

C趣味程序百例(19)1~9组成三个3位的平方数 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/135/2021_2022_C_E8_B6_A3_E5_91_B3_E7_A8_8B_c97_135460.htm

61.1~9组成三个3位的平方数 将1、2、3、4、5、6、7、8、9九个数字分成三组，每个数字只能用一次，即每组三个数不允许有重复数字，也不许同其它组的三个数字重复，要求每组中的三位数都组成一个平方数。 *问题分析与算法设计 本问题的思路很多，这里介绍一种简单快速的算法。首先求出三位数中不包含0且是某个整数平方的三位数，这样的三位数是不多的。然后将满足条件的三位数进行组合，使得所选出的3个三位数的9个数字没有重复。程序中可以将寻找足条件的三位数的过程和对该三位数进行数字分解的过程结合起来。 *程序与程序注释

```
#include void main() { int a[20],num[20][3],b[10]. /*a:存放满足条件的三位数*/ /*若不是10的倍数，则分解三位数*/ /*分解该三位数中的每一个数字*/ int i,j,k,m,n,t,flag. printf("The 3 squares with 3 different digits each are:\n"). for(j=0,i=11.i if(i!=0) /*若不是10的倍数，则分解三位数*/ { k=i*i. /*分解该三位数中的每一个数字*/ num[j 1][0]=k/100. /*百位*/ num[j 1][1]=k/10. /*十位*/ num[j 1][2]=k. /*个位*/ if(!(num[j 1][0]==num[j 1][1]||num[j 1][0]==num[j 1][2]|| num[j 1][1]==num[j 1][2])) /*若分解的三位数字均不相等*/ a[j]=k. /*j:计数器，统计已找到的满足要求的三位数*/ } for(i=1.i { b[1]=num[i][0]. b[2]=num[i][1]. b[3]=num[i][2]. for(t=i 1.t { b[4]=num[t][0]. /*取第t个数的三位数字*/ b[5]=num[t][1]. b[6]=num[t][2]. for(flag=0,m=1.!flag& m for(n=4.!flag& n
```

if(b[m]==b[n])flag=1. /*flag=1:数字有重复*/ 100Test 下载频道
开通，各类考试题目直接下载。详细请访问 www.100test.com