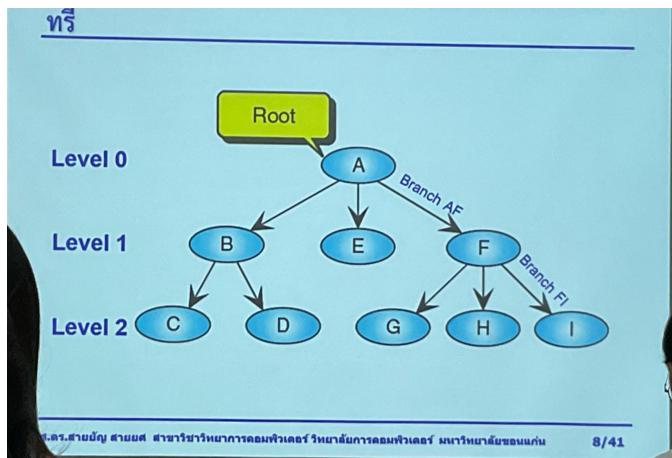


CP411107 Data Structures and Algorithms โครงสร้างข้อมูลแบบที่

องค์ประกอบของทรี

- โหนด (Node) ที่ใช้สำหรับบรรจุข้อมูล
- โดยจะมีกี่ชั้นเป็นส้นที่เรียกว่าโหนดเด็กด้วยกันที่เรียกว่า支 (Branch)
- จำนวนของบรรณ์ที่ล้มพังรักบินโหนดเรียกว่า ดีกรี (Degree)
- ถ้าหากทรีนี้เป็นทรีที่ไม่ใช่รากว่าโหนดแรกจะเรียกว่า Root (Root) โดยโหนดทุก ๆ โหนดยกเว้นรากโหนดจะมีเพียง 1 Predecessor (บรรพบุรุษ) ในขณะที่ Successor จะเป็น 0 หรือ 1 หรือมากกว่า 1 ก็ได้
- สำหรับ Leaf ก็คือโหนดใบที่ไม่มีบรรณ์เชื่อมโยงไปยังโหนดตัวถัดไป หรือโหนดที่ไม่มีตัวตามหลังหรือ Successor นั้นเองในขณะที่โหนดพ่อ (Parent) จะมีโหนดตามหลังหรือโหนดลูก (Child) ต่อท้าย โหนดลูกตั้งแต่สองโหนดหรือมากกว่าที่มาจากการพ่อเดียวกันจะเรียกว่า โหนดพี่น้อง (Siblings)

ผศ.ดร.สายสูญ สายสูญ สาขาวิชาการคอมพิวเตอร์ วิทยาลัยการคอมพิวเตอร์ มหาวิทยาลัยขอนแก่น 6/41



CP411107 Data Structures and Algorithms โครงสร้างข้อมูลแบบที่

ทรี

- โหนดราก (Root Node) คือ โหนด A ซึ่งเป็นโหนดรากหรือรากโหนดก็ได้เป็นโหนดแรกสำหรับโครงสร้างทรีที่ใช้เป็นโหนดเริ่มต้นเพื่อขยายลูกโหนดต่อไป
- โหนดพ่อ (Parents) คือโหนด A, B และ F ซึ่งก็คือโหนดที่มี Successor
- โหนดลูก (Children) คือโหนด B, E, F, C, D, G, H และ I ซึ่งก็คือโหนดที่มี Predecessor
- โหนดพี่น้อง (Siblings) คือโหนด {B,E,F},{C,D} และ {G,H,I}
- โหนดใบ (Leaf Node) คือโหนด C, D, E, G, H และ I ซึ่งก็คือโหนดที่ไม่มี Successor

สายสูญ สาขาวิชาการคอมพิวเตอร์ วิทยาลัยการคอมพิวเตอร์ มหาวิทยาลัยขอนแก่น

ทรี

- ดีกรี (Degree) ทั้งหมดของทรี มีค่าเท่ากับ 8
- ดีกรี (Degree) ทั้งหมดของโหนด F มีค่าเท่ากับ 3
- จำนวนของ Leaf โหนด มีค่าเท่ากับ 6

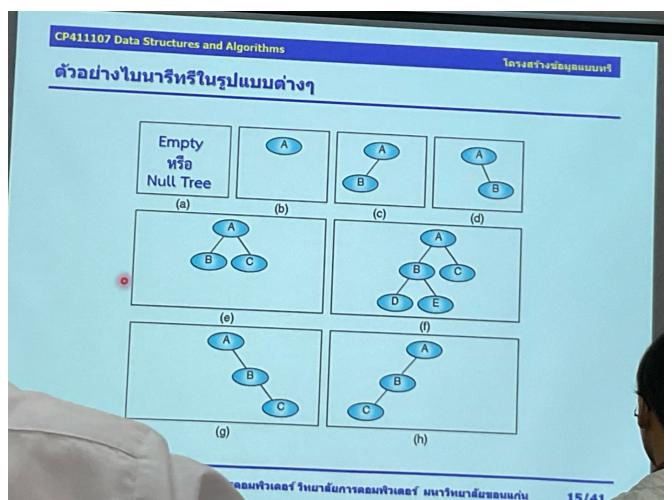
CP411107 Data Structures and Algorithms โครงสร้างข้อมูลแบบที่

