# 数据结构与算法（基础数据结构-数组）

给你一个文件里面包含全国人民（14亿）的年龄数据（0~180），现在要你统计每一个年龄有多少人？

给定机器为 单台+2CPU+2G内存。不得使用现成的容器，比如map等。

数组算法

在以上情况下你该如何以最高效的方法来解决这个问题？

排序算法：1 1 1 2 2 2 3 3 3 4 4 5。想过没？能不能解决这个问题？：不能 为什么？

排序的最高效算法：O(nlogn) 14亿，排不出来，而且内存也不够。

int a[] = new int[180];a[0]++;0表示的是0岁，a[0]的值表示的就是0有多少人

12：56

23：56111

52：9999888

下标：数组最优一个特点。这里可以通下标表示成有意义的数据，不只是数据里面的标记，年龄和下标对应。随机访问：可以直接通过下标定位到数组中的某一个数据

package algorithm.array;

import java.io.BufferedReader;

import java.io.FileInputStream;

import java.io.FileNotFoundException;

import java.io.InputStreamReader;

import java.io.UnsupportedEncodingException;

import javax.imageio.stream.FileImageInputStream;

public class AgeStas {

public static void main(String[] args) throws Exception {

String str = null;

String fileName = "E:\\课件\\算法分析与设计\\VIP\\基础数据结构\\数组\\age1.txt";

InputStreamReader isr = new InputStreamReader(new FileInputStream(fileName),"UTF-8");

long start = System.currentTimeMillis();

BufferedReader br = new BufferedReader(isr);

int tot = 0 ; //21亿

int data [] = new int[200];

while((str = br.readLine()) != null){ //一行一行的读 O(n)

int age = Integer.valueOf(str);

data[age] ++ ;

tot ++ ;

}

//O(n) 14亿. 100万/秒 \*1000 = 10亿 100~1000s之间 => 500s以下 60\*8=480s

System.out.println("总共的数据大小: " + tot);

for(int i = 0 ; i < 200 ; i ++){//下标从0开始的

System.out.println(i + ":" + data[i]);

}

//144239ms => 144s

System.out.println("计算花费的时间为:" + (System.currentTimeMillis() - start) + "ms");

}

}

为什么很多计算机编程语言中数组的下标要从0开始呢

定义一个数组一定会分配内存空间。数组的特点是 内存是一段连续的地址。

int a[] = new int[3];

到内存中申请空间：10001,10002,10003

存数据

a[0]=>10001=====>10001+0\*typesize

a[1]=>10002=====>10001+1\*typesize

a[2]=>10003=====>10001+2\*typesize

如果我们不从0开始

a[1] = 10001+(1-1)

a[2] = 10001+(2-1)

a[3] = 10001+(3-1)

如果下标从0开始，计算元素在内存中的位置时，直接用数组起始位置+index×objectSize就能获得元素的位置，index不用-1

1.数组的定义

所谓数组，是有序的元素序列。若将有限个类型相同的变量的集合命名，那么这个名称为数组名。组成数组的各个变量称为数组的分量，也称为数组的元素，有时也称为下标变量。用于区分数组的各个元素的数字编号称为下标。数组是在程序设计中，为了处理方便，把具有相同类型的若干元素按无序的形式组织起来的一种形式。这些无序排列的同类数据元素的集合称为数组。int的数组就不能存float也不能存double

数组是用于储存多个相同类型数据的集合。通常用Array表示，也称之为线性表，画图演示

2.特点

(1)数组是相同数据类型的元素的集合。

(2)数组中的各元素的存储是有先后顺序的，它们在内存中按照这个先后顺序连续存放在一起。内存地址

(3)数组元素用整个数组的名字和它自己在数组中的顺序位置来表示。例如，a[0]表示名字为a的数组中的第一个元素，a[1]代表数组a的第二个元素，以此类推。

3.表现形式

（1）一维数组

Int a[],String a[]

（2）多维数组

Int a[][],int a[][][]。 int a[m][n]:内存空间是多少？ m\*n a[0][10]: 链表解决，a[0]:->10>2 a[1]->15

4.随机访问：

数组是连续的内存空间和相同类型的数据。正是因为这两个限制，它才有了一个非常重要的特性：随机访问。但有利就有弊，这两个限制也让数组的很多操作变得非常低效，比如要想在数组中删除、插入一个数据，为了保证连续性，就需要做大量的数据搬移工作。

随机访问的重要应用：查找，面试重点

5.数组的缺点：插入和删除

实现代码：

设数组的长度为n，现在，如果我们需要将一个数据插入到数组中的第k个位置。删除第N个位置的数据

6.使用数组一定要注意访问越界问题。

ArrayList和数组：

本质是一样的，都是数组。ArrayList是JDK封装了。不需要管扩容等操作

数组的话就要你全部操作

两者之间应该如何选用？：

不知道数据大小的肯定选ArrayList。

如果你知道数据的大小而且你又非常关注性能那就用数组。

数组最需要注意的就是越界：所以一定要多加判断，尤其是在开始和结束。测试的时候也一样注意头和尾。

Java里面的内存分为几种？

Java分为堆栈两种内存。

什么是堆内存？：存放new创建的对象和数组

什么是栈内存？引用变量

堆栈都用Java用来存放数据的地方，与C++ / c不一样。java自动管理我们的堆栈。gc，new出来的你没管过。

堆栈的区别：

1.栈的速度要快

2.栈内存的数据可以共享，主要存一些基本数据类型。

int a = 3; //在栈中创建变量a 然后给a赋值，先不会创建一个3而是先在栈中找有没有3，如果有直接指向。如果没有就加一个3进来。

int b =3; //首先也要创建一个变量b，

1.面试经典：

String str1 = "abc"; String str2 = "abc"; System.out.println(str1==str2); //true

String str1 = "abc"; String str2 = "abc"; str1 = "bcd";

System.out.println(str1 + "," + str2); //bcd,abc

System.out.println(str1==str2); //false 虽然最开始 str1和str2都指向同一个变量abc但str1引用变化后不会改变str2的

String str1 = "abc";

String str2 = "abc";

str1 = "bcd";

String str3 = str1;

System.out.println(str3);  //bcd

String str4 = "bcd";

System.out.println(str1 == str4); //true

String str1 = new String("abc");

String str2 = "abc";

System.out.println(str1==str2); //false new在堆内存中新开了一个对象

String s1 = "ja";

String s2 = "va";

String s3 = "java";

String s4 = s1 + s2; //java 注意这个+号，java里面重载了+，其实调用了stringBuild，会new对象。

System.out.println(s3 == s4); //false

System.out.println(s3.equals(s4)); //true 只是比较值

数组是一个最基础最简单的数据结构，必须要完全搞懂。它是存储相同类型的一组数据，最大的两个特点就是

下标和随机访问。缺点就是插入和删除是很慢的，时间复杂度为O（n）。