

浅谈文科生学习 C 程序设计技巧

曾文静

四川大学锦城学院 计算机与软件学院 四川 成都 611731

【摘要】C 语言是一门基础的、通用的程序设计语。C 语言语法简单，应用于底层开发，被作为众多高校计算机专业的通识课程。本文就本人作为一名文科生学习 C 程序设计的过程，系统阐述了 C 程序设计的三大程序设计语法、学习 C 程序设计的困难点以及解决措施。

【关键词】C 语言程序设计；顺序结构程序；选择结构程序；循环结构程序

引言

现如今，互联网已成为全球产业转型重要的助推器之一。各行各业、计算机技术与互联网技术三者在未来都将实现高精度融合。自计算机问世到现在，不过百年的计算机产业有着不可限量的发展潜力。面对市场对专业计算机人才的强大需求，与计算机相关的专业成为了历年来高考学生志愿填报的热门专业。计算机专业属于理工科类，文科生在就读该类专业中将会存在更大的阻力。C 语言作为最基础的计算机语言，充当了引路人的角色。那么，C 语言的主要语法是什么呢？C 程序设计能力如何提高呢？C 程序设计思维如何培养？

1 C 程序设计的语法

1.1 顺序程序设计

1.1.1 C 程序的基本语句

1) 说明语句：数据设计语句（定义数据类型，完成对数据的初始化工作）。

2) 执行语句：操作设计语句（完成对数据的加工和流程控制工作）。

3) 简单语句：1. 表达式语句，eg: $x++$ ；2. 函数调用语句，eg: `scanf(“%d”, &y);`；

4) 复合语句：一组花括号所包含的简单语句，eg: `{n++; scanf(“%d”, &x);}`

1.1.2 C 程序顺序程序设计的操作方法

编译器编译和运行代码的顺序是从上至下、由左至右，因此在编写代码的时候需要充分考虑代码的编译情况。

1.2 选择结构程序设计

1.2.1 if 语句实现选择结构

if 语句作为最基础的选择语句，可通过 if 语句的形式进行延伸，以达到其他选择结构的编写。if 表达式逻辑值为真的情况，编译器随即执行紧跟着的代码块。若想对逻辑值为假的情况也进行操作，可以加上 else 语句；若想对逻辑值为假的情况再区分讨论，可以加上 else if 语句，但需注意的是 else 语句需在最后。

语法：

1) if 关键字（表达式）后不加 ‘;’。

2) else 关键字后不加表达式；且与上面尚未配对的 if 构成一对。

3) 若需执行两条及以上的基本语句需使用花括号限制。

if(表达式)

{

代码块；

}

else if(表达式)

{

代码块；

}

else

{

代码块；

}

1.2.2 switch 语句实现选择结构

通过 if 语句的嵌套使用可以将多分支选择实现，但是该方法会大大提高算法的时间复杂度，因此需要采取另一种语句从而实现多分支选择——switch 语句。通过对 switch 表达式与 case 常量的值进行比较，符合则实现该 case 后的语句，不符合的话，则依次向下进行判断，直到遇到 break 关键词跳出 switch 结构。

switch 语句的语法：

1) switch 表达式类型只能为整型或者字符型。

2) case 所跟的常量值类型需要与表达式的一致。

switch(表达式)

{

case 1:

语句；

case 2:

语句；

case n:

语句；

default:

语句；

}

1.3 循环结构程序设计

重复处理也是程序需要常常处理的问题之一。重复处理等同于循环操作，在解决该类问题的时候，需要用到 while 语句或者 for 语句。

1.3.1 while 语句实现循环结构

当 while 的表达式逻辑值为真时, 将会一直执行代码块, 反之, 则终止循环流程。

while 语句的语法:

while 关键词后跟的表达式即为满足循环的条件。

执行操作语句的条数为两条及以上, 需要用花括号限制。

while 语句的使用情况: 在不知道循环次数的时候。

while(表达式)

{

代码块;

}

1.3.2 for 语句实现循环结构

for 语句更能灵活使用, 可以完全发挥 while 语句的作用。

for 语句的语法:

1) for 语句具有三个表达式, 依次用分号进行分隔, 表达式可以不写在括号内, 如果是同时省略三个表达式的话, 则该循环不存在实用价值。

2) 执行操作语句的条数为两条及以上, 需要用花括号限制。

for(循环变量赋初值; 循环条件; 循环变量增值)

{

代码块;

}

2 学习 C 程序设计的技巧

1. 海量刷题是基础, 分析理解示例代码

自中学以来, 若是想要提升成绩, 采取的最基本的方法就是题海战术。这一战术也同样适用于提高编程能力。刚开始可以选择简单的题, 在练习中理解 C 语言的各种知识。在有了一定的编程基础之后, 可以着手于解决 C 语言问题, 例如: 约瑟夫闭环、Hanoi 塔问题、兑换硬币问题等等。

解决一道题应为三步骤: 解、编、析。

解, 即为解答题目, 从题干中得到有用条件, 考虑该题的编程实现应分为若干个功能块且分别从用何种方法实现。编, 即为编写代码, 根据上一步骤, 可以轻易写出程序的大致框架, 在本步骤就该充分考虑细节问题(各种边界问题等等)。析, 即为分析示例代码。因为示例代码不仅仅可以作为问题的答案, 更可以助力编程能力提升。初学者在阅读分析示例代码的同时, 应该要实现从低阶理解到高阶运用, 从做什么到如何做, 做到举一反三。

2.1 将中学数学与编程结合, 降低理解难度

业内有一句话叫作“数学是计算机之母”, 因此, 可见数学对于计算机来说异常重要的。^[2] 初入门的学习者可以尝试着采用中学数学的角度分析编程问题。通过中学数学与编程的有机结合, 拉近自身与编程的距离感, 更大的激发出自身对编程的兴趣。

例如: 用 C 语言语句计算出 $1+2+3+\dots+100$ 的总和

分析: 可将该式子看作首项为 1, 公差为 1, 末项为 100 的等差数列。

解法 1: 直接法

使用 for 循环将所有加数依次相加

```
#include<stdio.h>
```

```
main( )
```

```
{
```

```
int i, sum=0;
```

```
for(i=1;i<=100;i++)
```

```
{
```

```
sum+=i;
```

```
}
```

```
}
```

解法 2: 公式法

利用等差数列的求和公式, 求出前 100 项的和就为该式子的计算结果。

```
#include<stdio.h>
```

```
main( )
```

```
{
```

```
int front, rear;
```

```
printf(“请输入首项和末项的值(用空格分隔):\n”);
```

```
scanf(“%d %d”, &front, &rear);
```

```
printf(“1+2+3+...+100=%d”, 100*(front+rear)/2);
```

```
}
```

3 自我调试程序, 锻炼修改能力

编译器具备的功能是检查代码的语法是否存在错误, 并不可以检查到逻辑等问题的错误。调试代码是个极其痛苦的过程, 绝对不会是一帆风顺的, 需要详细记录调试的情况以及解决办法, 深入探析出现的问题, 找出解决之策, 总结成功的经验、失败的教训, 做到举一反三。

结语

总而言之, C 语言对于入门计算机行业是非常重要的。C 语言是计算机类专业的通识课, 可见它是尤为重要。虽然计算机属于理工科, 文科生接触起来会相对吃力, 但是正所谓星光不负赶路人, 只要坚持学下去, 就一定会收获的那一天。

【参考文献】

[1] 谭浩强. C 程序设计(第五版)[M]. 北京: 清华大学出版社, 2017: 111.