

An Application of Case Method on Laboratory Animal for C++ Teaching

Jun Tan¹ Ming zhong² Jian Liang² Zhaoyuan Huang¹

¹ Guangdong Province Key Laboratory of Computational Science, School of Math. & Compute, Sun Yat-sen University, Guangzhou, 510275, P. R. China
²Vins Information System Technology Company Ltd, Guangzhou, 510275, P. R. China

Abstract

Case teaching method can make students active in program course. Therefore, it is adopted in C++ training. Laboratory Animal System (LAS) is a standalone case for management of laboratory animal, the code is written in Visual C++ and it can be installed in Windows system. In this paper, we present an overview of LAS architectural design and user interface by C++. The case method will provide an interactive learning environment for students. According to the survey of feedback, it is summarized that students will be able to learn C++ program and gain knowledge more quickly and effectively than students using traditional methods of teaching.

Keywords: laboratory animal, C++, e-learning, case teaching, programming

1. Introduction

Among the six levels of abstraction of educational objectives proposed by Bloom [1], we target application (use methods in new situations, solve problems using knowledge) in our ITS, as opposed to program synthesis, which has been the focus of many earlier works (e.g., Model Tutor [2], program style[3]). Our work focuses on case teaching rather than the entire programming enterprise. We have been using ASP.NET to model the domain for LAS[4]. In our case, this

would be a model of the C++ language, consisting of objects in the language and their mechanisms of interaction. This model is used to simulate the expected knowledge-point of some particular C++ language construct, e.g., C++ pointers. The correct behaviors of the student are trained for that artifact. In our paper, the behavioral discrepancies are used to generate feedback to tutor.

This paper has three goals. The first goal is to present our approach to teaching object-oriented program in general and C++ in particular. Secondly, we discuss our experience associated with teaching C++, in particular focusing on the advantages and disadvantages of the method. Finally, it is to examine our curriculum design, it trains student capability for memory, speed, attention, flexibility, problem solving in program (shown in Fig1.).

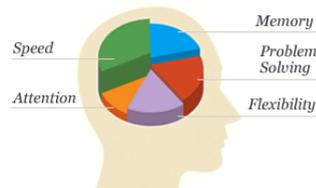


Fig. 1: Training goals of method

2. LAS Background

Laboratory animals such as Rhesus monkey and rats, they have been regarded as primary research tools because of medicinal studies[5]. To share and analyze multi-scale structural and functional data and

ultimately to integrate them with genomic and gene expression data, and to reduce the number of animals needed, management of LAS has play more important role for scientists in the world.

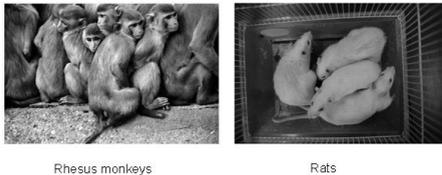


Fig. 2: Laboratory animal

We simulate Laboratory Animal System(LAS) with C/C++ programming technology in which users can roam and complete operation process in accordance with management regulations of barrier housing facilities. LAS system suits general-embedding data, and it has higher accuracy as the analysis of data can reflect issue and management information of specimens. LAS will offer a powerful new tool for exploring the morphologic effects of laboratory animals. Also it is good exercise for students using C/C++ program.

3. LAS system and example

In this section, we teach student how to design LAS system, the major function includes Define Laboratory Animal, Apply for Laboratory Animal, Return Laboratory Animal, Compute fee.

3.1. Flowchart

Data structures of the laboratory animal are defined, including the basis property (num, name, sex, standard) and extend attribute (unit, count, price). From above laboratory animal status, user may apply for animals; while user input data to requisition form accordance with current format. After finished the task of inputting, user push “submit” button, system will submit the requisition form to administrator. He can reject or accept the requisition form. User may obtain the laborato-

ry animal make use of the requisition form.

After related experiment, user must return laboratory animals, so it needs he records the information about laboratory animals and experiment. User inputs category and units of consumable, LAS system computes total expense based on fee standard of consumable. If animals were dead, LAS record the related information about laboratory animal carcass. Fig.3 demonstrate the flowchart of the proposed LAS system.

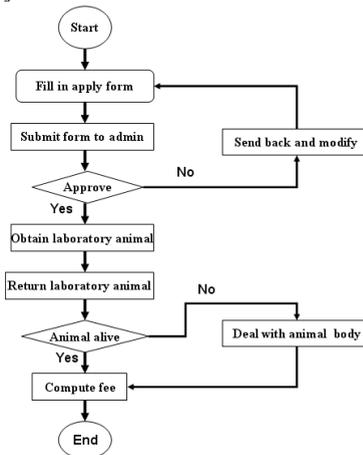


Fig. 3: Flowchart of LAS.

LAS automatically compute total expense of animal on standard fee, See Algorithm 1 for more details about this calculation.

Algorithm 1: compute fee of laboratory animal

Input:

- laboratory animal data: $X = (num, name, \dots, price)$
- apply user info: username, section
- check variant approve=true
- 01.set p_1 as animal price.
- 02.set c_1 as animal count.
- 03.set loop as Fee No., sum as total fee.
- 04. $t = 0, sum = 0.$
- 05. while($t \leq loop$)
- (a) set $sum = sum + p_1 * c_1$
- (b) if $p_1 \geq 0$ and $c_1 \geq 0$ continue
- (c) else print "price or count error!" break.
- (d) $t = t + 1.$
- 06. End

Output: sum

Let n as fee number, p_i is price, c_i is count, so we compute fee sum as follow:

$$sum = \sum_{i=1}^n p_i c_i + SF \quad (1)$$

SF is special fee if animal was dead.

