

# The Design and Practice of a Multi-solution Teaching Strategy: Taking the C Language Programming Course as an Example

Leigang Hao\*, Wenchao Guo, Hang Qi, Zhili Liu

Handan Vocational College of Science and Technology, Handan 056046, China

\*Corresponding author: Leigang Hao, hdmrhao@126.com

**Copyright:** © 2024 Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY 4.0), permitting distribution and reproduction in any medium, provided the original work is cited.

**Abstract:** This paper explores one problem with multiple solutions teaching strategy applied to the integration of theory and practice teaching courses in C language programming course. According to the cognitive characteristics of students for the same problem combined with the specific learning situation of higher vocational colleges, the course has adopted the three methods of basic flow control statement, switch statement, and array. Showing the methods and skills to solve the problem from different angles with the gradual implementation of the specific teaching helps students deepen their understanding of C language knowledge. It also guides students to use knowledge to solve practical problems, to achieve the teaching goals.

**Keywords:** One problem multiple solutions; Teaching strategy; C language; Practice

**Online publication:** June 20, 2024

## 1. Introduction

Multiple solutions to a problem means that the same problem or situation can have multiple different solutions or solutions. Multiple solutions to a problem are also very important in specific education and teaching<sup>[1]</sup>. The multi-perspective thinking and multi-method solution of a problem not only can broaden the thinking horizon of solving the problem but also help students systematize the knowledge they have learned. This paper takes the C language programming course as an example to show the specific teaching design and practice of one problem multi-solution teaching strategy in the teaching of the “integration of theory and practice” course in higher vocational colleges<sup>[2]</sup>.

## 2. The status and characteristics of C language programming courses in higher vocational colleges

As a professional basic course for electronic information majors in higher vocational colleges, the study of this course cultivates students' basic C program reading ability, simple algorithm writing ability, and the basic

method of programming. This course gradually forms the correct programming ideas and good programming style in students so that those who have not been in contact with programming can develop good program writing habits and programming thinking habits with the help of C language to step into the threshold of programming and lay a good knowledge foundation for programming <sup>[3]</sup>.

### **3. Key links to implement the multi-solution teaching strategy of one question**

The core of implementing the multi-solution teaching strategy is the teacher, the goal is the student, and the way is the classroom. In the implementation of the multi-solution teaching strategy, teachers should pay attention to the following aspects. First, teachers should have solid professional domain knowledge and programming skills and be able to use different solutions flexibly. Second, teachers should have good guiding ability and problem-solving thinking to stimulate students' thinking flexibility and innovation ability. Third, teachers should have good classroom control ability. Fourthly, teachers should have a good ability in instructional design <sup>[4]</sup>.

## **4. The specific application and practice of the multi-solution teaching strategy**

### **4.1. Design objectives**

Through a teaching case, the knowledge of the three instructional design units is connected in series to help students deepen their understanding of the basic process control statements, switch statements, and arrays, to achieve the goal of “string beads into lines” and the effect of applying what they have learned <sup>[5]</sup>.

### **4.2. Description of the subject**

The study subject is to type a date on the keyboard and calculate the day of the current year.

### **4.3. Ideological and political education of the course**

According to the pattern of the moon's movement, the ancients defined the time between the first sighting of a full moon and the second sighting of a full moon as one month. This is the commonly used lunar calendar. According to the relationship between the four seasons and the sun, it is prescribed as a year, which is the commonly used solar calendar. Compared with the lunar calendar, the number of days in a year in the solar calendar is more than 11 days longer than the lunar calendar <sup>[6]</sup>. To solve this problem of unifying the two calendars for calculating the number of days, the ancients used the method of having 7 leap years out of every 19 years. After careful calculation, Zu Chongzhi found that the leap method would differ by one day every 200 years. To solve this problem, he broke the traditional concept through repeated calculations, accurately calculated that a regression year is 365.24284481 days, which is only 50 seconds different from the modern scientific measurement of the year, and then calculated the new leap method of adding 144 leap months every 391 years <sup>[7]</sup>.

