

บทที่ 4 Array List

บทเรียนย่อย

- 4.1 Array Lists Operations and Concept
- 4.2 Array Lists Component
- 4.3 Array Lists Implementation

วัตถุประสงค์

- นิสิตมีความรู้ และความเข้าใจเกี่ยวกับแนวคิด และองค์ประกอบสำคัญต่าง ๆ ในการ จัดการโครงสร้างข้อมูลในรูปแบบของ Array Lists
- นิสิตสามารถเขียนโปรแกรมเพื่อดำเนินการตามแนวคิดของ Array Lists
- นิสิตสามารถนำแนวคิดของ Array List มาประยุกต์ใช้งานในการพัฒนาโปรแกรม

บทที่ 4 Array List

บทเรียนย่อย

- 4.1 Array Lists Operations and Concept
- 4.2 Array Lists Component
- 4.3 Array Lists Implementation

4.1 Array Lists Operations and Concept

Array Lists คือ โครงสร้างข้อมูลชนิดหนึ่งที่ถูกออกแบบและสร้างขึ้นจากการนำตัวแปรประเภทอาร์เรย์ (Array) มาเพิ่มกระบวนการในการจัดการข้อมูล เพื่ออำนวยความสะดวกและลดข้อจำกัดในการใช้งาน โดยอาศัยหลักการและการดำเนินการของลิสต์ (Lists)

ArrayList

capacity = 10 , count = 7

a[0]	a[1]	a[2]	a[3]	a[4]	a[5]	a[6]	a[7]	a[8]	a[9]
4	21	36	14	62	91	8			

การดำเนินการพื้นฐานของลิสต์ (Lists)

- สามารถเพิ่มข้อมูล
- สามารถลบข้อมูล
- สามารถเปลี่ยนแปลง หรือ แก้ไขข้อมูล
- สามารถค้นหาข้อมูล
- สามารถตรวจสอบจำนวนข้อมูลที่มีทั้งหมด
- สามารถตรวจสอบความจุในการเก็บข้อมูลของลิสต์
- สามารถตรวจสอบพื้นที่ว่างในการเก็บข้อมูล
- สามารถตรวจสอบพื้นที่เต็มในการเก็บข้อมูล

บทที่ 4 Array List

บทเรียนย่อย

4.1 Array Lists Operations and Concept

4.2 Array Lists Component

4.3 Array Lists Implementation

4.2 Array Lists Component

องค์ประกอบของ **Array Lists** จะประกอบด้วย คุณสมบัติ (Property) และกระบวนการทำงาน (method) เนื่องจากการสร้างขึ้นให้อยู่ในรูปแบบของคลาส (class) ซึ่งมีรายละเอียดดังนี้

คุณสมบัติ (Property)	กระบวนการทำงาน (method)
capacity	add
count	get
item	set
arry	remove

4.2 Array Lists Component [2]

คุณสมบัติ (Property)	กระบวนการทำงาน (method)
	indexOf
	size
	max_size
	isEmpty
	isFull
	show

Array Lists Class

ArrayList

- capacity : int
- count : int
- item : int *
- array : int *

- + ArrayList(size : int)
- + ~ArrayList()
- + add(value : int) : void
- + add(index : int , value : int) : void
- + get(index : int) : int
- + set(index : int , value : int) : void
- + remove(value : int) : void
- + indexOf(value : int) : int
- + size() : int
- + max_size() : int
- + isEmpty() : boolean
- + isFull() : boolean
- + show() : void

รายละเอียดคุณสมบัติของ Array Lists Class

คุณสมบัติ (Property)	รายละเอียด
capacity	ตัวแปรสำหรับการใช้ระบุความจุสูงสุดใน การ เก็บข้อมูล
count	ตัวแปรสำหรับการใช้นับจำนวนข้อมูลที่ เก็บไว้ทั้งหมด
item	ตัวแปรสำหรับเก็บที่อยู่ของข้อมูลที่เข้าถึง ล่าสุด
arry	ตัวแปรอาร์เรย์สำหรับเก็บข้อมูล