ผศ.ดร.สายสูญ สายสูญ สาขาวิชาการคอมพิวเตอร์ วิทยาลัยการคอมพิวเตอร์ มหาวิทยาลัยขอนแก่น 15/41

CP411107 Data Structures and Algorithms โครงสร้างข้อมูลแบบที่

ขับทรี (Subtrees)

สายสูญ สาขาวิชาการคอมพิวเตอร์ วิทยาลัยการคอมพิวเตอร์ มหาวิทยาลัยขอนแก่น 12/41



CP411107 Data Structures and Algorithms
โครงสร้างข้อมูลแบบทรี

ความสูงของไบนารีทรี (Height of Binary Tree)

- ความสูงมากที่สุดของทรี (Maximum Height)

$$H_{\max} = N$$
 - ตัวอย่าง มีโหนดจำนวน 3 โหนดแล้วต้องการจัดเก็บไว้ในไบนารีทรี ความสูงมากที่สุดของทรีนี้คือ $H_{\max} = 3$
- ความสูงน้อยที่สุดของทรี (Minimum Height)

$$H_{\min} = \lceil \log_2 N \rceil + 1$$
 - ตัวอย่าง มีโหนดจำนวน 3 โหนดที่ต้องการจัดเก็บไว้ในไบนารีทรี ความสูงน้อยที่สุดของทรีนี้คือ $H_{\min} = 2$

ผศ.ดร.ภานุสุก สามษพ ภาควิชาวิทยาการคอมพิวเตอร์ วิทยาลัยการคอมพิวเตอร์ มหาวิทยาลัยขอนแก่น 17/41

CP411107 Data Structures and Algorithms
โครงสร้างข้อมูลแบบทรี

ความสูงของไบนารีทรี (Height of Binary Tree)

- จำนวนโหนดน้อยที่สุด (Minimum Nodes)

$$N_{\min} = H$$
 - ตัวอย่าง ทรีต้นหนึ่งมีความสูงเท่ากับ 3 อย่างทรายว่าจำนวนโหนดที่สามารถมีได้น้อยที่สุดเท่าไหร่ $N_{\min} = 3$
- จำนวนโหนดมากที่สุด (Maximum Nodes)

$$N_{\max} = 2^H - 1$$
 - ตัวอย่าง ทรีต้นหนึ่งมีความสูงเท่ากับ 3 อย่างทรายว่าจำนวนโหนดที่สามารถมีได้มากที่สุดเท่าไหร่ $N_{\max} = 7$

ผศ.ดร.ภานุสุก สามษพ ภาควิชาวิทยาการคอมพิวเตอร์ วิทยาลัยการคอมพิวเตอร์ มหาวิทยาลัยขอนแก่น 18/41

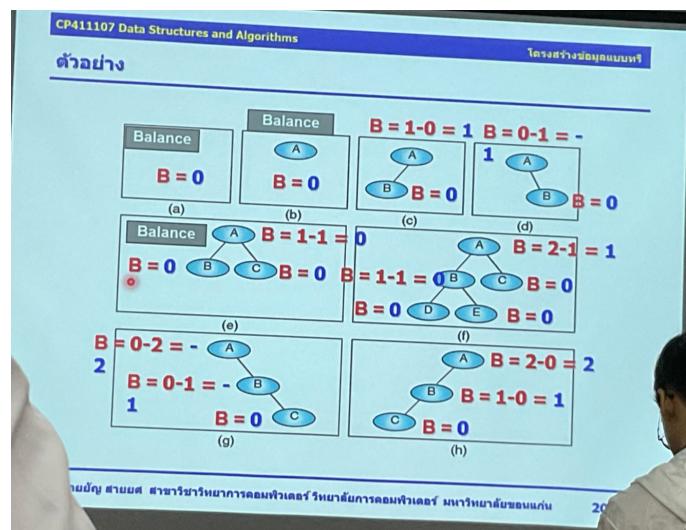
CP411107 Data Structures and Algorithms
โครงสร้างข้อมูลแบบทรี

ความสมดุล (Balance)

