The Development of an Innovative Programming Teaching Framework for Modifying Students' Maladaptive Learning Pattern

Tien-Chi Huang, *Member, IACSIT*, Yu Shu, Chia-Chen Chen, and Mu-Yen Chen

***Abstract*—Programming has traditionally been considered a fundamental subject in computer science and engineering domains. In educational contexts, however, about fifty percent of the students have trouble learning programming. The reason could be that their maladaptive cognition leads to maladaptive behavior, and there is no any assistant tool helps them reduce maladaptive learning to be self-directed. For the reason, in order to response the important policy direction of Taiwan technological and vocational education and strengthen students' practical programming skills, this study intends to develop an innovative programming teaching framework with appropriate teaching strategies for reducing students' maladaptive learning.**

***Index Terms*—Constructivism, self-directed learning, maladaptive learning pattern, topic maps.**

I. Introduction

Programming courses are basic requirements in some post-secondary information and engineering programs in Taiwan; this indicates how important and essential Programming courses are in the information and engineering fields of profession. However, despite the importance programming holds in technical education, it is often found in educational context that only 1/3 to 1/2 of the students are able to fully acquire materials of programming and put them to use. There are two main reasons: 1) the programs are updated and replaced too rapidly. When teaching programming, information and engineering faculties to through various types of program languages. If the students fail to learn the basics of these Programming courses, it is mostly because the teachers only focus on explaining the languages or the examples given in texts, but not letting students perform and apply the knowledge. It is boring to listen to language instruction all the time; moreover, advanced instructions are hard to understand, and the students do not know where to put them to use even if they understood them [1]. 2) The programming course dose not train the students on self-learning. Programming courses are different from theological courses; it values establishing environments for program development and understanding of program languages, and more importantly, it has the purpose to develop the ability to use the programming technologies to solve actual problems [2]. The students need to think hard to learning programming as same program executing results can be represented with different languages. If the teachers only demonstrate with examples in texts, it would make it difficult to motivate the students' self-learning and thinking, as examples in texts are easier than and hard to link directly to actual problems that need to be solved [1]. It can be seen from the two points above that programming is a subject that combines knowledge and action; not only do students need to understand the material, but actual performance is even more important. However, most students experience maladaptive cognition in their first stages of programming learning, resulting in maladaptive behaviors, and severely hindering the learning motivation and willingness in the following time. This research probes into problems of maladaptive cognition and maladaptive behaviors of students when learning programming, and discuss how to design apply educative educational frameworks that help the students in self-directed learning.

II. Theoretical Background

A. Maladaptive Learning Pattern in Programming

Maladaptive cognition is defined as "The maladaptive process of cognition and cognitive strategies of learners during learning [3]." This research discusses the three most common types of maladaptive cognition including surface strategies, learning disorganization, and distractibility [3], [4]. Surface strategies refers to the mindset of the learner to repeat, recite and practice the content of learning in order to store it in his or her short term memory. When learning programming, many students use surface strategies and only "memorize" program codes or instructional languages, never actually thinking about the process to solve problems [5]. In the other hand, learning disorganization refers to the difficulty for the learner to establish or maintain a well-structured and systematical method of studying [4]. One frequently encountered difficulty when teaching programming is that if the instructor did not write the program codes in advance, it is time consuming to writing it in case, and even more so when errors are found, as more time is needed to debug. However, if an example program written in advance is shown to the students, it is more difficult for these learners to learn about systems and relations from these dozens or hundreds of lines of program codes. In this type of situation, learning disorganization often affects the willingness and results of the learners in learning programming. The third type of maladaptive cognition is

Manuscript received April 10, 2013; revised June 20, 2013.

T. C. Huang and M. Y. Chen are with the Department of Information Management, National Taichung University of Science and Technology, Taiwan, R.O.C. (e-mail: tchuang@nutc.edu.tw, mychen@nutc.edu.tw).

Y. Shu is with the Taichung Shinmin Senior High School, Taiwan, R.O.C. (e-mail: h1257@shinmin.tc.eud.tw).

C. C. Chen is with the Department Management Information Systems, National Chung Hsing University, Taiwan, R.O.C. (e-mail: emily@thu.edu.tw).

