A New Teaching Pattern Based on PBL and Visual Programming in Computational Thinking Course

Peipei Gao
College of Computer and Control Engineering
Nankai University
Tianjin, China
watersky@nankai.edu.cn

Mingxiao Lu*
College of Computer and Control Engineering
Nankai University
Tianjin, China
lumx@nankai.edu.cn

Hong Zhao
College of Computer and Control Engineering
Nankai University
Tianjin, China
zhaoh@nankai.edu.cn

Min Li
College of Computer and Control Engineering
Nankai University
Tianjin, China
limintj@nankai.edu.cn

***Abstract*—As a fundamental and introductory course in undergraduate education, Computer Basis aims to cultivate the scientific computational thinking mode and the ability of solving problem for students who are not the computer majors. In this study, a new pedagogical method named PBL-VP combining the Problem-Based Learning (PBL) and the visual programming Blockly was applied to Computer Basis. This paper described a classical example about problem solving to explain the teaching process of the PBL-VP in detail. The process of problem solving was divided into several operable steps to gradually guide students to analyze problems, provide solutions and establish scientific logic thinking. Blockly was introduced to obscure the grammar of programming language and implement the algorithm visually. Finally, we evaluated the effectiveness of the PBL-VP teaching method on student learning by a contrast test. The empirical results proved that students in the class with PBL-VP earned higher final grades and the teaching method we proposed played a positive role in computer general education.**

***Keywords*—Problem-Based Learning (PBL), visual programing, computational thinking, problem solving, computer general education**

I. Introduction

The introductory computer course is an important part of the general education in universities, which shoulders the responsibility of cultivating students' computational thinking (CT)[1], and equipping students with the ability of solving practical problems with computational logic and tools. The traditional talent-cultivation models in universities have faced great challenge under the background of education construction of "new engineering" and "disciplinary integration" in China. Therefore, the teaching approaches and contents need to be constantly updated. Our university has applied blended learning (b-learning)[2] into Computer Basis, an introductory computer course opened for freshmen. The b-learning method, combining traditional off-line instruction with emerging online instruction through the online learning platforms such as Moso Teach, Tronclass and so on, has been proved to play a positive role on students' learning.

Given that the CT is too abstract to understand, learners without professional computer knowledge, are often confused by those obscure concepts, and difficultly apply the CT to solving practical problems. Furthermore, the teaching object of Computer Basis is mainly freshmen, most of whom do not have the knowledge foundation of algorithm, data structure, programming and so on. In order to cultivate students' capacity of solving problems using CT concretely, it is necessary to explore a suitable and easy-to-use programming tool to help learners implement algorithm.

Regarding the above problems, this study has explored a new pedagogical method named PBL-VP, which combines the Problem-Based Learning (PBL) [3-5] and the visual programming Blockly [6-8]. The purpose of this paper is to explain the teaching process of the PBL-VP in detail through a classical example. Finally, a comparative experiment was conducted to evaluate the teaching effect of the PBL-VP teaching pattern in this study.

II. Teaching Process of PBL-VP

PBL is a student-centered pedagogy in which students learn about a subject through the experience of solving an open-ended problem found in trigger material. Teachers play the role of guidance and cohesion throughout the PBL process.

In PBL-VP pattern, the teaching process is divided into the following six steps based on the problem orientation, as shown in Fig. 1.

- Problem selection and lead-in
- Problem abstraction and modeling
- Algorithm design
- Programming implementation
- Code generation

In this process, the teacher proposes a sub-question at each step, progressively guide students to analyze the question and think over the solutions, and help students form a complete understanding of the knowledge points about CT eventually. It is common that students probably get off track due to lack of professional knowledge during the learning process, thus continuous interaction with teachers is needed to revise students' cognition of the problem[9]. That is to say students generally need to go through the cycle of "trying-negating-trying again". In this paper, a classic mathematical problem-Map Navigation is taken for example to elaborate the critical steps. Further, a mathematical description of the problem is given in the PBL-VP process.