### **4.4. Instructional design unit 1: Basic flow control statements**

#### **4.4.1. The characteristics of basic flow control statements**

A basic flow control statement is the basis of building program logic. By mastering the basic flow control statement, students can realize the condition judgment, loop execution, and jump control of the program, and then lay a foundation for solving more complex programming tasks in the future <sup>[4]</sup>.

#### 4.4.2. Core program source code

```
// accumulate the number of days before month by looping
for(index=1; index<month; index++){
    if(index == 1||index == 3||index == 5||index == 7||index == 8||index == 10||index == 12){
        sums += 31;
    }else if(index == 2){
        // Check if it's a leap year
        if((year %4 ==0 && year%100 !=0)||(year % 400 ==0)){
            sums += 29;
        }else {
            sums += 28;
        }
    }else {
        sums += 30;
    }
}
// Accumulate the date of the month
sums+= day;
```

#### 4.4.3. Teaching reflection and summary

In this scheme, the number of days before the current input month is accumulated through the for-loop statement, and the number of days in the current month is accumulated to achieve the goal of calculating the current date of the day of the year. The difficulties in the implementation of this program lie in the multi-level judgment and the nesting of loops and judgments. Many students fail to run the program due to unclear logic and unskilled sentence structure. Teachers can guide students to draw a flow chart to gradually solve each difficulty and achieve the goal of deepening their understanding <sup>[8]</sup>.

### 4.5. Teaching design unit 2: Switch statement

#### 4.5.1. The characteristics of switch statements

The switch statement provides a way of multi-branch selection. The syntax structure is relatively simple and clear, and the readability is high. It is suitable for some cases that need to select the execution branch according to different conditions, such as selecting different operations according to user input.

#### 4.5.2. Core program source code

```
// Use switch... case features add up the number of days per month
switch(month){
    case 12:
    case 11: if(month>11){sums+=30; }
    case 10: if(month>10){sums+=31; }
    case 9:  if(month>9){sums+=30; }
    case 8:  if(month>8){sums+=31; }
    case 7:  if(month>7){sums+=31; }
    case 6:  if(month>6){sums+=30; }
```

```

    case 5: if(month>5){sums+=31; }
    case 4: if(month>4){sums+=30; }
    case 3: if(month>3){sums+=31; }
    case 2: if(month>2){
// Check if it's a leap year
if((year %4 ==0 && year%100 !=0)||year % 400 ==0){
    sums+=29;
}else{
sums+=28;
}
}
    case 1: if(month>1){sums+=31; }
}
// Add the date of the month
sums += day;

```

### 4.5.3. Teaching reflection and summary

This program uses the multi-branch characteristics of the switch statement, cleverly uses the reverse order to accumulate the days of the month, and finally accumulates the number of the date of the month to achieve the goal of calculating the current date of the day of the year. The case statement can indicate the current processing of the content. In the case of additional conditions, it can effectively solve the problem of accumulated errors in the month<sup>[9]</sup>. By comparing with the basic process control statement, students can further deepen their understanding of the use conditions and applicable scenarios of switch statements, pay attention to avoid easy errors, and improve their mastery of switch statements through practice<sup>[10]</sup>.

## 4.6. Instructional design unit 3: Array

### 4.6.1. The characteristics of the array

The array is widely used in many programming tasks. It is a structure used to store multiple data of the same type. Through index access to the elements, it can quickly locate and manipulate the data at a specific location.

### 4.6.2. Core program source code

```

// Use an array to store the days of the month
Int days [13] =,31,28,31,30,31,30,31,31,30,31,30,31 {0};
// Check if it is a leap year
if((year %4 ==0 && year%100 !=0)||year % 400 ==0){
    days[2]=29;
}
// accumulate the days of the previous month by looping
for(index=0; index<month; index++){
    sums += days[index];
}
// Accumulate the date of the month
sums+= day;

```

### 4.6.3. Teaching reflection and summary

This scheme cleverly uses the characteristics of array continuous storage data and efficient memory access, cleverly stores the days of the 12 months of the year into the array continuously, and uses a simple loop statement to achieve the accumulation of days, which effectively avoids the multiple judgment problems similar to if and switch statements, and improves the readability and execution efficiency of the program<sup>[11]</sup>. Through the study of this case, students can further deepen their understanding of the characteristics of the array and learn the methods of using the array in the future<sup>[12]</sup>.