รายละเอียดกระบวนการทำงานของ Array Lists Class

กระบวนการทำงาน (method)	รายละเอียดการทำงาน
ArrayList(int size)	Constructor สำหรับสร้าง Array โดยระบุขนาดสูงสุด ตามค่าของพารามิเตอร์ที่ส่งเข้ามา
~ArrayList()	Destructor สำหรับลบ Array ที่กำหนดขึ้นออกจากหน่วยความจำ
add(int value)	เพิ่มข้อมูลเข้า Array List ในพื้นที่ว่างตามลำดับ
add(int index , int value)	เพิ่มข้อมูลเข้า Array List ลงในตำแหน่งที่ระบุ

รายละเอียดกระบวนการทำงานของ Array Lists Class [2]

กระบวนการทำงาน (method)	รายละเอียดการทำงาน
<code>int get(int index)</code>	คืนค่าข้อมูลในตำแหน่งที่กำหนด จาก Array List
<code>set(int index , int value)</code>	ปรับเปลี่ยนหรือแก้ไขค่า ในตำแหน่งที่กำหนด
<code>remove(int value)</code>	ลบข้อมูลออกจาก Array List ตามค่าที่กำหนด
<code>int indexOf(int value)</code>	ค้นหาข้อมูลจากค่าที่กำหนด และคืนค่าตำแหน่งที่ค้นหาข้อมูลพบ หากไม่พบจะคืนค่า -1

รายละเอียดกระบวนการทำงานของ Array Lists Class [3]

กระบวนการทำงาน (method)	รายละเอียดการทำงาน
int size()	คืนค่าจำนวนข้อมูลที่มีอยู่ใน Array List
int max_size()	คืนค่าจำนวนสูงสุดที่สามารถจัดเก็บข้อมูลได้ใน Array List
boolean isEmpty()	ตรวจสอบข้อมูลใน Array List โดยถ้าไม่มีข้อมูลจะคืนค่า TRUE หากมีข้อมูลจะคืนค่า FALSE
boolean isFull()	ตรวจสอบข้อมูลใน Array List ว่าเต็มหรือไม่ ถ้าเต็มคืนค่า TRUE หากไม่เต็มคืนค่า FALSE

รายละเอียดกระบวนการทำงานของ Array Lists Class [4]

กระบวนการทำงาน (method)	รายละเอียดการทำงาน
show()	แสดงผลข้อมูลที่มีใน Array List ทั้งหมด ผ่านทางหน้าจอ

บทที่ 4 Array List

บทเรียนย่อย

- 4.1 Array Lists Operations and Concept
- 4.2 Array Lists Component
- 4.3 Array Lists Implementation

4.3 Array Lists Implementation

การสร้างคลาส Array List ภาษา C++

```
class ArrayList {  
    private :  
        int capacity;  
        int count;  
        int * item;  
        int * array;  
    public :  
        ArrayList( int size );  
        ~ArrayList( );  
        ...  
};
```


4.3 Array Lists Implementation [2]

การสร้างคลาส Array List ภาษา C++

```
class ArrayList {  
    ...  
    add( int value );  
    add( int index , int value );  
    int get( int index );  
    set( int index , int value );  
    remove( int value );  
    int indexOf( int value );  
    ...  
};
```

4.3 Array Lists Implementation [3]

การสร้างคลาส Array List ภาษา C++

```
class ArrayList {  
    ...  
    int size( );  
    int max_size( );  
    boolean isEmpty( );  
    boolean isFull( );  
    show( );  
};
```

การดำเนินการใน Constructor และ Destructor

```
ArrayList :: ArrayList( int size ){  
    array = new int[ size ];  
}
```

```
ArrayList :: ~ArrayList( void ){  
    delete [] array;  
}
```

การเรียกใช้งานคลาส Array List

```
int main(void){
```

```
    ArrayList *obj_arrList = new ArrayList(10);
```

```
    obj_arrList->add(5);
```

```
    obj_arrList->add(8);
```

```
    obj_arrList->show();
```

```
    ArrayList obj_arrList(10);
```

```
    obj_arrList.add(5);
```

```
    obj_arrList.add(8);
```

```
    obj_arrList.show();
```

```
}
```