DOI: 10.7763/IJIET.2013.V3.342

distractibility. It means the level of easiness that the learning process is distracted by task-irrelevant thoughts and unrelated foreign stimulation, and the concentration level is lowered. Many Programming courses are conducted in integrated development environments, IDEs. These development environments are usually provided to experienced programmers for quick program development. For new learners, the complicated IDE interface can make them unable to focus on learning problem solving and programming [1], resulting in distractibility related problems.

B. Applying Topic Maps into Teaching

Topic maps are invented in 1999 and earned the international standards of the number ISO 13250 in 2000. It combines the merits of traditional fields of indexing, library science and knowledge representation, and provides a set of functions that effectively systemizes knowledge, much like the GPS, in this vast and unsystematic world of information. With this system, it is easy to explore, reason, and solve great amounts of problems brought by disordered information [6]. Topic maps use XML as its standard language. This is because that XML has the features of cross-platform, mobility, structure, portability and reusability. After being published by W3C (World Wide Web Consortium) in February of 1998, W3C, Extensible Markup Language (XML) has quickly become the international standard of electronic file making, and fits well into the application of documents on the Internet.

In an online learning environment, the application of Topic maps brings a unified method of presentation for different learning resources. Not only can this present the relativity of knowledge within these resources and make knowledge guiding of online learning resources possible, but it also increases the feasibility of combining multiple archives of learning resources. In the previous studies, we have successfully implemented civic learning systems with the Topic maps techniques. The teacher used Topic maps to connect several concepts with ease; while students are able to construct their own knowledge from these connections [7], [8]. This research utilizes visual Topic map technology to design a set of programming education framework and applies it to Programming courses along with teaching strategies, hoping that with the aid of this framework, the maladaptive learning conditions of students when learning programming can be improved, and the goal of training self-directed learning can be achieved.

C. Experiential Learning Theory

Dewey’s experience theory states that real education comes from experience [9]. Following this concept, progressivism argues even more strongly that the traditional teacher centered form of education shall be eliminated and converted into student centered education [10]. When applied to actual teaching, this means lowering the ratio of instructions given by teachers in courses, raising the ratio of students taking parts in activities, lowering teacher-student ratio, raising the interaction time between each student and the instructor, shift the focus from textbooks to the students’ life experiences, value individual differences, move from the classroom to real life and the society, and shifting from learning through listening to learning through performing [9], [11].

Kolb extends from the educator Dewey’s theories and proposes an experience learning module [12]. The module combines experience learning, teaching, thinking and social experience, and is suitable for on-site application of education. It has from major elements: (1) concrete experience (CE), (2) reflective observation (RO), (3) logical analysis and abstract conceptualization (AC), and (4) active experimentation (AE) (see Fig. 1). The pursue of concrete experiences means when designing courses, the instructor should offer students as many actual performing learning activities to students as possible, or give students flexible learning time and space so that they can learn through their own experiences, instead of only explaining abstract course material. Reflective observation is to lead students to observe through experience based on the concept of reflection. The aim is to do more than knowing the surface of the experience or the descriptions of phenomena, but to also observe and think through these experiences from multiple angles. Later in the stage of logical analysis, the instructor leads the students to subjectively feel, objectively observe and systematically structure and analyze their learning experiences, and derive their concept of integration. Finally, students need to be able to make decisions and solve problems for their own learning.

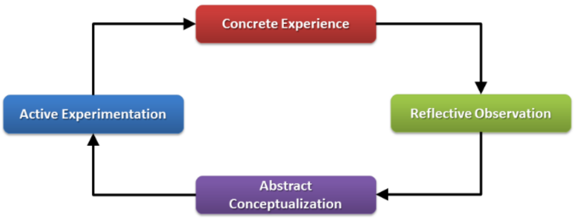


Fig. 1. Klob’s experiential learning cycle.

In this model, the instructor plays two roles: the designer and the facilitator. The former means the teachers has to obtain high creativity and develop open activities. The main focus of these activities is not to find the correct answer, but to explore new experience. Activities with diversity enable students of different skill levels to enjoy the process of learning. The facilitator means that the instructor should hold an open attitude towards all ways of problem solving, and encourage the students find problems and solve them from multiple angles and in multiple ways.