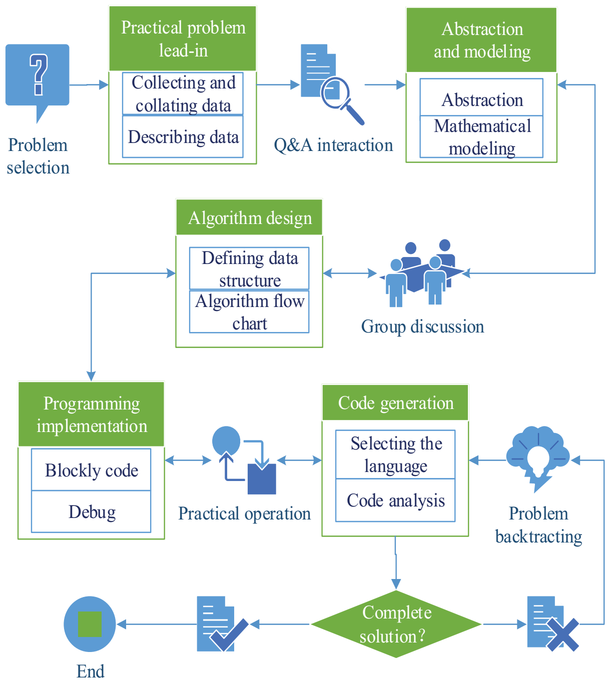

Fig. 1. The flowchart of PBL-VP teaching process.

A. Problem Selection and Lead-in

Problem selection is the primary part of the introduction phase in PBL-VP process, then followed by data collection, collation and description. Students can benefit from the selected problem with typicality and practicability, since empirical studies indicate that some kind of actual data in reality can make students understand the problem more effectively.

In the map navigation example, the selected problem is that students are required to use a mobile map to plan the optimal navigation path from the classroom to several target nodes selected on the map randomly, as shown in Fig.2.

The next task of students is collecting and organizing data regarding to the problem which is shown in Fig.3. To be specific, students need to find the path costs (which can be the cost time, the travelling distances or the spent fare) between pairs of points connected by roads, and then to calibrate these data on the map. According to the map in Fig.3, most students can instantly reach the correct answer: *D(1,2)=10*, when they are asked to calculate the best path from the classroom 1 to landmark 2.

Furthermore, the teacher can lead students to think about the question: "If the direct cost between node 1 and node 2 gets changed, will the optimal path change as well?" According to the map shown in Fig.4, the optimal path from 1 to 2 needs to be recalculated, and the answer can be easily drawn: *D(1.2)=D(1,3)+D(3.2)=11*. Through the rethinking, students are trained to shift their perspective from local to global and plan the optimal path of all nodes. This section aims to guide students to realize the importance of careful thinking and strong logic in problem solving.

B. Problem Abstraction and modeling

One of the CT's features is abstraction that can be used to solve multiple variations of an initial problem. The abstraction is a more complex symbolic process, and different from that in the fields of mathematics and physics. It needs to abstract the questions and the relevant data into digital forms that computers can understand.

In the PBL-VP process, the second step of problem solving is to guide students to abstract the problem into a quantitative mathematical problem, which can be calculated by computers. In our example, the map navigation problem is abstracted into finding the optimal paths between nodes in a graph, where the landmarks as the nodes, the roads as the edges, and the path cost as the weight of the edge. Fig.5 shows the structured mathematical model.