- ความสมดุลของไบนารีทรีจะสามารถทราบได้จากค่า Balance Factor เท่ากับ 0 ซึ่งค่าดังกล่าวคำนวณได้จากการนำความสูงของซ้ายหรือด้านซ้าย (H_L) มาหักลบกับความสูงของขวาหรือด้านขวา (H_R)

$$B = H_L - H_R$$

ผศ.ดร.ภานุสุก สามษพ ภาควิชาวิทยาการคอมพิวเตอร์ วิทยาลัยการคอมพิวเตอร์ มหาวิทยาลัยขอนแก่น 19/41



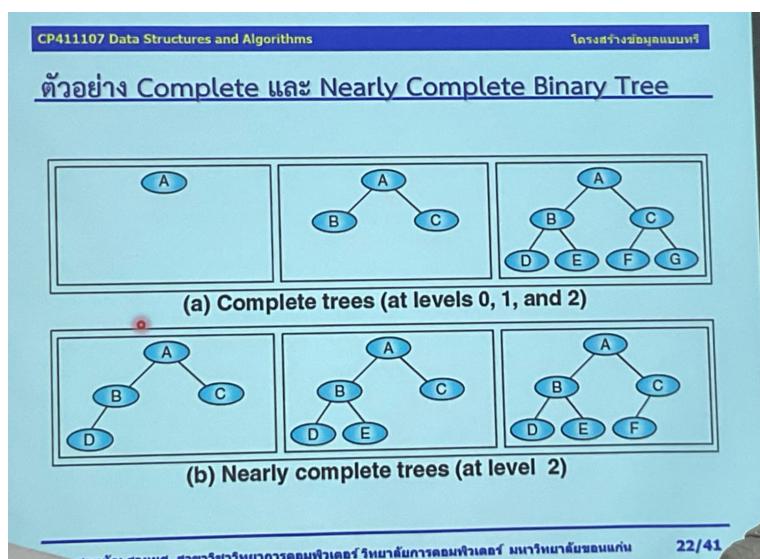
CP411107 Data Structures and Algorithms
โครงสร้างข้อมูลแบบทรี

ไบนารีทรีแบบสมบูรณ์และเกือบสมบูรณ์

ไบนารีทรีแบบสมบูรณ์และเกือบสมบูรณ์ (Complete and Nearly Complete Binary Trees)

- Complete Binary Tree เป็นทรีที่มีซ้ายหรือด้านซ้ายและขวาหรือด้านขวาเท่ากัน
- Nearly Complete Binary Tree เป็นทรีที่มีโหนดเดิมทุกโหนดยกเว้นใน Level สุดท้ายที่มีโหนดเฉพาะทางด้านซ้าย

ผศ.ดร.ภานุสุก สามษพ ภาควิชาวิทยาการคอมพิวเตอร์ วิทยาลัยการคอมพิวเตอร์ มหาวิทยาลัยขอนแก่น 21/41



CP411107 Data Structures and Algorithms
การแทนในอาร์เรียด้วยตัวอักษร

- รูปแบบของใบnaireที่แบบสมบูรณ์

ใบnaireที่แบบสมบูรณ์

ใบnaireที่แบบสมบูรณ์

อาร์เรย์

2n+1
Index ของ

2n+2
Index ของ

24/41

CP411107 Data Structures and Algorithms
การแทนในอาร์เรียด้วยตัวอักษร

- รูปแบบของใบnaireที่แบบไม่สมบูรณ์หรือเกือบสมบูรณ์

การเปลี่ยนใบnaireแบบไม่สมบูรณ์ให้เป็นสมบูรณ์ได้ด้วยการเพิ่มโนดว่างให้เดิม

อาร์เรย์

25/41

CP411107 Data Structures and Algorithms
การแทนในอาร์เรียด้วยลิกกิลส์ต์ Search by recursive function

26/41

CP411107 Data Structures and Algorithms
วิธีการท่องแบบแนวลึก (Depth-First Traversals)

- เป็นการท่องเข้าไปในทรีด้วยการเดินทางผ่านจักรุทโน้นลง

(a) Preorder traversal (b) Inorder traversal (c) Postorder traversal

28/41

CP411107 Data Structures and Algorithms
แบบ Preorder

N L R เดินทางไปที่ซ้ายแล้วขวา

algorithm preOrder (val root <node pointer>)

- if (root is not null)
 - process(root)
 - preOrder(root->leftSubtree)
 - preOrder(root->rightSubtree)
- return

(a) Processing order

(b) "Walking" order

29/41

CP411107 Data Structures and Algorithms
แบบ Inorder

L N R เดินทางไปซ้ายแล้วขวา

algorithm inOrder (val root <node pointer>)

- if (root is not null)
 - inOrder(root->leftSubtree)
 - process(root)
 - inOrder(root->rightSubtree)
- return

