

CP411107 Data Structures and Algorithms

โครงสร้างข้อมูลแบบลำดับเชิงเส้น

การสร้างคิวด้วย Link List

การสร้างคิวด้วย Link List

หมายถึง การแทนที่ข้อมูลของคิวด้วย Link list ซึ่งเป็นการจัดสรรเนื้อที่หน่วยความจำแบบ Dynamic

A diagram illustrating a conceptual queue. It consists of four rectangular boxes arranged horizontally. The first box is labeled 'plum' above and 'front' below. The second box is labeled 'kiwi' above. The third box is labeled 'grape' above. The fourth box is labeled 'fig' above and 'rear' below.

(a) Conceptual queue

A diagram illustrating a physical queue implemented as a linked list. At the top, there is a control box containing three fields: 'front' (with a red square), 'count' (with the value '4'), and 'rear' (with a red square). Below this, a sequence of four nodes is shown, each containing a fruit name and a red square. The nodes are 'plum', 'kiwi', 'grape', and 'fig'. Arrows connect the red square of one node to the next node. The 'front' pointer points to the 'plum' node, and the 'rear' pointer points to the 'fig' node. The 'plum' node is also labeled 'front' below it, and the 'fig' node is labeled 'rear' below it.

(b) Physical queue

ผศ.ดร.สายัญ สายยศ อาจารย์วิชาวิทยาการคอมพิวเตอร์ ภาควิชาวิทยาการคอมพิวเตอร์ มหาวิทยาลัยขอนแก่น

16/28

| CP411107 Data Structures and Algorithms   | โครงสร้างข้อมูลแบบลำดับเชิงเส้น             |
|---|---|
| <h2>Algorithm พื้นฐานของ Queue</h2>   |   |
| 1. Create queue:  | สร้าง queue head จาก dynamic memory         |
| 2. Enqueue:   | เพิ่มรายการเข้าไปใน queue                   |
| 3. Dequeue:   | ลบรายการออกจาก queue                        |
| 4. Queue front:   | เรียกใช้ข้อมูลที่ด้านหน้าของ queue          |
| 5. Queue rear:  | เรียกใช้ข้อมูลทีด้านหน้าของ queue           |
| 6. Empty queue:   | ตรวจสอบว่า queue ว่างหรือไม่                |
| 7. Full queue:  | ตรวจสอบว่า queue เต็มหรือไม่                |
|   | (มีหน่วยความจำ จัดให้ได้หรือไม่)            |
| 8. Queue count:   | บอกจำนวนรายการใน queue                      |
| 9. Destroy queue:   | ลบข้อมูลทั้งหมดใน queue และคืนหน่วยความจำ   |
|   | ให้ระบบแล้วลบและคืนหน่วยความจำของ head node |
| หส.ดร.สายัญ ตามยศ สาขาวิชาวิทยาการคอมพิวเตอร์ วิทยาลัยการอุดมศึกษาและเทคโนโลยี มหาวิทยาลัยขอนแก่น |   |
| 25/28   |   |

CP411107 Data Structures and Algorithms

โครงสร้างข้อมูลแบบลำดับเชิงมิติ

## Doubly Linked List

- แต่ละโหนดมี Link 2 Link เพื่อชี้โหนดถัดไป และโหนดก่อนหน้า
- การดำเนินการใด ๆ สามารถเดินไปข้างหน้าและย้อนกลับได้

The diagram illustrates a Doubly Linked List. It consists of a series of nodes, each represented as a box divided into two parts: a data field (indicated by a diagonal line) and a 'Link' field. The first node has a 'Head' pointer pointing to its data field. Each node's 'Link' field points to the next node in the sequence. The previous node's 'Link' field points to the current node, allowing for bidirectional traversal. The last node's 'Link' field is null, indicated by a diagonal line.

ผศ.ดร.สายมัญญ์ สายมณี สาขาวิชาบริหารคอมพิวเตอร์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

30/28

```

C:\Users\ADMIN\OneDrive\Desktop\trn\trn\CP411107 Data Structure\1.6
File Edit Format Run Options Window Help
class Node:
    def __init__(self, data=None):
        self.data = data
        self.next = None
        self.prev = None

Front = Node(None)
Rear = Node(None)
for i in range(10):
    P=Node(i)

    Rear.next = P
    P.prev=Rear
    Rear = P

    if Front.next==None:
        Front=P
        Rear=P

Temp=Front
while Temp!=None:
    print(Temp.data)
    Temp = Temp.next

```

CP411107 Data Structures and Algorithms

โครงสร้างข้อมูลแบบทวิทางเชื่อม

## การบ้าน

จงเขียนโปรแกรมโครงสร้างข้อมูลแบบคิวด้วยลิงคีสทิล

ผศ.ดร.सानัญ สาขอยศ สาขาวิชาวิศวกรรมคอมพิวเตอร์ วิทยาเขตสุพรรณบุรี

การบ้านวิชาสาขาคอมพิวเตอร์ 32/28