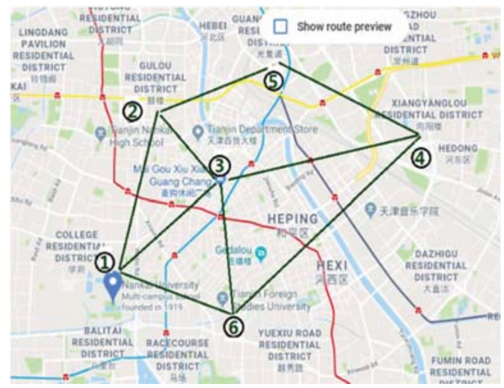

Fig. 2. The description of the selected problem

Fig. 3. The collection and collation of data

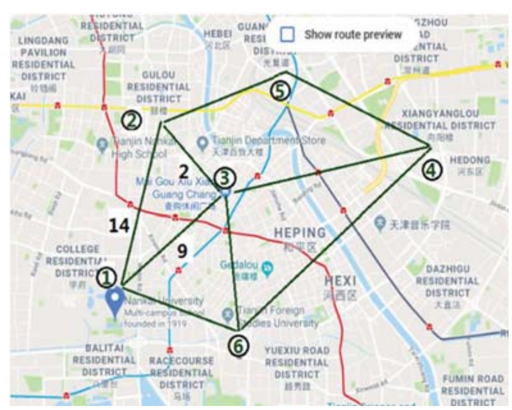

Fig. 4. Changing data in the problem

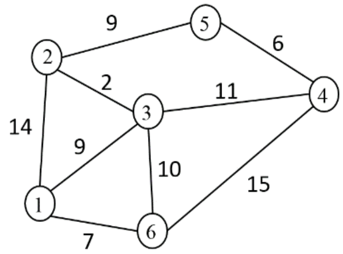

Fig. 5. The abstract model of problem.

Further, the mathematical description is given: for the weighted graph $G = (V, E)$, where V is a set of vertices, E is a set of edges, the starting point $v \in V$, any vertex $v_i \in V$ $(v_i \neq v)$, to solve $D(v, v_i)$, which represents the optimal path from v to v_i.

The teacher can further extend the optimal path problem from map navigation to other application fields in this section. Here are a few examples from concrete to abstract: How to find the most economical way to send a message from one computer to another on the Internet; How to develop an equipment renewal plan that minimizes the total cost of acquisition and maintenance of the equipment in a period of time. The purpose of the extension is to guide students to grasp the logic and method of problem solving in many fields.

C. Algorithm Design

After establishing the mathematical model and determining the data structure, students are instructed to design the algorithm that can solve the problem within a finite number of steps. The designed algorithm should be universal to the same kind of problems.

For students without professional programming experience, the map navigation problem can be simplified by only calculating the minimum cost of the optimal path, without recording the visited nodes in the path. Dijkstra's algorithm, a classic algorithm for finding the shortest paths between nodes in a graph, is properly simplified in this paper. Fig.6 shows the flow chart of the designed algorithm, where V is a set of vertices in the weighted graph $G = (V, E)$, and V is divided into the set S of visited nodes and the set U of the unvisited vertices, $u_i \in U$.

Algorithm design may be the most difficult but also the most important part of the whole teaching process of PBL-VP. Group discussion can play a positive role on helping stretch students' mind in this session. Students will realize the truth that a practical problem which is ever-changing in life can be accomplished in limited steps.

D. Algorithm Implementation using Blockly

In the PBL-VP teaching pattern, we introduced a visual programming tool Blockly to make the algorithm implement easier, with which learners can drag and rearrange visual blocks for common operations.

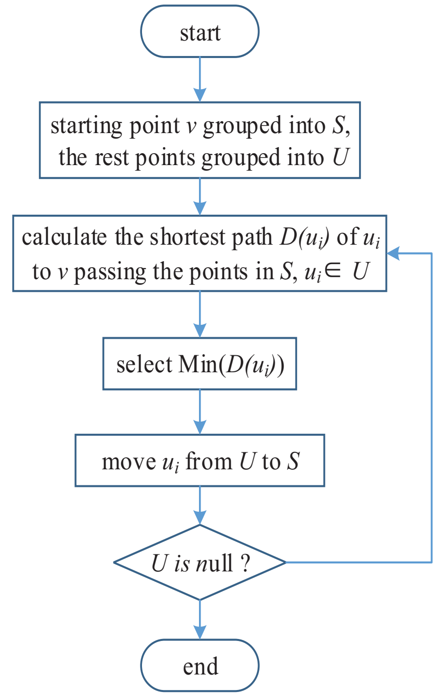

Fig. 6. The design of the algorithm flowchart

From a user's perspective, Blockly which builds code in an online, intuitive and visual way is also a library that adds a visual code editor to the web browsers and mobile apps. This editor uses interlocked graphic blocks to represent code concepts like variables, logical expressions, loops and so on. It allows users to apply programming principles without taking syntax into account, and it can generate grammatically correct code in the programming language users selected[8].

Cultivating students to implement the designed algorithm with Blockly should follow the principle from simplicity to complexity and from the part to the whole. For example, it is easier for students to begin with implementing the algorithm for finding the optimal path of the weighted graph with only 3 vertices in Fig.4. The process of implementing the algorithm with Blockly is shown in Fig.7.

