

# C语言激趣式教学探索<sup>①</sup>

江代有

(长安大学 信息工程学院, 西安 710064)

**摘要:** C语言语法灵活、规则繁多、枯燥难学, 课堂气氛沉闷, 教学效果低下。要改变这种状况, 当务之急是培养和激发学生的学习兴趣 and 热情。在多年的教学实践中, 笔者进行了一些尝试和探索。

**关键词:** C语言; 教学探索; 激趣方式

## C Language Stimulated Interest in Exploring Teaching

JIANG Dai-You

(Information Engineering Academy, Chang'an University, Xi'an 710064, China)

**Abstract:** C language is flexible in syntax and has many rules. It is boring and difficult to learn. To change this situation, it is necessary to train and motivate the students' interest and enthusiasm. In years of teaching practice, the author conducted a number of attempts and explorations.

**Key words:** C language; teaching; stimulated interest in methods

C语言是高校理工科很多专业的一门实践性和应用性很强的基础课和必修课, 其重要性不必赘述。然而C语言语法灵活、规则繁多、枯燥难学, 很容易使学生产生厌学情绪, 教学效果极为低下。怎样才能克服学生的厌学情绪, 改变这种教学状况呢? 笔者认为当务之急是要培养和激发学生的学习兴趣 and 热情, 因为“兴趣与爱好是最好的老师”。下面结合自己的教学实践, 对C语言的激趣式教学进行一些探索。

### 1 比喻法

在C语言的教学过程中, 很多问题如果单纯地按照其本义讲解很难理解, 学生往往会听得一头雾水。对于这些难理解的问题笔者通常使用比喻法加以讲解。例如:

在C语言中, 经常要用到变量的交换语句如{t=a;a=b;b=t;}。语句虽然简单, 但对于初学者还是很难理解。此时, 可以将a、b、t比喻为瓶子, a瓶子装的是醋, b瓶子装的是酱油。若要将a瓶子的醋与b瓶子的酱油进行交换, 必须借助空瓶子t。先将瓶子a

中的醋倒入瓶子t中(语句描述为t=a); 接着将瓶子b中的酱油倒入瓶子a中(语句描述为a=b), 最后将瓶子t中的醋倒入瓶子b中(语句描述为b=t), 从而成功地借助第三个瓶子t将瓶子a中的醋和瓶子b中的酱油进行了互换。

在讲到双重循环时, 将外循环比喻为时钟的分针, 内循环比喻为时钟的秒针, 秒针走一圈分针走一格, 同样的道理是: 内循环执行完一个轮回, 外循环才执行一次。讲到全局变量和局部变量时, 将全局变量比喻为国家统一的法律和法规, 局部变量比喻为各省根据需要制定的地方法律和法规。

类似的比喻还有很多, 例如: 在讲到指针时将指针比喻为宿舍的房间号或开启房间的钥匙。而指针的两个运算符&和\*是两个互逆的单目运算符, &运算符是对其后的操作数求地址, \*运算符是对其后的指针变量求值。也就是说, 求&运算得到的是房间号, 求\*运算得到的是在该房间的人。讲到二维数组的行指针和列指针时, 将行指针比喻为排长, 列指针比喻为班长, 排长“指向”班, 他走一步就跳过一个班, 即“行”;

<sup>①</sup> 收稿时间:2011-10-19;收到修改稿时间:2011-12-12

而班长“指向”战士，他走一步只指向一个战士，即“列”<sup>[4]</sup>。

这些形象生动的例子使得抽象概念具体化，复杂问题简单化，从而激发了学生的学习兴趣，加深了对知识的理解和掌握。

## 2 比较法

C 语言中有很多相似或相近的内容，在教学中笔者经常用比较法对它们的异同点进行反复讲解，从而加深学生对内容的理解与掌握，收到事半功倍的效果。例如：

在讲到 printf() 函数与 scanf() 函数时，讲了它们的共同点（都可以用 d、o、x、f、c、s 等格式符）后，着重强调了它们的区别：scanf() 函数里的变量前一定要加取地址运算符 &，而 printf() 函数则不加；printf() 函数在输出实数时，可以限定输出小数位数，而 scanf() 函数则不能。

在讲到冒泡法、选择法与插入法排序时，笔者用具体实例进行比较。叫学生做 10 张卡片，每张卡片写上大小不同的数字并把卡片一字排开放在桌上，然后将相邻的卡片两两进行比较，小的放前，大的放后，这样经过反复比较交换后才能排好序，这就是冒泡法。而选择法则是从第一张卡片开始，找出身高最高的卡片与第一张互换，再从第二张卡片开始找到身高最高的卡片与第二张互换，依此类推直至完成。插入法排序则是先从无序的卡片中拿出第一张放在别处，然后再从剩下的无序卡片中拿出第一张亦即原第二张与刚拿出的进行比较，若原第二张身高高于原第一张，则插入在前，否则插入在后。可以看出选择法速度较慢，插入法不适用于大量数据的排序<sup>[5]</sup>。

