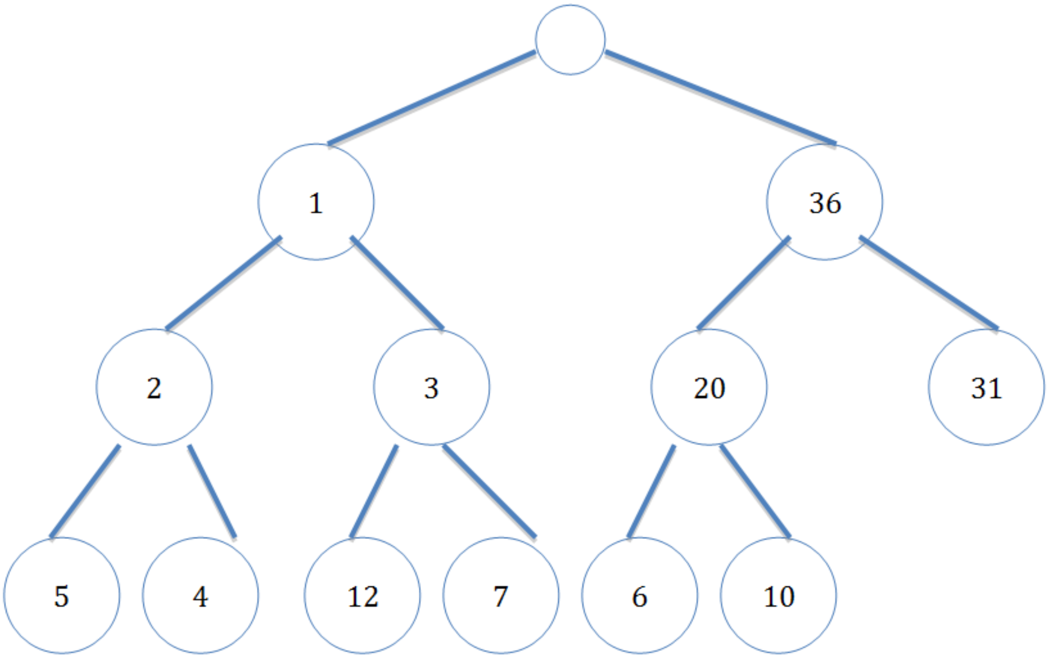


Borrar 14: Rot



3.

a)

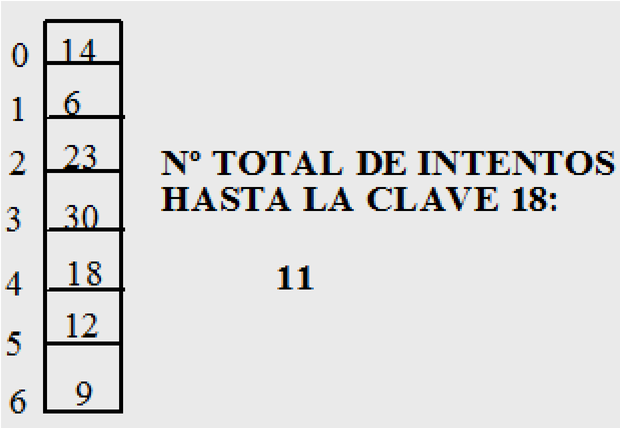


b) La complejidad de la inserción en un montículo doble en el peor caso es O($\log_2$ n). Como la cantidad de comparaciones que tenemos que hacer en el peor caso es igual al número de niveles que tiene el árbol, la complejidad en el peor caso será la altura del árbol. Como por definición un DEAP es un árbol binario completo, podemos afirmar que la altura de un DEAP es O($\log_2$ n).

4. a)

k (x) = (x MOD (B-1)) + 1

H(23)=23 MOD 7=*2*
H(14)=14 MOD 7=*0*
H(9)=9 MOD 7=2
    k(9)=(9 MOD 6)+1=4
    h_1(9)=(2+4) MOD 7=*6*
H(6)=6 MOD 7=6
    k(6)=(6 MOD 6)+1=1
    h_1(6)=(6+1) MOD 7=0
    h_2(6)=(0+1) MOD 7=*1*
H(30)=30 MOD 7=2
    k(30)=(30 MOD 6)+1=1
    h_1(30)=(2+1) MOD 7=*3*
H(12)=12 MOD 7=*5*
H(18)=18 MOD 7=*4*



b) En caso que se hiciese la comprobación previa de que la tabla está llena, tendría un coste de 1 solo paso. En caso contrario, habría que realizar B intentos hasta llegar a la posición de inicio, con lo que tendría un coste de B pasos. Tras alcanzar esa conclusión de tabla llena, se podría realizar la inserción tras un proceso de ampliación de la tabla a un tamaño B' mayor que B, reinsertando las B claves.

c) Para acceder a la posición lista[B] tendríamos que pasar por las B-1 posiciones anteriores, por lo que cada acceso tendría un coste de O(B). En el peor caso de una inserción, tendríamos que hacer B pasos, por lo que la complejidad sería de $O(B^2)$.

d) Las dos primeras claves se insertarían en un solo paso, pero la última (1) no se podría insertar (hecho que se comprobaría tras B intentos) a pesar de haber espacio en la tabla. Esta situación se da porque B no es un número primo, y al ser k(1) = 2, B y k(1) comparten factores primos comunes mayores que uno, por lo que no explorará todas las posiciones de la tabla