Lab3.4 Q2(c)-Q10

Name: YUEN Yiu Yeung StudentNo:200171873 Class:IT114105/1C

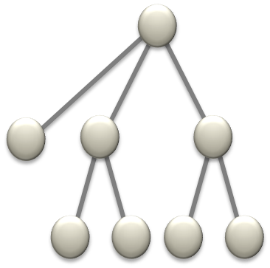
Declaration:  
I understand the meaning of academic dishonesty, in particular plagiarism, copyright infringement and collusion. I am aware of the consequences if found to be involved in these misconducts. I hereby declare that the work submitted for the “ITP4510 Data Structures & Algorithms” is authentic record of my own work.

Q2(c)

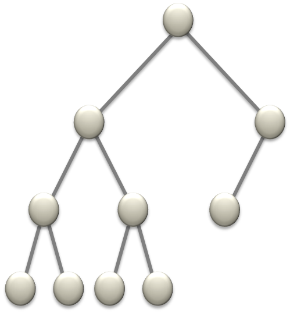
|  |  |  |  |
| --- | --- | --- | --- |
| Word to search for | Results | Time needed (BST) | Time needed (linked list) |
| water | □ Found □ Not Found | 10300 | 333500 |
| ever | □ Found □ Not Found | 26700 | 17000 |
| snail | □ Found □ Not Found | 9200 | 165900 |
| better | □ Found □ Not Found | 7400 | 98300 |
| apple | □ Found □ Not Found | 8600 | 72900 |
| door | □ Found □ Not Found | 11200 | 267900 |
| foolish | □ Found □ Not Found | 10600 | 85500 |

Q3.

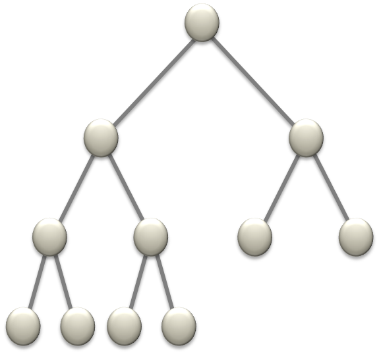
* A general tree



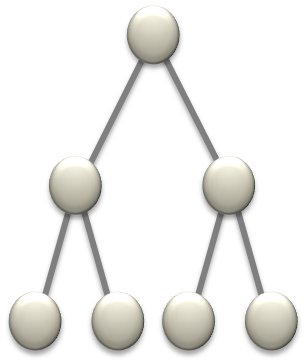
* A binary tree



* A proper binary tree



* A complete binary tree



Q4.

Number of leaf nodes = N / 2 + 1

Number of non-leaf nodes = N / 2

Q5.

Total number nodes in the tree: 28+1 – 1 = 511

Let d be the depth

Total number of nodes of a complete binary tree: 2d+1 - 1

d = log2(Total number of nodes + 1) - 1

Q6.

(a)

0 1 2 3 4 5 6 7 8 9

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Q | B | U |  | G | R | W |  |  | E |

10 11 12 13 14 15 16 17 18 19

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| J |  |  |  |  |  |  |  |  |  |

20 21 22 23 24 25 26 27 28 29

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| F | I | P |  |  |  |  |  |  |  |

30 31 32 33 34 35 36 37 38 39

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |

40 41 42 43 44 45 46 47 48 49

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | H |  |  |  |  |  |  |

50 51 52 53 54 55 56 57 58 59

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |

60 61 62

|  |  |  |
| --- | --- | --- |
|  |  |  |

(b)

(i) pre-order traversal: Q B G E F J I H P U R W

(ii) in-order traversal: B E F G H I J P Q R U W

(iii) post-order traversal: F E H I P J G B R W U Q

Q7. (Attached scan photo of hand-writing)

Q8.

BinaryNode search (BinaryNode t, key x)

begin

if t is null

return null;

if (x is less than t.data.key)

return search (t.left, x);

else if (x is greater than t.data.key)

return search (t.right, x);

else

return t;

end

Q9.

7(a) is an ordinary binary tree

7(b) is a complete binary tree

7(c) is an almost complete binary tree (non-strictly)

7(d) is an almost complete binary tree (strictly)

7(e) is a binary tree of N nodes with depth N-1. All non-leaf nodes only have a right son. In fact, it is similar to a linear list.

7(f) is same as (e) where non-leaf nodes only have a left son.

Q10.

Postfix expression: A B + C \* D E + / F \*

Reason: (Attached scan photo of hand-writing)

Prefix expression: \* / \* + A B C + D E F