(a) Processing order

(b) "Walking" order

30/41

CP411107 Data Structures and Algorithms
ໄຊໂຄສະໜັບນິພຈົນ

ແບບ Postorder

L R N ໄຊເຕີກົດທີ່ກ່ອນທີ່ໄດ້

```
algorithm postOrder (val root <node pointer>)
1 if (root is not null)
  1 postOrder(root->leftSubtree)
  2 postOrder(root->rightSubtree)
  3 process(root)
2 return
```

ນາຄ.ສາຍນັກ ສາມພະ ສາຂາວິຊາວິທະຍາກອນທີ່ກ່ອນທີ່ໄດ້ ວິທະຍາກົດທີ່ກ່ອນທີ່ໄດ້ ມາຮັດກຳນົມຂອງພັກ
31/41

CP411107 Data Structures and Algorithms
ໄຊໂຄສະໜັບນິພຈົນ

ເອັກຝົບເພື່ອສັນທິ Expression Tree

ເອັກຝົບເພື່ອສັນທິ ອີ່ໃບນາຣີທີ່ຈຶ່ງແກ່ນິພຈົນ ຈຶ່ງມີຄຸນສົມບັດ ດັ່ງນີ້

- ດ້ວຍກຳດໍາເນີນກາຮືອໂໂປຣແກຣມດໍາລັດໄວ້ທີ່ໂທນດໃນ (Leaf Node)
- ຕ້ວດໍາເນີນກາຮືອໂໂປຣແຕວຈະເກີນໄວ້ທີ່ຽຸກໂທນດທີ່ໂທນດໄນ (Internal Nodes) ທີ່ໄຟໃຫ້ໂທນດໃນ
- ຂັບທີ່ ໃນທີ່ນີ້ກີ່ຄົນິພຈົນຢ່ອຍ ໂດຍຈະມີຽຸກໂທນດເປັນ ໂອໂປຣແຕວ

ນາຄ.ສາຍນັກ ສາມພະ ສາຂາວິຊາວິທະຍາກອນທີ່ກ່ອນທີ່ໄດ້ ວິທະຍາກົດທີ່ກ່ອນທີ່ໄດ້ ມາຮັດກຳນົມຂອງພັກ
33/41

CP411107 Data Structures and Algorithms
ໄຊໂຄສະໜັບນິພຈົນ

Expression Tree

ນາຄ.ສາຍນັກ ສາມພະ ສາຂາວິຊາວິທະຍາກອນທີ່ກ່ອນທີ່ໄດ້ ວິທະຍາກົດທີ່ກ່ອນທີ່ໄດ້ ມາຮັດກຳນົມຂອງພັກ
34/41

CP411107 Data Structures and Algorithms
ໄຊໂຄສະໜັບນິພຈົນ

Inorder Traversal

```
1 if (root is not null)
  1 if root->token is operand
    1 print(root->token)
    else
      1 print(open parenthesis)
      2 infix(root->leftSubtree)
      3 print(root->token)
      4 infix(root->rightSubtree)
      5 print(close parenthesis)
  2 return
```

ນາຄ.ສາຍນັກ ສາມພະ ສາຂາວິຊາວິທະຍາກອນທີ່ກ່ອນທີ່ໄດ້ ວິທະຍາກົດທີ່ກ່ອນທີ່ໄດ້ ມາຮັດກຳນົມຂອງພັກ
36/41

CP411107 Data Structures and Algorithms
ໄຊໂຄສະໜັບນິພຈົນ

ສຽງ

- ອົງດີປະກອບຂອງທີ່
- ຂັບທີ່ (Subtree)
- ໄຊນາຣີທີ່ (Binary Trees)
- ກາຮແກນທີ່ທີ່ໃນທີ່ກ່ອນທີ່ໄດ້
- ກາຮທົ່ວໂລ່ງເຂົ້າໄປໃນໄຊນາຣີ

ນາຄ.ສາຍນັກ
ມາຮັດກຳນົມຂອງພັກ
39/41

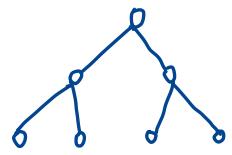
CP411107 Data Structures and Algorithms
ໄຊໂຄສະໜັບນິພຈົນ

ກາຮບ້ານ

- ຈົງເຂັ້ມໂປຣແກຣມສ່ວັງທີ່ ດ້ວຍກາຮໃຈ້ເວັບ 3 ຮະດັບ
- ຈົງເຂັ້ມໂປຣແກຣມສ່ວັງທີ່ ດ້ວຍກາຮໃຈ້ລັດສິບ 3 ຮະດັບ

ມຕນາ

ຊັບພາua tree ດ້ວຍ Array (ຈະຫຼື)



ຂອງນີ້ພາua tree ດ້ວຍ array ເປົ້າໄດ້ເປົ້າໃຫຍ່ໄລ້ກວ່າກຳນົດໄລ້