

Java实现简单的递归操作

原创

Alex_the_Dawn

2017-04-03 11:22:48

86803

★ 收藏 39

版权

分类专栏: 算法-java

文章标签: java

递归

递归算法

在数据结构算法设计中, 或者一个方法的具体实现的时候, 有一种方法叫做“递归”, 这种方法在思想上并不是特别难, 但是实现起来还是有一些需要注意的。虽然对于很多递归算法都可以由相应的循环迭代来代替, 但是对于一些比较抽象复杂的算法不用递归很难理解与实现。

递归分为直接递归和间接递归, 就简单分享一下两个小的直接递归。

对于递归的概念, 其实你可以简单的理解为自己定义自己, 记得小时候看过一部电视剧《狼毒花》, 里面主角叫做“常发”, 但是个文盲, 老师问他叫什么, 他说“常发”。“哪个常?”“常发的常啊!”“哪个发?”“常发的发啊!”结果第二节课老师就让一群小朋友一起喊“常发的常, 常发的发, 傻瓜的傻, 傻瓜的瓜”。言归正传, 显然在多数情况下递归是解释一个想法或者定义的一种合理方法。在思想上递归类似于数学中曾经学过的数学归纳法。

递归的实现:

递归的实现要注意有两点: 一个递归的选项和一个非递归的选项, 后者成为**基础情形** (base case)。基础情形是递归的终结情形, 没有基础情形或者处理不好都会导致无穷递归, 这是我们不想要的结果。**递归实现起来最关键的是处理好基础情形。**结合具体事例在说一下递归**回溯**的过程。

下边来写两个小程序:

1、爬楼梯算法: 已知一个楼梯有n个台阶, 每次可以选择迈上一个或者两个台阶, 求走完一共有多少种不同的走法。

方法如下:

```
public int climbStairs(int n) {  
    int i=1;  
    if(n<=0)  
        return 0;  
    if(n==1){  
        return i;  
    }  
    if(n==2){  
        i++;  
        return i;  
    }  
    else  
        return climbStairs(n-1)+climbStairs(n-2);  
}
```

递归函数有返回值的比没有返回值的麻烦一点, 因为一个函数只有一个返回值, 但是递归还要求有基础情形的存在, 所以还必须有if判断来终止递归。所以在每一个if或者else后边都有一个return, 这样保证函数在任何一种情况下都有且仅有一个返回值。

分析一下这个算法:

A: 如果有0个台阶, 那么有0种走法, 这个不用多说;

B: 如果有1个台阶, 那么有1种走法;

C: 如果有2个台阶, 那么有2种走法 (一次走1个, 走两次; 一次走两个);

以上的B和C就是基础情形。

D: 接下来就是递归了, 如果台阶数目多于2个, 那么首先第一步就有两种选择: 第一次走1个, 或者第一次走两个。这样除了第一次后边的走法就有了两种情形:

climbStairs(n-1)和climbStairs(n-2)。这样一直递归下去, 直到出现到了基础情形

(即n=1或n=2的情形), 递归到这个地方 (基础

点赞Mark关注该博主, 随时了解TA的最新博

👍 点赞29

💬 评论6

🔗 分享

★ 收藏39

📱 手机看

💰 打赏

...

关注

—

说的和递归密切相关的“回溯”了。回溯，顾名思义就是从结果倒着回去，找到整个过程，进而分析这个路径或者说是实现的过程。

需要注意的是，这个算法实现思路简单，但是复杂度并没有降低，还牵扯回溯保存堆栈问题(其实递归的设计尽量避免这种嵌套两个的递归方式 (climb (n) 中包含 climb (n-1) 和 climb (n-2))，这种操作会使得堆栈开辟空间随着n的增大以指数型增长，最终程序很容易崩溃)，而且在台阶数目多到一定数量的时候会越界(走法次数会超出int的范围)，所以递归程序很大程度上就是思想实现设计上简单理解一些。

下边是源代码：

```
1 package leetcode;
2
3 public class ClimbStairs {
4     // *****
5     public int climbStairs(int n) {
6         int i=1;
7         if(n<=0)
8             return 0;
9         if(n==1){
10             return i;
11         }
12         if(n==2){
13             i++;
14             return i;
15         }
16         else
17             return climbStairs(n-1)+climbStairs(n-2);
18     }
19     // *****
20     public static void main(String []args){
21         ClimbStairs cs=new ClimbStairs();
22         int a =cs.climbStairs(4);
23         System.out.println(a);
24     }
25
26 }
```

然后还有几个比较典型的递归问题：比如说迷宫问题，或者最经典的汉诺塔问题，下边都给出源码，大家一块儿学习一下。

汉诺塔问题：一次只能移动一个盘子；不能把大盘子放在小盘子上；除去盘子在两个柱子之间移动的瞬间，盘子必须都在柱子上。（在这三点要求下把盘子从起始柱子A全部移动到目标柱子C上）



代码如下：

基础情形：n==1的时候终止递归，进行回溯。

```
1 public class HanNuoTower {
2     public void tower(int n,char s,char m,char e)//n个塔从s经过m最终全部移动到e
3     {
4         if(n==1)
5             move(s,e);
6         else
7         {
8             tower(n-1,s,e,m);
```

点赞Mark关注该博主, 随时了解TA的最新博

点赞29

评论6

分享

收藏39

手机看

打赏

...

