**难度: 简单**

给定一个由 整数 组成的 非空 数组所表示的非负整数，在该数的基础上加一。

最高位数字存放在数组的首位， 数组中每个元素只存储单个数字。

你可以假设除了整数 0 之外，这个整数不会以零开头。

示例 1：

输入：digits = [1,2,3]

输出：[1,2,4]

解释：输入数组表示数字 123。

示例 2：

输入：digits = [4,3,2,1]

输出：[4,3,2,2]

解释：输入数组表示数字 4321。

示例 3：

输入：digits = [0]

输出：[1]

提示：

1 <= digits.length <= 100

0 <= digits[i] <= 9

来源：力扣（LeetCode）

链接：https://leetcode-cn.com/problems/plus-one

著作权归领扣网络所有。商业转载请联系官方授权，非商业转载请注明出处。

**思路:**

1). 数组的最后一个给它加一,

2). 如果加一之后,不是10,那么直接返回

3). 如果是的话,直接在原有的数组基础上+1,然后把第一个置为1,其它的为0即可

|  |
| --- |
| public int[] plusOne(int[] digits) {  for (int i = digits.length -1; i >= 0; i--) {  digits[i]++;  digits[i] = digits[i] % 10;  if (digits[i] != 0) {  return digits;  }  }  digits = new int[digits.length + 1];  digits[0] = 1;  return digits;  } |