向量

抽象数据类型：从数组到向量

θ2-A2

秩秩斯干，幽幽南山。

贵贱长少，秩之焉，莫不从祖公而贵敬之，是天下之大节也。

邓俊辉

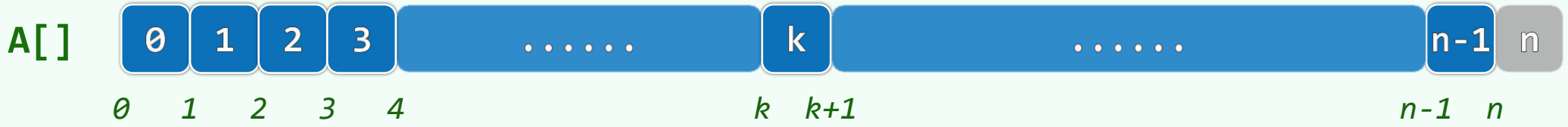
deng@tsinghua.edu.cn

到了，就是这里。妈妈，你只管找号头，311，就是爸爸的号

数组 ~ 循秩访问

❖ C/C++语言中，数组元素与编号一一对应：

A[0], A[1], A[2], ..., A[n-1]



❖ 反之，元素各由**编号**唯一指代，并可**直接**访问

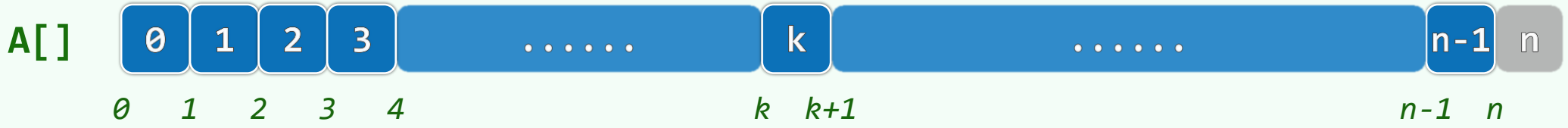
故亦称作线性数组（linear array）

❖ 若每个元素占用的空间量为s（已计入padding），则A[i]的**物理地址** = A + i×s

数组 ~ 向量

- 向量是数组的抽象与泛化，由一组元素按线性次序封装而成

  各元素与[0, n)内的秩（rank）一一对应： `using Rank = unsigned int; //call-by-rank`



- 操作、管理维护更加简化、统一与安全
- 元素类型可灵活选取，便于定制复杂数据结构：

```
using PFCTree = BinTree<char>;   //PFC树
using PFCForest = Vector<PFCTree*>;   //PFC森林
```

Vector ADT

操作	功能	适用对象
size() / empty()	报告元素总数 / 判定是否为空	向量
get(r) / put(r, e)	获取秩为r的元素 / 用e替换秩为r元素的数值	向量
insert(r, e) / insert(e)	将e作为秩为r的 / 最后一个元素插入	向量
remove(lo, hi) / remove(r)	删除秩为r / 区间内的元素	向量
disordered() / sort(lo, hi) / unsort(lo, hi)	检测是否整体有序 / 整体排序 / 整体置乱	向量
find(e , lo, hi) / search(e, lo, hi)	在指定区间内查找目标e	向量 / 有序向量
dedup() / uniquify()	剔除相等的元素	向量 / 有序向量
traverse(visit())	遍历向量，统一按visit()处理所有元素	向量

ADT操作实例

操作	输出	向量组成（自左向右）
初始化		
insert(0, 9)		9
insert(0, 4)		4 9
insert(1, 5)		4 5 9
put(1, 2)		4 2 9
get(2)	9	4 2 9
insert(3, 6)		4 2 9 6
insert(1, 7)		4 7 2 9 6
remove(2)	2	4 7 9 6
insert(1, 3)		4 3 7 9 6
insert(3, 4)		4 3 7 4 9 6
size()	6	4 3 7 4 9 6

操作	输出	向量组成（自左向右）
disordered()	3	4 3 7 4 9 6
find(9)	4	4 3 7 4 9 6
find(5)	-1	4 3 7 4 9 6
sort()		3 4 4 6 7 9
disordered()	0	3 4 4 6 7 9
search(1)	-1	3 4 4 6 7 9
search(4)	2	3 4 4 6 7 9
search(8)	4	3 4 4 6 7 9
search(9)	5	3 4 4 6 7 9
search(10)	5	3 4 4 6 7 9
uniquify()		3 4 6 7 9
search(9)	4	3 4 6 7 9

STL Vector

```
#include <iostream>

#include <vector>

using namespace std;

vector<int> v;  //an empty vector of integers

vector<int> s( 289, 7 );            //{ 7, 7, 7, 7, 7, 7, 7,      7, 7, ..., 7 }

s.insert( s.begin() + 7, 2024 );    //{ 7, 7, 7, 7, 7, 7, 7, 2024, 7, 7, ..., 7 }

s.erase( s.end() - 280, s.end() ); //{ 7, 7, 7, 7, 7, 7, 7, 2024, 7, 7, }

for ( int i = 0; i < s.size(); i++ )

   cout << s[i] << endl;
```