3.2. Programming example

The limited space of the article does not show an full analysis. However, we will try to give some idea of the functionality of LAS with two versions of a (very) small program, presented in Fig. 4. and 5. These programs define animal with struct format and output data to file whether data of animal is correct.

```

1  #include <iostream>
2  #include <string>
3  #include <malloc.h>
4  #define length sizeof(struct Animal)
5  struct Animal //define animal
6  {
7      int Aninum;
8      string Aniname;
9      string standard;
10     float price;
11     int count;
12     char unit;
13     char sex;
14     char alive;
15     struct Animal *next; //use linked list
16 };
17 struct Animal *head, *p1, *p2, *tail;

```

Fig. 4: Program code with definition of animal.

```

8      FILE *fp;
9      fp=fopen("Ani_info.txt","r"); //open file
10     p1=(struct Animal*)malloc(length);
11     fread(p1,length,1,fp);
12     p1=(struct Animal*)malloc(length);
13     fread(p1,length,1,fp);
14     p2->next=p1;
15     for(;;)
16     {
17         p2=p1;
18         p1=(struct Animal*)malloc(length);
19         fread(p1,length,1,fp);
20         p2->next=p1;
21         if(p1->number==0)
22             cout<<"this is end of file!"<<endl; break; }

```

Fig.5: Example of output to file with link list.

3.3. User interface

The tool, Visual C++ 6.0, has been implemented for the Windows environment. The output format and content can be controlled by giving different options for the tool. For example, the output can contain only the given fees compared to the maximum, or also the information of the animal. The exact requirements can also be checked from the released configuration file with teacher's comments. There is also an window interface to the pro-

gram, although most students use it from command line, just like the C++ compiler.

Fig.6. An example of LAS user interface.

4. Effectiveness and student feedback

The case teaching method has been used in C++ programming courses since the year 2008 and has been used by over 1500 students. Experiences and feedback on method usage have been collected by different kinds of questionnaires from students and in personal discussions with them over the years. The teaching personnel have also been interviewed about their experiences in using the method and in tutoring students who have used it. The discussion messages always provide good insight to the most difficult and the most interesting issues in the course. By the method, teaching covers almost knowledge points of C++ course (shown in Table 1).

We collected an anonymous survey from C++ course students after about 2/3 of the course was passed. Although the course discussion group had mostly negative comments about the method from students, the results of the survey showed the opposite, even though there were still some technical problems with the new method. The results are presented in Figure 7. Most of the students considered the method as a good solution and means to

support learning new issues in programming. However, it can be seen that many of them thought that the required measurement levels were too fastidious.

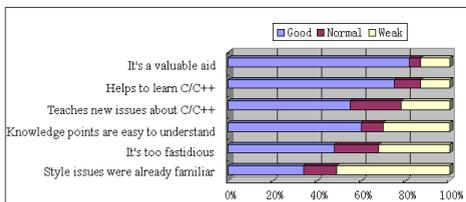


Fig.7. Student feedback on the method.

5. Conclusion

In this article, we have presented the method of case teaching in C++ programming courses. Student have built LAS system and worked them into automatically measurable program.

The technical solution seems to be adequately flexible for different education purposes. Based on one case of laboratory animal, some usability and technical issues that could be further developed. It has been shown that the method have specific reliability and code convention. LAS requirements with All technical advancements have been included in our courses, to support learning with best possible knowledge and practices.

6. Acknowledgements

Table 1: C/C++ Knowledge point used in system

	Define LA	Apply for	Return LA	Compute fee
If else and Switch case	✓	✓	✓	✓
Loops in C++	✓	✓	✓	✓
Functions in C++	✓	✓	✓	✓
Accessing Pointers	✓	✓	X	X
Structures in C++	✓	✓	✓	X
Storing Arrays	✓	✓	X	✓
Character Strings in C++	✓	✓	✓	X
Linked Lists	✓	✓	✓	✓
Inheritance	✓	X	X	X
C++ Class Design	✓	✓	X	X
C++ Formatted Output	X	✓	✓	✓
File I/O	X	✓	✓	X

This work was supported by the Technology Program of Guangdong(2011B061300081)(2012B091100334), and “Computational Science Innovative Research Team” program.

7. References

- [1] B.S.Bloom,et al. “Taxonomy of Educational Objectives: The Classification of Educational Goals”. *Handbook I*, New York, Longmans, Green, 1956.
- [2] A.N.Kumar,et al. “Model-based reasoning for domain modeling in a web-based intelligent tutoring system to help students learn to debug c++ programs”. *Intelligent Tutoring Systems*, pp. 792-801.2002.
- [3] K.Mutka, et al. “Supporting students in C++ programming courses with automatic program style assessment”. *Journal of Information Technology Education*, pp. 245-262,2004.
- [4] T. Jun,et al. “The Management Information System of Laboratory Animal Center(LACS) Based on ASP.Net,” *Proc. Of Itap 2012*, pp. 5-8, 2012.
- [5] R.M. Brune, et al. “A Three Dimensional Model of the Mouse at Embryonic Day 9,” *Developmental Biology*, vol. 216, pp. 457-468, 1999.