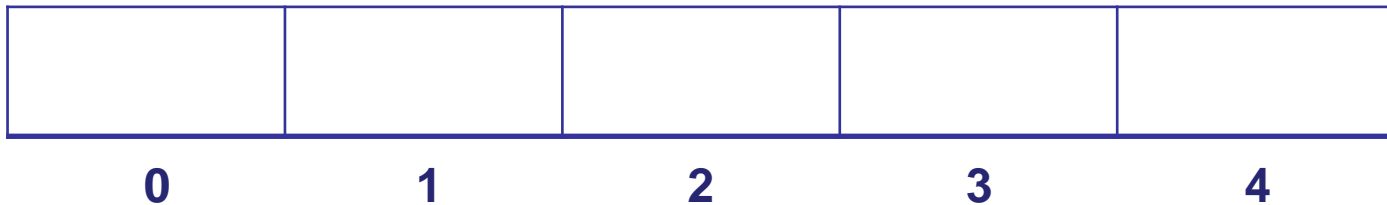
แนวทางการเพิ่มข้อมูล

```
ArrayList obj_arrList( 5 );
```

ArrayList

capacity = 5 , count = 0 , item = null

arry



แนวทางการเพิ่มข้อมูล [2]

```
obj_arrList.add( 79 );
```

ArrayList

capacity = 5 , count = 1 , item = null

arry

79				
0	1	2	3	4

แนวทางการเพิ่มข้อมูล [2]

```
obj_arrList.add( 63 );
```

ArrayList

capacity = 5 , count = 2 , item = null

arry

79	63			
0	1	2	3	4

แนวทางการเพิ่มข้อมูล [3]

```
obj_arrList.add( 81 );
```

ArrayList

capacity = 5 , count = 3 , item = null

arry

79	63	81		
0	1	2	3	4

แนวทางการเรียกข้อมูล

`obj_arrList.get(1);`

ArrayList

capacity = 5 , count = 3 , item = null

arry

79	63	81		
0	1	2	3	4

แนวทางการเรียกข้อมูล [2]

obj_arrList.get(1);

ArrayList

capacity = 5 , count = 3 , item = null

arry

79	63	81		
0	1	2	3	4

แนวทางการเรียกข้อมูล [3]

`obj_arrList.get(1);`

ArrayList

capacity = 5 , count = 3 , item = null

arry

79	63	81		
0	1	2	3	4

แนวความคิดแก้ไขข้อมูล

```
obj_arrList.set( 1 , 34 );
```

ArrayList

capacity = 5 , count = 3 , item = null

array

79	63	81		
0	1	2	3	4

แนวทางการแก้ไขข้อมูล [2]

obj_arrList.set(1 , 34);

ArrayList

capacity = 5 , count = 3 , item = null

array

79	63	81		
0	1	2	3	4

แนวความคิดแก้ไขข้อมูล [3]

`obj_arrList.set(1 , 34);`

ArrayList

capacity = 5 , count = 3 , item = null

arry

79	34	81		
0	1	2	3	4

แนวทางการค้นหาข้อมูล

`obj_arrList.indexOf(81);`

ArrayList

`capacity = 5 , count = 3 , item = null`

arry

79	63	81		
0	1	2	3	4

แนวทางการค้นหาข้อมูล [2]

`obj_arrList.indexOf(81);`

ArrayList

capacity = 5 , count = 3 , item = null

arry

79	63	81		
0	1	2	3	4

แนวทางการค้นหาข้อมูล [3]

`obj_arrList.indexOf(81);`

ArrayList

capacity = 5 , count = 3 , item = null

arry

79	63	81		
0	1	2	3	4

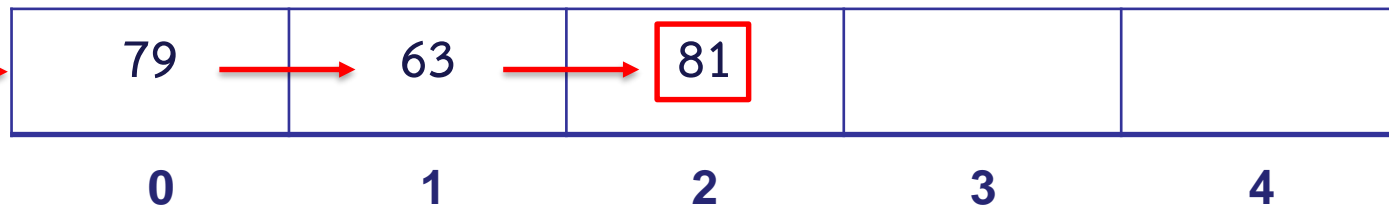
แนวทางการค้นหาข้อมูล [4]

