KING SAUD UNIVERSITY

COLLEGE OF COMPUTER & INFORMATION SCIENCES

DEPARTMENT OF COMPUTER SCIENCE

CSC212 Data Structures
Second Semester 1425/1426 AH

Final Examination:	Thursday 25.04.1426 A.H./02.06.2005 C.E.
Duration:	3 Hours
Instructors:	Dr. Aqil Azmi and Mr. Saad Al-Ahmadi

---

1. [Marks 10]

Answer True or False. No need to state the reason (أكتب بوضوح ولا داعي للتوضيح).

a. If $f_1(n)$ is $O(g_1(n))$ and $f_2(n)$ is $O(g_2(n))$ then $f_1(n) + f_2(n)$ is $O(\min(g_1(n), g_2(n)))$.

b. In the max heap the smallest key will be in one of the leaf nodes.

c. There is always a collision when using a perfect hashing function.

d. The height of a BST will never exceed $O(\log n)$.

e. Load factor measures the efficiency of the hash table.

f. With threaded trees we can do sequential preorder traversal without using stacks.

g. The main objective of B-Tree is to maximize the number of keys while minimizing the height of the tree.

h. A binary tree with n nodes is full if $n = 2^{\ell} - 1$ where ℓ is its height.

i. Any sorted array is a heap.

j. ADT stands for Advanced Data Types.

2. [Marks 10]

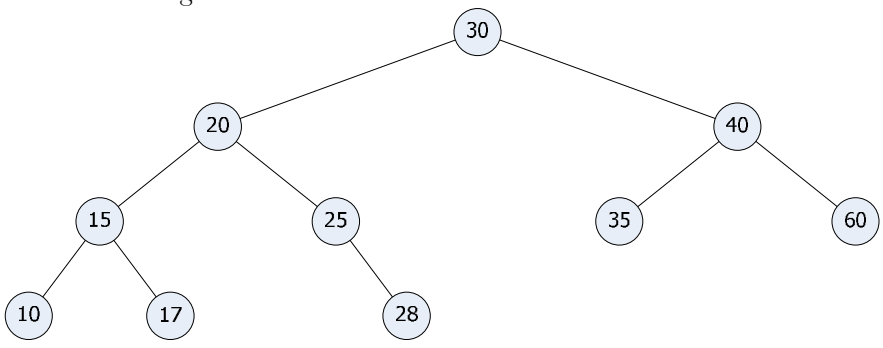
For each of the following, find the best big-O and the corresponding C and n_0.

a. $f(n) = 5n^4 - 6n^3 + 20$.

b. $f(n) = \log(n^2 + 1)$.

3. [Marks 20]

Consider the following AVL tree:



Draw the resulting tree after each of the following operations (كلّ فقرة تكمّل الفقرة السابقة):

a) Insert 5.

b) Delete 30.

c) Insert 32.

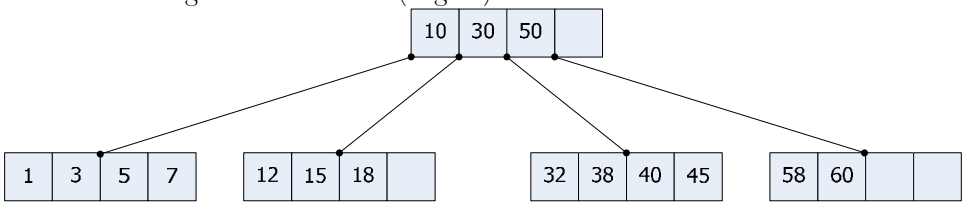
d) Insert 26.

e) Delete 10.

(OVER)

4. [Marks 20]

Given the following B-Tree of order (degree) $d = 2$.



Draw the resulting tree after each of the following operations (كلّ فقرة تكمّل الفقرة السابقة):

a) Insert 48.
b) Insert 9.
c) Delete 7.
d) Delete 5.

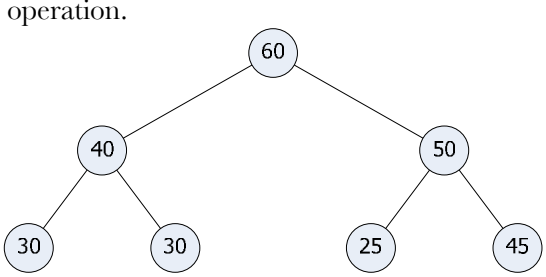
5. [Marks 10]

Sort the following array of integers using Merge Sort. Show all the necessary steps.

77	55	15	20	85	30	35	10	60	40	50	25

6. [Marks 10]

Write the correct key sequence starting with the key 30 in order to generate the heap below through siftup operation.



7. [Marks 10]

Write the member function:

```
template <class T>
int AVLTree<T>::height();
```

which returns the height of the AVL tree by tracing only one path to a leaf, rather than investigating all the nodes in the tree. You may assume the existence of a member function `int AVLTree<T>::BF(AVLNode<T> *p)` that returns the balance factor of the node `p`.

8. [Marks 10]

Write the client function:

```
void create(int heap[], int n, BT<int> &t);
```

Input: array `heap` with `n` elements (root at position 1, last node at position `n`).
Output: a binary tree (node and link based) representation of the heap. See the example below.