Binary Tree

1. จาก tree ต่อไปนี้จงตอบคำถาม

A group of yellow circles with black text

Description automatically generated with low confidence

1. Node ที่เป็น Root ได้แก่ ......................................................................................................................................................................

2. Node ที่เป็น leaf ได้แก่ ........................................................................................................................................................................

3. เมื่อพิจารณา Node ที่มีค่าเป็น 4 Node ใดบ้างที่เป็น Child ของ node ดังกล่าว .....................................................................................

4. เมื่อพิจารณา Node ที่มีค่าเป็น 4 Node ใดบ้างที่เป็น Parent ของ node ดังกล่าว ..................................................................................

5. เมื่อพิจารณา Node ที่มีค่าเป็น 5 Node ใดบ้างที่เป็น Parent ของ node ดังกล่าว ..................................................................................

6. เมื่อพิจารณา Node ที่มีค่าเป็น 5 Node ใดบ้างที่เป็น Parent ของ node ดังกล่าว ..................................................................................

7. หาก travers tree ดังกล่าว แบบ Pre-order จะได้ output เป็น ...........................................................................................................

8. หาก travers tree ดังกล่าว แบบ In-order จะได้ output เป็น .............................................................................................................

9. หาก travers tree ดังกล่าว แบบ Post-order จะได้ output เป็น .........................................................................................................

1. จงเขียนแผนภาพของการทำงานของ Binary Tree ในโปรแกรมต่อไปนี้ที่ละบรรทัด และตอบคำถามเกี่ยวกับการท่อง (Traversal) ไปใน tree ดังกล่าว
2. Node \*root = new Node('H');
3. root->left  = new Node('A');
4. root->right = new Node('I');
5. (root->left) ->left  = new Node('R');
6. (root->left) ->right = new Node('U');
7. (root->right)->left  = new Node('5');
8. ((root->left)->right)->left = new Node('H');
9. ((root->right)->left)->right = new Node('5');

1.

A picture containing sketch, symbol, white, clipart

Description automatically generated

2.

A picture containing sketch, drawing, clipart, white

Description automatically generated

3.

A picture containing sketch, line art, clipart, drawing

Description automatically generated

4.

5.

6.

7.

8.

หาก travers tree ดังกล่าว แบบ Pre-order จะได้ output เป็น ..................................................................................................................

หาก travers tree ดังกล่าว แบบ In-order จะได้ output เป็น ........... R A H U H 5 5 I...............................................................................

หาก travers tree ดังกล่าว แบบ Post-order จะได้ output เป็น ................................................................................................................

1. จงเขียนแผนภาพของการทำงานของ Binary Tree ในโปรแกรมต่อไปนี้ที่ละบรรทัด และตอบคำถามเกี่ยวกับการท่อง (Traversal) ไปใน tree ดังกล่าว (ออกแบบบรรทัดเองเลยครับ)
2. Node \*root = new Node('n');
3. root->left  = new Node('A');
4. root->right = new Node('a');
5. (root->left) ->left  = new Node('t');
6. (root->left) ->right = new Node('k');
7. (root->right)->left  = new Node('i');
8. ((root->left)->right)->left = new Node('c');
9. ((root->right)->left)->right = new Node('t');
10. ((root->left)->left)->left = new Node('A');
11. ((root->left)->left)->right = new Node('t');
12. ((root->left)->right)->right = new Node('O');
13. ((root->right)->left)->left = new Node('T');
14. ((root->right)->right) = new Node('4');
15. ((root->right)->right)->left = new Node('n');
16. ((root->right)->right)->right = new Node('4');

หาก travers tree ดังกล่าว แบบ Pre-order จะได้ output เป็น ...........................................................................................................

หาก travers tree ดังกล่าว แบบ In-order จะได้ output เป็น .............................................................................................................

หาก travers tree ดังกล่าว แบบ Post-order จะได้ output เป็น .........................................................................................................

1. จงเขียนแผนภาพของการทำงานของ Binary Tree ในโปรแกรมต่อไปนี้ที่ละบรรทัด และตอบคำถามเกี่ยวกับการท่อง (Traversal) ไปใน tree ดังกล่าว (ออกแบบบรรทัดเองเลยครับ)

1.    Node \*root = new Node('H');

2.    root->right = new Node('i');

3.    ((root->right)->right) = new Node('g');

4.    ((root->right)->right)->right = new Node('h');

5.    (((root->right)->right)->right)->right = new Node('5');

หาก travers tree ดังกล่าว แบบ Pre-order จะได้ output เป็น ...........................................................................................................

หาก travers tree ดังกล่าว แบบ In-order จะได้ output เป็น .............................................................................................................

หาก travers tree ดังกล่าว แบบ Post-order จะได้ output เป็น .........................................................................................................

1. Tree มีข้อดีกว่า Linked list อย่างไร (ขอสั้นๆ) ......................................................................................................................................................................................................................................................................................................................
2. Linked list มีข้อดีกว่า Tree อย่างไร (ขอสั้นๆ) ......................................................................................................................................................................................................................................................................................................................
3. ในการ insert และ find ข้อมูลใน binary tree มีความต่างในด้านประสิทธิภาพต่างจาก linked list queue และ stack หรือไม่ (ขอยาวๆ)

...................................................................................................................................................................................................................................................................................................................... ........................................................................................................................................................... ...........................................................................................................................................................

...................................................................................................................................................................................................................................................................................................................... ........................................................................................................................................................... ...........................................................................................................................................................

...................................................................................................................................................................................................................................................................................................................... ........................................................................................................................................................... ...........................................................................................................................................................