E. Code Generation

Blockly can export the encapsulated blocks to several programming languages such as JavaScript, Python, PHP, Lua and Dart. After completing the algorithm implement with Blockly, students can choose a familiar programming language to export code and obtain the overall scheme of the problem to be solved. It is beneficial for students to generate code in several languages in turn and to compare the similarities and differences of their grammar. Students can understand the differences between "algorithm" and "program" through this process.

In this session, students are guided to realize that language is only a form of expression, and the way of thinking of human beings is the origin of everything.

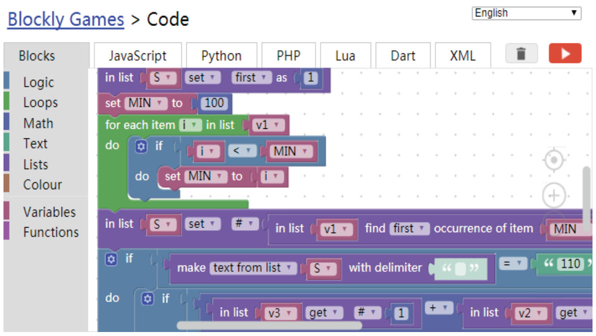

Fig. 7. The algorithm implementation using Blockly

Fig. 8. Generating Python code

III. Effectiveness Analysis

To determine the effects of the PBL-VP teaching method combining the Problem-Based Learning (PBL) and the visual programming Blockly on students' achievement, we conducted a comparative study in the 2018-2019 first-semester.

A. Participants

The respondents were 194 freshmen enrolled in two parallel classes of Computer Basis, one class with 99 students using the PBL-VP teaching method we proposed, and another with 95 students using the traditional teaching method. The two classes were both taught by the same instructor.

Computer Basis course is designed to give students a general understanding of the computational thinking, and to examine topics including software system, hardware system, computer networks, data storage and processing and some classic algorithm. Student success in achieving course outcomes was assessed through the final grade - a combination of 10 tests and 5 assignments which students were required to complete. Fig.9 showed the composition of the final grade for each student. T1-T10 represented the scores of 10 tests about CT algorithm arranged by the teacher. HW1-HW5 represented the scores of 5 assignments. The final score of each student was calculated from the weighted average of the T1-T10 and HW1-HW5, where each test score accounted for 5% and each homework score accounted for 10%.

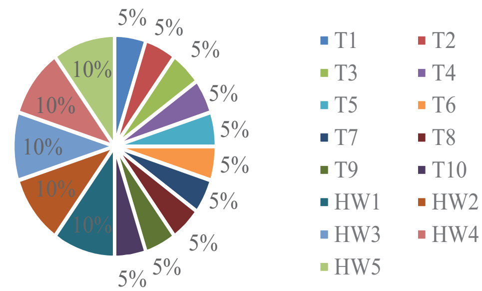

Fig. 9. The composition of the final grade

B. Data Analysis

In order to explore the differences in teaching effect between the PBL-VP and the traditional teaching method and ascertain the contribution of the PBL-VP to students' achievement, the independent sample t-test was conducted using the IBM Statistical Program for the Social Sciences (SPSS 22.0). We analyzed each student's final grade that was assessed through 10 tests and 5 assignments as the dependent variable in both courses. The data were obtained from the instructor's record of student grades and grading rubrics of the final grade. The independent variable for the study was the use of PBL-VP teaching method. The class with PBL-VP was called as class 1 and the class without PBL-VP was called as class 2.

In the independent sample t-test, 40 samples (N=40) were randomly selected from the population of each class. Table I showed the descriptive statistics of students' final grades in both classes. The mean of the final grade in class 1 was greater than the mean of class 2 (74.71>59.48). Students in PBL-VP class tended to have higher final scores than those in traditional class.

TABLE I. Group Statistics

	Class	N	Mean	Std. Deviation	Std. Error Mean
Final Grade	1	40	74.71	9.49	1.50
	2	40	59.48	13.49	2.13

TABLE II. Independent Samples Test

		Levene's Test for Equality of Variances		t-test for Equality of Means						
		F	Sig.	t	df	Sig.(2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference	
									Lower	Upper
Final Grade	Equal variances assumed	2.120	.149	5.837	78	.000	15.226	2.609	10.033	20.419
	Equal variances not assumed			5.837	70.011	.000	15.226	2.609	10.023	20.428

Table II showed the results of the independent sample t-test. The significance of Levene's test for equality of variances (0.149>0.05) indicated that the assumption of the homogeneity of variance of the tested variable (Final Grade) was true. The significance value of t-test for equality of means was 0.000 (0.000<0.05), which indicated that there were significant differences in the effects of the two teaching methods at 95% confidence. The PBL-VP teaching method has a significantly and positively impact on students' final grades.

