

C趣味程序百例(26)卡布列克常数 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\\_135788.htm](https://www.100test.com/kao_ti2020/135/2021_2022_C_E8_B6_A3_E5_91_B3_E7_A8_8B_c97_135788.htm)

83.卡布列克常数 验证卡布列克运算。任意一个四位数，只要它们各个位上的数字是不全相同的，就有这样的规律：1)将组成该四位数的四个数字由大到小排列，形成由这四个数字构成的最大的四位数；2)将组成该四位数的四个数字由小到大排列，形成由这四个数字构成的最小的四位数(如果四个数中含有0，则得到的数不足四位)；3)求两个数的差，得到一个新的四位数(高位零保留)。重复以上过程，最后得到的结果是6174，这个数被称为卡布列克数。

\*问题分析与算法设计 题目中给出的处理过程很清楚，算法不需要特殊设计，可按照题目的叙述直接进行验证。

\*程序与程序注释

```
#include <stdio.h>
void vr6174(int). void
parse_sort(int num,int *each). void max_min(int *each,int *max,int
*min). void parse_sort(int num,int *each). int count=0. void main()
{ int n. printf("Enter a number:"). scanf("%d",
&max,&min). /*求数字组成的最大值和最小值*/
num=max-min. /*求最大值和最小值的差*/ printf("[%d]:
%d-%d=%d\n", count,max,min,num). /*输出该步计算过程*/
vr6174(num). /*递归调用自身继续进行卡布列克运算*/ } } void
parse_sort(int num,int *each) { int i,*j,*k,temp. for(i=0;i { j=each
3-i. *j=num. num/=10. } for(i=0;i for(j=each,k=each 1.j if(*j>*k) {
temp=*j.*j=*k.*k=temp.} return. } void max_min(int *each,int
*max,int *min) /*将分解的数字还原为最大整数和最小整数*/ {
int *i. *min=0. for(i=each.i *min=*min*10 *i. *max=0. for(i=each
```

```
3.i>=each.i--)/ *还原为最大的整数*/ *max=*max*10 *i. return. }
```

\*运行结果 1) Enter a number:4312 [1]:4312-1234=3078

[2]:8730-378=8352 [3]:8532-2358=6174 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](http://www.100test.com)