## 5. Teaching effect analysis

Basic flow control statement, switch statement, and array are three teaching units of continuous design in the C language programming course. Showing the methods and skills to solve problems from different angles helps students to deepen their understanding of C language knowledge and guides students to use knowledge to solve practical problems to achieve the teaching goal of “string beads into lines, simple and efficient”<sup>[13]</sup>. Through questionnaire analysis, it is found that students have a firm grasp of the basic knowledge of C language programming, and have improved the practical ability of structured programming and using C language to solve practical problems. Many students have become interested in programming language and are willing to learn harder content through the study of this course<sup>[14]</sup>.

## 6. Summary

Different thinking angles of the same problem will lead to different methods and strategies. Multiple solutions to a problem is the process of inspiring and guiding students to use different methods to solve the same problem from different angles. The thinking and comparison of various methods and processes not only can help students better understand and master the basic knowledge and key points of C language programming but also deepen their understanding of knowledge in thinking and comparison, broaden the thinking, and lay a foundation for the future application of what they have learned<sup>[15]</sup>. This paper takes the C language programming course as an example to show the specific teaching design and practice of one problem multi-solution teaching strategy in the teaching of the integration of theory and practice courses in higher vocational colleges, hoping to provide a reference for other teachers.

## Disclosure statement

The authors declare no conflict of interest.

## References

- [1] Wu L, 2012, Teaching Practice of Computer C Language. Application of Integrated Circuits, 39(10): 160–161.
- [2] Zhang WH, 2020, Classroom Teaching Design of Bubble Sort in C Language. Computer Knowledge and Technology, 16(32): 183–184.
- [3] Liang JH, 2019, Analysis and Research of Sorting Algorithm in Data Structure Based on C Language. Journal of Luohe Vocational and Technical College, 18(03): 32–34.
- [4] Li YH, Xin JJ, 2023, Multiple Teaching of C Language Programming Based on OBE Concept. Journal of Shangluo University, 37(05): 77–83.

- [5] Yang RN, Lu YL, Yan DQ, et al., 2022, Teaching Practice of Education Function Mining of C Language Course. *China Information Technology Education*, 2022(12): 94–96.
- [6] Bi XL, 2022, Reconsidering the Status of the Ancient Chinese Solar Calendar Represented by the 24 Solar Terms and Its Impact on Chinese Political Culture. *Yuejiang Academic Journal*, 2022(004): 14.
- [7] Zhao XX, Xia W, 2023, Teaching Practice of JavaScript Programming Course. *Application of Integrated Circuits*, 40(5): 240–241.
- [8] Zhang YH, 2018, Teaching Practice and Thinking of High School Information Technology Pointing to Deep Learning: Taking the Teaching of Loop Nesting as an Example. *China Information Technology Education*, 2018(15): 3.
- [9] Wang J, Li M, 2024, Research and Practice on Blended Teaching of C Language Programming in Higher Vocational Colleges. *Modern Commerce and Industry*, 45(08): 267–268.
- [10] He YH, 2024, Research on Ideological and Political Case Design of C Language Loop Sentence. *Computer Programming Skills and Maintenance*, 2024(02): 8–11.
- [11] Wu L, 2012, Teaching Practice of Computer C Language. *Application of Integrated Circuits*, 39(10): 160–161.
- [12] Wang S, 2021, Thinking about C Language Programming Course Teaching in Vocational Colleges. *Occupation*, 2021(22): 88–90.
- [13] Zhang F, Zhang HR, 2021, Analysis of the Application Effect of Experiential Teaching Model Based on the Cultivation of Computational Thinking: Taking C Language Programming Course in Higher Vocational Colleges as an Example. *China Modern Educational Equipment*, 2021(05): 149–151.
- [14] Li H, 2020, Analysis on the Construction of Blended Learning Model of C Language Course in Colleges and Universities. *Computer Knowledge and Technology*, 16(36): 166–167.
- [15] Wang JP, 2020, Research on the Teaching Strategy of Flipped Classroom Based on C Language Programming. *Chinese Journal of Multimedia and Network Teaching*, 2020(03): 5–7.

**Publisher's note**

Bio-Byword Scientific Publishing remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.