IV. Conclusion

A new pedagogical method PBL-VP is introduced into courses about computational thinking, combining the Problem-Based Learning (PBL) and the visual programming Blockly based on the teaching and practice experiment of many years. In the PBL-VP pattern, the student is positioned to the subjectivity of learning and the teacher plays the role of organization and guidance. The introduction of the visual programming Blockly has greatly reduced the degree of difficulty for students in the process of code implementation. For students who are new to computational thinking, it is far more important to understand the ideas and methods of problem solving than to write a few pieces of code.

This paper explains the teaching process of the PBL-VP in detail through a classical example-map navigation problem solving. The process of problem solving is divided into several operable steps to gradually guide students to analyze problems, provide solutions and establish scientific logic thinking. In the teaching practice of PBL-VP, teachers and students can together experience the power of rigorous logic thinking, the effectiveness of thinking model in comprehension by analogy in practical applications, the importance of abstraction and decomposition in solving complex problems, and the relationship between ideas and expressions.

Furthermore, an empirical investigation is provided into the effectiveness of the PBL-VP teaching method through a comparative experiment. The results of the independent sample t-test show that the PBL-VP teaching method we proposed has a significantly and positively impact on students' final grades and is suitable for the learning of Computer Basis courses in university.

Acknowledgement

The authors gratefully acknowledge the research funding from the following programs:

- Research on Teaching Methods of Classroom Discussion Promoting Discipline Integration in Fundamental Computer Education and (Association of Fundamental Computing Education in Chinese Universities, 2018-AFCEC-288);
- Teaching reform of "Database Technology and Application" Based on O-AMAS(Nankai University, NKJG2019038).

References

[1] S. I. Swaid, "Bringing Computational Thinking to STEM Education," Procedia Manufacturing, vol. 3, pp. 3657-3662, 2015.

[2] H. Zhao, Y. Guo, K. Wang, and M. Lu, "The Determinants of Students' Academic Achievement in B-Leaming Environment: An Empirical Investigation," in 2018 13th International Conference on Computer Science & Education (ICCSE), 2018, pp. 1-5.

[3] H. Qingfeng, H. Xiaotao, and F. Cai, "Application of PBL Based Sandwich Teaching on C Programming," in 2018 13th International Conference on Computer Science & Education (ICCSE), 2018, pp. 1-3.

[4] B. R. Bessa, S. Santos, and B. J. Duarte, "Toward effectiveness and authenticity in PBL: A proposal based on a virtual learning environment in computing education," Computer Applications In Engineering Education, vol. 27, pp. 452-471, Mar 2019.

[5] M. Binshabaib, M. Shalabi, S. Mansour, L. Alageel, N. Alsuwidan, and S. S. Alharthi, "Role of problem-based learning in undergraduate dental education: a questionnaire-based study," Advances in medical education and practice, vol. 10, pp. 79-84, 2019 2019.

[6] S. Gorowski, R. Maguda, and P. Topa, "Multi-agent Systems Programmed Visually with Google Blockly," in Parallel Processing And Applied Mathematics. vol. 10778, R. Wyrzykowski, J. Dongarra, E. Deelman, and K. Karczewski, Eds., ed, 2018, pp. 476-484.

[7] S. Yamashita, M. Tsunoda, and T. Yokogawa, "Visual Programming Language for Model Checkers Based on Google Blockly," in Product-Focused Software Process Improvement. vol. 10611, M. Felderer, D. M. Fernandez, B. Turhan, M. Kalinowski, F. Sarro, and D. Winkler, Eds., ed, 2017, pp. 597-601.

[8] D. Galan, R. Heradio, L. de la Torre, S. Dormido, and F. Esquembre, "Conducting Online Lab Experiments with Blockly," IFAC-PapersOnLine, vol. 50, pp. 13474-13479, 2017/07/01/ 2017.

[9] B. Eichmann, F. Goldhammer, S. Greiff, L. Pucite, and J. Naumann, "The role of planning in complex problem solving," 2019.