`obj_arrList.indexOf(81);`

ArrayList

capacity = 5 , count = 3 , item = null

array



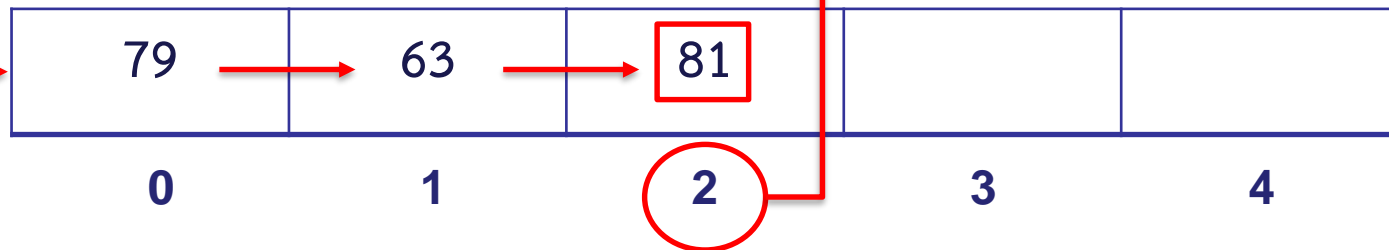
แนวทางการค้นหาข้อมูล [5]

`obj_arrList.indexOf(81);`

ArrayList

`capacity = 5 , count = 3 , item = &array[2]`

array



แนวความคิดการเพิ่มข้อมูลแบบระบุตำแหน่ง

```
obj_arrList.add( 2 , 24 );
```

ArrayList

capacity = 5 , count = 3 , item = null

arry

79	63	81		
0	1	2	3	4

แนวความคิดการเพิ่มข้อมูลแบบระบุตำแหน่ง [2]

`obj_arrList.add(2 , 24);`

ArrayList

capacity = 5 , count = 3 , item = null

array

79	63		81	
0	1	2	3	4

แนวความคิดการเพิ่มข้อมูลแบบระบุตำแหน่ง [3]

```
obj_arrList.add( 2 , 24 );
```

ArrayList

capacity = 5 , count = 4 , item = null

array

79	63	24	81	
0	1	2	3	4

แบบฝึกหัด

1. สร้าง คลาส ArrayList เพื่อจัดเก็บข้อมูลเลขจำนวนเต็ม โดยมี
ความสามารถในการจัดการข้อมูล ดังนี้

- สามารถเพิ่มข้อมูลแบบปกติและแบบระบุตำแหน่ง
- สามารถเปลี่ยนแปลง หรือ แก้ไขข้อมูล
- สามารถค้นหาข้อมูล
- สามารถตรวจสอบจำนวนข้อมูลที่มีทั้งหมด
- สามารถตรวจสอบความจุในการเก็บข้อมูลของลิสต์
- สามารถตรวจสอบพื้นที่ว่างในการเก็บข้อมูล
- สามารถตรวจสอบพื้นที่เต็มในการเก็บข้อมูล

2. นำคลาสที่สร้างขึ้นไปทดสอบการใช้งานในฟังก์ชัน main โดยทำการ
ทดลองทุกความสามารถที่มีในคลาส ArrayList ของตนเอง

Class ของแบบฝึกหัด

ArrayList
<ul style="list-style-type: none">- capacity : int- count : int- item : int *- arry : int *
<ul style="list-style-type: none">+ ArrayList(size : int)+ ~ArrayList()+ add(value : int) : void+ add(index : int , value : int) : void+ get(index : int) : int+ set(index : int , value : int) : void+ indexOf(value : int) : int+ size() : int+ max_size() : int+ isEmpty() : boolean+ isFull() : boolean+ show() : void