




多维数组和矩阵----子数组最大累加和

题目：

给定一个数组arr，返回子数组的最大累加和。例如arr = {1,-2,3,5,-2,6,-1};所有的子数组中[3,5,-2,6]可以累加出最大的和12，所以返回12。

代码：



```
1 import java.util.Arrays;
2 import java.util.Date;
3
4 /**
5  * 求和最大的连续子数组，有可能不唯一，返回一个即可
6  */
7 public class MaxSubArray {
8
9     // 暴力破解法 O(N²)
10    static void findByForce(int []arr){
11        int maxSum = arr[0];
12        for (int j = 0; j < arr.length; j++) {
13            int sum = arr[j]; // 某个元素为子数组的第一个元素
14            int maxOfJ = sum;
15
16            for (int i = j+1; i < arr.length; i++) {
17                sum += arr[i]; // 累加后续元素
18                if (sum>maxOfJ) {
19                    maxOfJ = sum;
20                }
21            }
22            if (maxOfJ>maxSum) {
23                maxSum = maxOfJ;
24            }
25        }
26        System.out.println(maxSum);
27    }
28
29    // 递推法 O(N)
30    static int findByDp(int []arr){
31        if (arr.length==0) {
32            return 0;
33        }
34        int sumJ = arr[0]; // 前J个元素的最大贡献
35        int max = sumJ;
36        // int left = 0,right = 0; // 最大子数组的左右边界下标
37        for (int j = 1; j < arr.length; j++) {
38            if (sumJ >= 0) { // 左子表的最大和为正，继续向后累加
39                sumJ += arr[j];
40            } else {
41                sumJ = arr[j];
42                // left = j; // 丢弃前部分和的同时,更新left
43            }
44        }
45        return max;
46    }
47 }
```

公告

建立了一个交流群，欢迎感兴趣的朋友们加入。
(备注博客园即可)
博客园总阅读量：652997。

群号：228372496



扫一扫二维码，加入群聊。

昵称： |旧市拾荒|
园龄： 6年10个月
粉丝： 176
关注： 2
[关注成功](#)

积分与排名

积分 - 360934

排名 - 2252

随笔分类 (399)

【1】 Hadoop(11)

【2】 Linux(2)

【3】 Python基础(16)

【4】 Python爬虫(9)

【5】 TensorFlow(13)

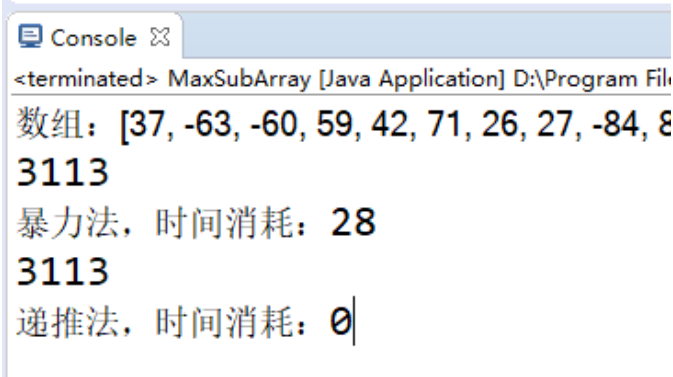
【6】 Spark(25)

【7】 错误解决(19)

```
43         }
44
45         if (sumJ > max) {
46             max = sumJ;
47 //             right = j; // 更新max的同时更新right
48         }
49     }
50     // System.out.println(max+",left="+left+",right:"+right);
51     return max;
52 }
53 /**
54  * 获取指定范围指定个数的随机数组成的数组
55  */
56 public static int[] getRandomArr(int length, int min, int max) {
57     int[] arr = new int[length];
58     for (int i = 0; i < length; i++) {
59         arr[i] = (int) (Math.random() * (max + 1 - min) + min);
60     }
61     return arr;
62 }
63 public static void main(String[] args) {
64 //     int []arr = new int[]{1,-2,3,5,-2,6,-1};
65     int[] arr = getRandomArr(10000, -100, 100);
66     System.out.println("数组： " + Arrays.toString(arr));
67 //     long now = new Date().getTime();
68 //     findByForce(arr);
69 //     long next1 = new Date().getTime();
70 //     System.out.println("暴力法，时间消耗： " + (next1 - now));
71
72     long now = System.currentTimeMillis();
73     findByForce(arr);
74     long next1 = System.currentTimeMillis();
75     System.out.println("暴力法，时间消耗： " + (next1 - now));
76
77     System.out.println(findByDp(arr));
78     long next2 = System.currentTimeMillis();
79     System.out.println("递推法，时间消耗： " + (next2 - next1));
80 }
81
82 }
```



