

序言

前言

程序 = 算法 + 数据结构

为什么要学习数据结构？

软件方面：数据处理以及数据处理速度。

硬件方面：实时数据监控

什么是数据结构（Data Structure）？

数据（Data Element）：所有能够输入到计算机中并且能识别的符号集合的统称。

数据元素（Data item）：表示事物的一组数据。

```
1 {  
2   id（数据）（主关键字）:1（数据）  
3   name（关键字）:"jack",  
4   age:21  
5 }
```

数据是数据元素的最小识别单位（不可分割）。

能够识别该元素的一个或多个数据项叫做关键字，能唯一识别数据元素的关键字称作主关键字。

数据结构就是有特定关系的数据元素的有限集合。

数据结构的概念？

包含三个方面：数据的逻辑结构，数据的储存结构，对数据的操作。

逻辑结构

数据元素之间的逻辑关系

线性结构

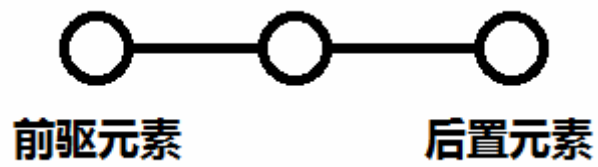
模拟

```
1 [{  
2   name:"jack",  
3   age:6  
4 }, {  
5   name:"bobo",  
6   age:7  
7 }]
```

关系

线性

图解



树状结构

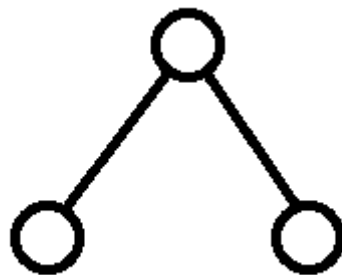
模拟

```
1  [{
2    name:"jack",
3    age:6,
4    pet:{
5      dog:"koko",
6      cat:{
7        name:"coco",
8        age:7
9      }
10   }
11  }]
```

关系

结构

图解



图结构

模拟

```
1  | 没用过
```

图解



数据的储存结构？

数据元素以及关系在计算机中的储存成为数据的**储存结构**，也成**物理结构**

逻辑结构和**储存结构**是毫不相干的，**储存结构**是**逻辑结构**在计算机内存中的实现。

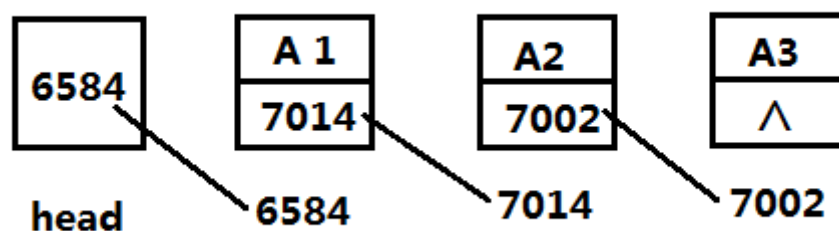
顺序储存结构

顺序存储结构：使用一组连续的内存单元依次存放数据元素，数据元素在内存中的物理存储顺序和他们的逻辑次序相同。

序号	存储元素	存储地址
1	a1	a
2	a2	a+k
	...	
	...	
i	ai	$a+(i-1)*k$
	...	
n	an	$a+(n-1)*k$
	...	

链式存储结构

链式存储结构：使用若干地址分散的存储单元存储数据元素。



1. 当然我们需要一个地址域记住头元素的地址。
2. 不开辟连续的存储空间（可连可不连）
3. 逻辑的相连，地址不一定相连。
4. 数据域和地址域组成一个节点表示一个数据元素。

数据类型和组成？

标准数据类型（固有数据类型）：int float double

自定义数据类型：抽象数据类型（ADT: Abstract Data Type）