关注

一

```

9         move(s,e);
10        tower(n-1,m,s,e);
11    }
12 }
13 public void move(char s,char e){
14     System.out.println("move "+s+" to "+e);
15 }
16 public static void main(String []args){
17     HanNuoTower hnt =new HanNuoTower();
18     hnt.tower(4,'A','B','C');
19 }
20
21 }

```

迷宫走法：二维数组构成一个迷宫，1表示通路，0表示不通，找到一条路径从起始点（traverse函数的参数）到终点（右下角点）。

基础情形：row=grid.length-1&&column=grid[0].length-1时done=true;

```

1
2 public class Maze {
3     private final int TRIED=3;
4     private final int PATH=7;
5
6     private int [][] grid={ {1,1,1,0,0,1,0,1,0,0},
7                             {0,0,1,1,1,0,0,0,0,0},
8                             {1,0,1,0,0,0,1,1,1,1},
9                             {1,1,1,1,1,0,0,0,1,1},
10                            {0,0,0,0,1,1,1,0,0,0},
11                            {1,0,1,0,1,0,0,1,0,0},
12                            {1,0,0,1,1,1,1,1,1,1}    };
13     public boolean traverse(int row,int column){
14         boolean done =false;
15         if(valid(row,column))
16         {
17             grid[row][column]=TRIED;
18             if(row==grid.length-1&&column==grid[0].length-1)
19                 done=true;
20             else
21             {
22                 done=traverse(row+1,column);//down
23                 if(!done)
24                     done=traverse(row,column+1);//right
25                 if(!done)
26                     done=traverse(row-1,column);//up
27                 if(!done)
28                     done=traverse(row,column-1);//left
29             }
30             if(done)
31                 grid[row][column]=PATH;
32         }
33         return done;
34     }
35     private boolean valid(int row,int column){
36         boolean result=false;
37         if(row>=0&&row<grid.length&&column>=0&&column<grid[0].length)
38             if(grid[row][column]==1)
39                 result=true;
40         return result;
41     }
42     public String toString(){
43         String result="\n";
44         for (int row=0;row<grid.length;row++){
45             for(int column=0;column<grid[0].length;column++){
46                 result +=grid[row][column]+
47

```

点赞Mark关注该博主，随时了解TA的最新博

点赞29

评论6

分享

收藏39

手机看

打赏

...

关注

一

```
48         result+="\n";
49     }
50     return result;
51 }
52 public static void main (String []args){
53     Maze maze=new Maze();
54     System.out.println(maze);
55     if(maze.traverse(0, 0))
56         System.out.println("The maze was successfully travelled!");
57     else
58         System.out.println("There is no possible path.");
59     System.out.println(maze);
60 }
61
62 }
63
```

还有一个九连环的操作，有兴趣的话可以一起看看。[Java递归解决九连环问题](#)
如有不得当之处，还望诸位大神指教！

0元
买单

限时 0 元买单，程序员必备 10 大热门课
课程一：《3小时快速学习计算机基础》、课程二：《精华Python小课 3天零基础入门》...

10-27

Java中关于递归的一个简单的例子 晨暮的博客 1307
通过递归的方法在控制台打印出文件名及子目录下的文件import java.io.File; public class Demo2 { public s...

- 优质评论可以帮助作者获得更高权重
- 评论
- 别说我太单纯: 算发真是太高级了 6天前 回复

别说我太单纯: 收藏了.2020.10.21 6天前 回复

别说我太单纯: 递归实现起来最关键的是处理好基础情形 6天前 回复

老衲正在改BUG: 感谢作者 支持原创 简单易懂 适合初学者 2年前 回复

不再犹豫531: 感谢作者 支持原创 简单易懂 适合初学者 2年前 回复

码哥 Alex_the_Dawn 博主 回复: 都是新手，互相学习！ 2年前 回复

java递归查询分类及分类下所有子分类 Sang 1万+
该案例是实际开发中运用，java递归查询分类及分类下所有子分类。代码走起：1.jsp页面布局样式这里不...

java 递归详解 牧小农 1万+
刚学java的时候通常难以理解递归程序设计的概念。递归思想之所以困难，原因在于它非常像是循环推理...

Java实现简单的递归操作_leon_is_sad的博客-CSDN博客 10-3
Java实现简单的递归操作 首先我们应该了解递归的两大要素:终止条件和递归公式 以下便是一个递归的例...

Java实现简单的递归操作_Java那点事 | 学习•思考•分..._CSDN博客 9-4
在数据结构算法设计中,或者一个方法的具体实现的时候,有一种方法叫做“递归”,这种方法在思想上并不是...

java递归简单实例 mtlmzb的专栏 1万+
程序调用自身的编程技巧称为递归（recursion）。递归做为一种算法在程序设计语言中广泛应用。一个...

java递归实现树（Tree） 12-25
一个简单的小例子递归实现list按照index排序的树

java递归的简单实现方式_天堂地址不详-CSDN博客 10-23
递归在平常的开发过程中使用的不是太多,但是作为一个经典算法,我们必须也要了解一下,递归在使用的过...

Java实现简单的递归操作 - Java那点事 | 学习•思考•分
虽然对于很多递归算法都可以由相应的循环迭代来代替,但是对