结果：



注意：上面的代码没有考虑数组全为负数的情况，如果数组全为负数的话，返回数组的最大值即可。

如果您觉得阅读本文对您有帮助，请点一下“**推荐**”按钮，您的“**推荐**”将是我最大的写作动力！欢迎各位转载，但是未经作者本人同意，转载文章之后**必须在文章页面明显位置给出作者和原文连接**，否则保留追究法律责任的权利。

本文来自博客园，作者：[|旧市拾荒|](#)，转载请注明原文链接：<https://www.cnblogs.com/xiaoyh/p/10295107.html>

分类：**【9】 算法基础**

好文要顶

关注成功

收藏该文

|旧市拾荒|
粉丝 - 176 关注 - 2

关注成功

2

推荐

0

反对

支持成功 撤回

【8】 数据结构(38)

【9】 算法基础(85)

【A】 机器学习(19)

【B】 数学基础(6)

【C】 其他杂记(13)

【D】 算法训练(3)

【E】 Java开发(50)

【F】 Java并发(22)

【G】 ElasticSearch(36)

【H】 设计模式(32)

阅读排行榜

1. Python math库常用函数(72889)
2. 错误：PermissionError: [WinError 32] 另一个程序正在使用此文件，进程无法访问。"+文件路径"的解决方案(31967)
3. python jieba库的基本使用(26925)
4. 微调(Fine-tune)原理(20931)
5. Gephi安装过程中出现错误：can't find java 1.8 or higher(20587)

评论排行榜

1. Gephi安装过程中出现错误：can't find java 1.8 or higher(6)
2. pip 安装模块时出现error: Microsoft Visual C++ 14.0 is required. Get it with "Microsoft Visual C++ Build Tools":(6)
3. 不忘初心、方得始终—2019总结暨2020展望(4)
4. 最长连续递增子序列（部分有序）(4)
5. 机器学习项目实战----泰坦尼克号获救预测(一)(3)

推荐排行榜

« 上一篇: 多维数组和矩阵----边界为1的最大子方阵

» 下一篇: [多维数组和矩阵---子矩阵最大累加和](#)

posted @ 2019-01-20 15:40 | 旧市拾荒 | 阅读(332) 评论(0) 编辑 收藏 举报

[刷新评论](#) [刷新页面](#) [返回顶部](#)

 发表评论 [升级成为园子VIP会员](#)

[提交评论](#) [退出](#) [订阅评论](#) [我的博客](#)

[Ctrl+Enter快捷键提交]

- 【推荐】通义灵码，灵动指间，快码加编，你的智能编码助手
- 【推荐】编程路上的催化剂：大道至简，给所有人看的编程书
- 【推荐】阿里云云市场联合博客园推出开发者商店，欢迎关注
- 【推荐】阿里云暖冬特惠，2核2G轻量应用服务器首购61元/年



编辑推荐：

- C# AsyncLocal 是如何实现 Thread 间传值
- 一个例子形象地理解同步与异步
- C# 线程本地存储 为什么线程间值不一样
- [动画进阶] 神奇的 3D 卡片反光闪烁动效
- 记一次缓存失效引发的惨案!

阅读排行:

- C#开源免费的开发效率提升利器：DevToys开发人员的瑞士军刀！
- 全球 IPv4 耗尽，下个月开始收费！
- WebAssembly入门笔记[1]：与JavaScript的交互
- C# AsyncLocal 是如何实现 Thread 间传值
- 【原创】为什么Linux不是实时操作系统

1. 数学基础系列(三)---第一中值定理、微积分基本定理、牛莱公式、泰勒公式(9)

2. Java开发学习(一)---初识Spring及其核心概念(7)

3. 微调(Fine-tune)原理(7)

4. Java开发学习(二十九)---Maven依赖传递、可选依赖、排除依赖解析(6)

5. 数学基础系列(一)---函数、极限、连续性、导数(6)