III. The Design of a Programming Teaching Framework

The researcher interviewed ten university freshmen who are also novices that have half a year or less in programming experience. It was found that the actual writing part is the post difficult aspect in programming to novices; they often have no clue on how to write the first program code after opening the software, or miss the target because they fail to see the big picture. Later, the difficulty of execution leads to the lack of interest and patience in practice, and ultimately the lack of learning motivation; some of them end up giving up programming. This reflects a blind point in past Programming courses: the learning start from imitating and copying program codes. With this spoon-fed way of education, the learners know how but do not know why. Therefore, they cannot effectively apply their programming knowledge in solving new questions, let along designing information systems. Therefore, according to the interview results and Kolb's experiential learning theory, we propose an experiential and thinking-based programming teaching framework (Fig. 2).

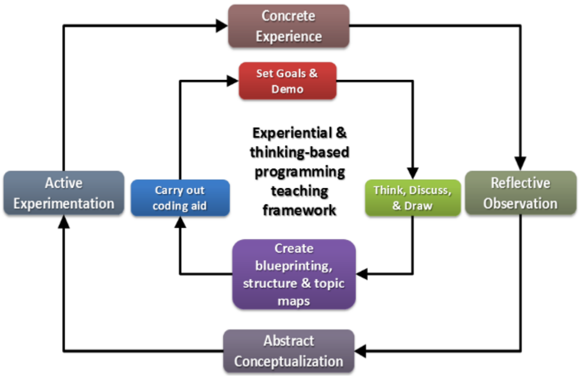

Concrete Experience
Set Goals & Demo
Experiential & thinking-based programming teaching framework
Active Experimentation
Carry out coding aid
Think, Discuss, & Draw
Reflective Observation
Create blueprinting, structure & topic maps
Abstract Conceptualization

Fig. 2. The proposed programming teaching framework.

This framework emphasizes building programming courses on the foundation of thought training, and states that programming course that lack logical thinking training is just skill training. It cannot truly raise the programming ability of students, and it is difficult for the students to perform learning transfer. As experts form knowledge with important thoughts or concepts, course designs should also be able to lead the students towards concept understanding.

Therefore, this study proposes an innovative teaching model for programming, named thinking-based programming teaching framework which is composed of four steps:

Step A. Set Goals and Demo: Teacher sets the purpose of a program and shows the final results to students. At the beginning of learning, the instructor can set a clear goal for the course. This might not have positive effects on expert learners, but it is very helpful for novices due to that novices usually fail to set specific goals systematically [13]. And the process of goal setting will greatly enhance the motivation of novices. Goal oriented teaching process is quite needed especially when students are facing abstract learning content. For example, make a simple website program that presents the current month and a corresponding photo of the season.

Step B. Think, Discuss, and Draw: At this stage, the students have to think based on the goal set by the instructor to determine what elements are needed, which conditions have priorities and which ones follow, and what resources and materials are needed to complete the goal. This is performed through group discussion or brainstorming sessions. From the discussion results, the students need to produce concept or flow charts of two modes. The whole process is to strengthen the thinking and planning abilities of students on learning programming, and the researcher decides to use the think-aloud method from the psychology field. Think-aloud method was originally developed by Newell and Simon [14] to study problem-solving strategies. For example, Olson, Duffy, & Mack asked their students to perform a learning mission of abstract making with thinking-aloud, and pronounce their whole process of cognition, the more detailed the better [15]. Through exposing the inner cognition, the learning strategies used by the students can be analyzed. The very strength of the method is that it is the closest possible way to get to the cognitive processes of readers. We extend the original think-aloud method to think-aloud by writing. Two concrete strategies are listed as follows:

Step B1. Establish infrastructure and concept maps: The students summarize and describe the target work, listing the basic elements needed to complete the target, for example: the four basic elements of webpage, time, words, and images.

Step B2. Sketch out and Create flow-charts: students have to come up with concrete sketch-outs based on the elements found in Step B1, and write down the details and relations of these elements in the format of flow-charts. For example: establishing a webpage project, capture the system time, how to present images in the webpage, automatically judge the correlations between images and the time, and the sizes of fonts and images. At this stage, the instructor starts to lead students using program codes, and help the students build up the link between program codes and elements through logical thinking. For example, the automatic judgment of the correlations between images and the time should use if-else or switch.

Step C. Create blueprinting, structure and Topic maps: the instructor uses the built up Topic map as a tool to explain the complete structure of the program work. The Topic map at this stage is like a blueprint of a building, and letting students understand the structure and relations within the whole program. For example: main program – determine month – MSDN Library.