在讲到指向数组行的指针定义 int (\*p)[] 与指针数组的定义 int \*p[]，指向函数的指针定义 int (\*p)() 与返回指针的函数定义 int \*p() 形式时，很多学生会混淆它们的定义。笔者重点讲了它们形式的异同：如果 \*p 加上括号就表明它们是指针，是什么指针呢？如果指针后面跟“[]”则表明是指向数组行的，如果后面跟“()”，则表明是指向函数的。经过这样的比较讲解，学生记忆起来更加清晰。

在教学过程中，笔者还经常通过对程序进行小小的改动来启发学生思考程序改动前后的运行结果，比较改动前后的源程序及运行结果的变化，从而激发学

生的学习兴趣，加深对知识的理解。例如：下面的程序运行后会得到九九乘法表。

```
main()
{int i,j;
for (i=1;i<=9;i++)
for(j=1;j<=i;j++)
{printf(“%d”,i*j);
if (i==j)
printf(“\n”);}}
```

如果将该程序的两个 for 循环语句改为如下语句：

```
for (i=9;i=1;i--)
for(j=9;j=i;j--)
```

其它语句不变，程序的运行结果会发生什么变化呢？同学们纷纷讨论，气氛十分活跃。

下面程序的功能可以实现两个字符串的连接：

```
#include “stdio.h”
void main()
{char s1[10],s2[10];
int i,j;
gets(s1);get(s2);
for(i=0;s1[i]!='\0';i++);
for(j=0;s2[j]!='\0';j++)
s1[i]=s2[j];
s1[i]='\0';
puts(s1);}
```

在讲解上面的程序时，笔者让学生把程序进行小小的修改以实现两个字符串的复制。课堂气氛很快活跃起来，相互讨论的有之，抢先回答的有之。有的学生很快就想出了答案：把第一个 for 循环括号后的分号删除，再把其后的三行语句加上花括弧（{}）即可。学生普遍反应这种教学效果极好，过后印象深刻。

## 3 启发法

启发式教学是从学生的知识基础和接受能力出发，通过教与学的互动，充分调动学生的主动性和创造性。在教师的启发诱导下，引导学生主动学习积极思考。在教学过程中，笔者经常应用启发式教学方法，激发学生的学习兴趣，培养学生的逻辑思维 and 实际动手能力。

例如，讲循环的概念时，笔者先设置问题“编程计算 1+2+3 的和”。根据前面所学的知识，学生很容易

想到：需要定义3个变量用于存放1、2、3。再进一步要求：“编程计算1+2+3+……+100的和”。如果再定义100个变量来存放这100个数，很显然是不现实的。此时可顺理成章地引入循环的概念、循环语句等，并进一步启发学生什么情况使用while循环，什么情况使用for循环。

讲到数组的概念时，笔者也先设置问题“编程计算3个同学的平均成绩”。很显然需要定义3个类型相同的变量用于存放这3个同学的成绩。再进一步要求：“计算10个同学的平均成绩”。此时引入数组的概念、性质等，学生不仅能很容易记住数组概念，还能知道在什么情况下使用数组，事半功倍。

讲到结构体时，笔者把上述问题改为“计算10个同学的平均成绩并可根据姓名查询成绩”，以此引入结构体的概念，启发学生比较数组与结构体概念的异同。从而得出结论：数组只能存放相同类型的数据，而结构体可以存放不同类型的数据。例如：一个学生的姓名、性别、年龄、成绩等，就需要用到字符数组、字型、整型、浮点型等多种数据类型。

讲到函数时，笔者先让学生“利用选择语句编程求出两个变量中的最大值”，问题很简单，学生很快就做出来了。笔者进一步要求“编程求出5个变量的最大值”，如果还按原有的方法将变量两两进行比较，程序需要很多步骤，非常烦琐。此时因势利导地引入函数的概念，先编写比较两个变量大小的函数然后再进行调用，启发学生掌握函数的定义、调用等格式。

#### 4 趣例法

趣例法可以充分调动学生学习的积极性，将学生的被动学习转换为主动学习，从而提高教学效果。在课程的进行中，笔者就经常举一些趣味性的算例来讲解必要的知识点，例如“百钱买百鸡”、“猴子吃桃”、“菲波纳契数列”、“水仙花数”、“汉诺塔”等有趣例子，以此来提高学生的学习兴趣，活跃课堂氛围。

例如，在介绍多重循环的时候笔者就举了以下例子：

三对情侣参加婚礼，三个新郎为A、B、C，三个新娘为X、Y、Z，有人不知道谁和谁结婚，于是询问了六位新人中的三位，但听到的回答是这样的：A说他将和X结婚；X说她的未婚夫是C；C说他将和Z结婚。这人听后知道他们在开玩笑，全是假话。请编

