

计算机等级考试C语言程序设计例解(02) PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/134/2021\\_2022\\_\\_E8\\_AE\\_A1\\_E7\\_AE\\_97\\_E6\\_9C\\_BA\\_E7\\_c97\\_134788.htm](https://www.100test.com/kao_ti2020/134/2021_2022__E8_AE_A1_E7_AE_97_E6_9C_BA_E7_c97_134788.htm)

02. 找一个最小的自然数 $x$ ，使它等于不同的两对自然数的三次幂之和，即使得：

$x = a^3 + b^3 = c^3 + d^3$  其中 $a, b, c, d$ 都是自然数，且有 $a \neq c$

和 $a \neq d$  解：问题要找的解是两个自然数对，以自然数对为解的

候选者，如程序能这样枚举解的候选者，使枚举出来的自然数对的三次幂之和构成一个不减的序列，则当发现两个自然数对的三次幂之和相等时，这两对自然数就是问题的解。

将这种思想写成抽象算法描述如下： $\{ i_1=1, j_1=1, x=i_1^3 + j_1^3$

$\} do \{ i_0=i_1, j_0=j_1, min=x, /*保存上一个解的候选者*/ 确定下一对自然数 $i_1, j_1$ .  $x=i_1^3 + j_1^3$ .  $\} while(x \neq min)$ .$

$printf("%d = %d^3 + %d^3 = %d^3 + %d^3 \n", x, i_0, j_0, i_1, j_1)$ . } 问题已转

化成如何按上述要求逐一自然数对。为了寻找产生候选者规则

的线索，将问题简化为找一个最小的自然数 $x$ ，使它等于不同的

两对自然数的平方之和。下面列出部分两个自然数的平方之和

的数表 $s[i][j]$ ，其中： $s[i][j] = i^2 + j^2$  从上面的 $s[i][j]$ 表查得：

$50 = 1^2 + 7^2 = 5^2 + 5^2$   $65 = 1^2 + 8^2 = 4^2 + 7^2$  所以50是两对自然平方

和的最小者。要寻找的产生候选者的规则就是要寻找一个方法，

使枚举产生的候选者(自然数对)的平方和构成以下数列

：2 5 8 10 13 ... 45 50 50 仔细考查表中 $s[i][j]$ 与 $i$ 和 $j$ ，不难发现有

以下性质：1)  $s[i][j] > s[i][k]$ , 对于所有的 $i$ ，当 $j > k$  2)

$s[i][j] > s[k][j]$ , 对于所有的 $j$ ，当 $i > k$  3)  $s[i][j] = s[j][i]$  因问题将自然

数对 $(i, j)$ 和 $(j, i)$ 视为同一个自然数对，所以只需考虑 $i \geq j$ 的情况，

性质1)说明对于表中的每一行，应从左到右逐个考查，

且没有必要保存一整行的候选者供选择，一行只要保存一个已足够。当某行的当前候选者已被确认不是解时，则可生成该行的下一个候选者，等候被考虑。由以上分析，可用下面的两个一维数组表示当前正在考虑的状态： $\text{int } s[?], j[?]$ 。其中？意指数组的大小还未确定。数组 $j[]$ 的成份 $j[k]$ 表示 $s$ 表中的第 $k$ 行当前待考虑的列号。所以， $s[]$ 和 $j[]$ 有关系： $s[k]=k^3 \cdot j[k]^3$ 。将以上分析结果反映到找解方法中，原先的找解算法可改写成如下形式：

```

for(k=1;k { /*每行都从第一列一始考查*/
j[k]=1. s[k]=k*k*k 1. } i=1. /*最初候选者在第一行*/
do {
min=s[i].i0=i.j0=j[i]. 为i行设定下一个候选者存入s[i]. 在s[]中找最小的候选者为正式候选者，并将找到的位置存于i中；
}while(s[i]!=min). printf("%d=%d^3 %d^3=%d^3 %d^3\n",min,i0,j0,i,j[i]). }

```

按上述算法编写程序还有两处不足，需进一步确定或调整：一是为个数不确定的数组 $s[]$ 和 $j[]$ 送初值；另一个是个数不确定的候选者中选正式候选者。由性质1，由性质2)，引入当前考虑的行的范围，最大行 $ih$ 和最小行 $il$ ，其中 $ih$ 是指有 $j[k]$ 为1的最小下标 $k$ ，因为当前还不可能在 $ih$ 行之后选到最小的 $s[i]$ ，所以置初值和选最小元可局限于 $k$ 。再次把上述思想反映到算法中，找解算法又可改写成如下形式：

算法--找一个最小的自然数 $x$ ，使它等于不同的两对自然的三次幂之和

```

{ il=1.ih=1. /*首先只局限于第一行*/
j[1]=1.s[1]=2.i=1. d{ min=s[i].i0=i.j0=j[i]. if(j[i]==1) { /*调整ih,并为j[ih]与s[ih]设定初值*/
ih . j[ih]=1. s[ih]=ih*ih*ih 1. }
if(j[i]==i)il . /*调整il*/ else { /*为i行设定下一个待候选者*/
j[i] . s[i]=i*i*i j[i]*j[i]*j[i]. } /*以下确定新的i，使得s[i]=min(s[il],...s[ih])*/
i=il. for(k=il 1.k if(s[k]

```

```
}while(s[i]!=min(. printf("%d=%d^3 %d^3=%d^3  
%d^3\n",min,i0,j0,i,j[i]). }
```

以上算法可作为最后的算法，下面的程序作了必要的优化，避免反复计算一个整数的三次幂。引入数组p[]，其中p[i]存储i\*i\*i。 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)