程找出谁将和谁结婚<sup>[2]</sup>。

对于这样的问题，可以先进行算法分析：将X和A结婚表示为X='A'，将Y不与A结婚表示为Y!='A'。按照题目中的叙述可以写出表达式：

X!='A'          A不与X结婚  
X!='C'          X的未婚夫不是C  
Z!='C'          C不与Z结婚

题意还隐含着X、Y、Z三个新娘不能结为配偶，则有：

X!=y 且 x!=z 且 y!=z

代入上述表达式中进行推理运算，编程如下：

```
#include "stdio.h"
void main()
{int x,y,z;
for(x='A';x<='C';x++)
for(y='A';y<='C';y++)
{if(x!=y)
for(z='A';z<='C';z++)
{if(x!=z && y!=z)
{if(x!='A' && x!='C' && z!='C')
printf("X to %c\n Y to %c\n Z to %c\n",x,y,z);
}}}}
```

三个循环分别穷举了X、Y、Z的全部可能配偶，运算结果显示X与B结婚，Y与C结婚，Z与A结婚。课堂实践证明，上面的例子既满足学生的好奇心，又贴近生活，还能充分说明多重循环的使用，何乐而不为呢？

除了上述的几种激趣方式外，在教学过程中，笔者还经常使用陷阱法和演示法等激趣方式。陷阱法就是在讲解程序示例时，笔者经常有意将语句输错或漏写语句，制造一些“陷阱”，通过提问让学生判断对错，提高他们的观察力和敏锐性，激发他们寻找程序错误的兴趣，活跃课堂氛围。例如：忘记定义变量，语句后漏掉分号、定义的数据类型与输出的格式符不一致，scanf()函数中漏写变量的取地址运算符，混淆使用符号“=”与“==”，括号不匹配，数组下标越界等。演示法就是在教学过程中，借助视频、动画等对一些算法进行模拟，比如在讲解汉诺塔问题和排序法时，笔者用flash做了模拟动画来进行演示，以此把抽象问题形象化，激发学生的学习兴趣，起到事半功倍的效果。

(下转第243页)

```

bsp_nprocs(); // bsp 进程数查询;
bsp_sync(); // bsp 进程的栅栏同步;
bsp_store (to, from, data, size, num);
bsp_fetch (to, from, data, size, num); // bsp 进程之
间最优种群的迁徙。

```

求解过程的参数统计如表 1 所示,从中可以看出随着处理器个数的增加,最优解出现的进化代数明显降低。

表 1 并行 GP 实现的参数统计

处理器数	最优解出现的进化代数	运行时间 (秒)	加速比(%)
1	27	5437	0
2	24	4056	25.4
4	21	3121	42.6
8	23	2417	55.5
16	14	1879	65.4

## 5 结语

本文主要研究了基于 BSP 的并遗传程序设计问题。基于两点考虑,一是全面分析了基于 BSP 并行计算的粗粒度 GP 求解方法,二是遗传程序设计是遗传算法中的一个重要分支,国内在该领域的研究不够热烈,借此文抛砖引玉,希望能引起同行的兴趣。

(上接第 214 页)

## 5 结语

经教学实践检验,激发兴趣的教学模式效果十分显著。在发挥教师主导作用的前提下,采用各种方式激发学生的学习兴趣,让学生主动、愉快地学习,充分调动学生的积极性、主动性和创造性,引导学生融会贯通地掌握 C 语言,从而取得良好的教学效果。

### 参考文献

1 谭浩强.C 程序设计.第 3 版.北京:清华大学出版社,2005.

### 参考文献

- 1 Koza JR. Genetic Programming: on the Programming of Computers by means of Natural Selection. MIT Press. 1996.
- 2 刘大有,卢奕南,王飞.遗传程序设计方法综述.计算机研究与发展,2001,38(2):213-222.
- 3 Alberto C, Amelia Z, Sebastián V. A parallel genetic programming algorithm for classification. Lecture Notes in Computer Science, 2011,6678:172-181.
- 4 Kenichi M, Takaya A, Takashi O. Autonomous acquisition of cooperative behavior based on a theory of mind using parallel genetic network programming. Artificial Life and Robotics, 2011,16(2):157-161.
- 5 Kent S. Speeding up Genetic Programming: A Parallel BSP Implementation. Genetic Programming 1996: Proc. of the First Annual Conference. 28-31 Jul. 1996.
- 6 Corne DW, Oates MJ, Kell DB. On fitness distributions and expected fitness gain of mutation rates in parallel evolutionary algorithms. Parallel Problem Solving from Nature - PPSN VII. 2002. 132-141.
- 7 Valiant LG. A bridging model for parallel computation. Communications of the Association for Computing Machinery, 1990,33(8):103-111.

- 2 陈翔鹰,陈英.C 语言趣味程序百例精解.北京:北京理工大学出版社,1994.56-90.
- 3 盛雪丰.浅谈 C 语言程序设计教学的艺术化.科技信息,2009,(17):207.
- 4 葛丽萍.C 语言教学中的启发式教学应用研究.农业网络信息,2008,(6):130-131.
- 5 唐小健.浅谈比较教学法在《C 语言》教学中运用.职业教育研究,2010,(5):139.