Step D. Carry out coding aid: As the subjects are novices of programming, they might have the concepts of program structure, but still do not understand the details fully. Therefore, the researcher designed a set of learning tools to assist coding called Coding-Aid. First, the instructor makes the program codes of the target work into a [Fill in the blank selective program code hand out (see Appendix 1)], and the students have to fill in the blanked out program codes based on the structures of the target work and the previous two stages, while completing the coding mission on the computers.

IV. Discussion

TABLE I: Four Main Steps in the Proposed Teaching Framework and Corresponding Maladaptive Cognition for Modification

The notion of experiential learning	Learning activities	The targeting maladaptive cognitionfor modification
Describe purpose	Set Goals & Demo	Learning disorganization
Find resources	Think, Discuss & Draw	Surface strategies Learning disorganization
Confirm strategy	Blueprinting, Structure & Topic map	Surface strategies Learning disorganization
Evaluate learning outcome	Aid-Coding	Surface strategies Learning disorganization Distractibility

Different from how valued skills over thoughts and practice over understanding in traditional Programming courses, this structure utilizes the concept of constructivism, combining the cognition strategy teaching of metacognition and the technical assistance of Topic map. The novices are helped from the perspective of cognitive development to solve their issues of maladaptive learning, and raise their learning effectiveness, so the students have the courage to try program design (Table I).

The four major steps of the teaching structure are expected to lower different types of maladaptive learning cognitions. First, as the teachers set learning targets and present the executing results of the program, the novice students understand the purpose of learning, and establish an appropriate learning strategy. Secondly, through the process of think, discuss and draw, the learning initiative is back to the students, who are stimulated and lead to think about the elements to reach their goals through instincts and analysis. Different from traditional programming teaching, this teaching structure does not expose the program codes directly to students. At this stage, we lead the students to learn programming through deep cognitive strategies.

At the third stage, the researcher present the relativity between the program codes based on the instructor's Topic maps (as in Fig. 3) to strengthen organization abilities in programming. For example, Mayer offered many principles of multimedia material design, and advocated that the combination of words and images can produce multiple representations within learner, which is helpful for long term memory [16]. In other words, the learners can commence a deeper set of cognition strategies through this process.

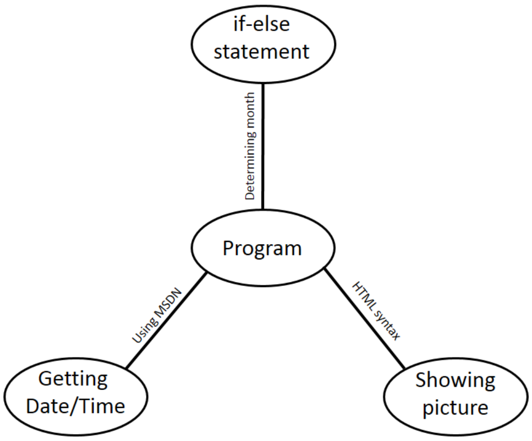


Fig. 3. A simple topic map.

Led by the Topic maps, the learners have sufficient prior knowledge towards the learning content, and the problem of not knowing where to start (learning disorganization) that learners face with programming can hopefully be improved.

In the finally stage, handouts assist students to think and complete their work in a decided program code framework. To novices, adequate assistance can raise the learning effects. As the assistance of the basic program codes are provided, the students no longer have to complete their work by memorizing codes. In the contrary, the cognition recourses can be used to solve the fill-in-the-blank questions of program codes, and the issues of students using only their memories. With appropriate guidance in program codes, when novices start writing programs, they no longer face the situation where they don't know where to start, and their concentrations can be stabilized through better guidance.

V. Conclusion

A. Implications for Innovative Teaching

After in-depth interviews with eight programming instructors, the researcher listed some problems faced by teaching: the first is with the traditional teaching methods, teaching programming has not become effective even with the advanced technology. Secondly, it is hard to effectively get a hold of how much the students understand the materials in class, and it is impossible to guide and debug every programming problem encountered by students due to time and man power limits. Finally, novices often hold a resisting attitude towards abstract program codes and give up learning without trying to practice or understand deeply.

Based on these points, the researcher promotes that regarding materials, besides the regular text, the instructors can also utilize the assistance of flow-charts and Topic maps in teaching, representing graphic relations between program codes and logical concepts. The needs of students with different learning styles can be fulfilled with a diverse set of materials.

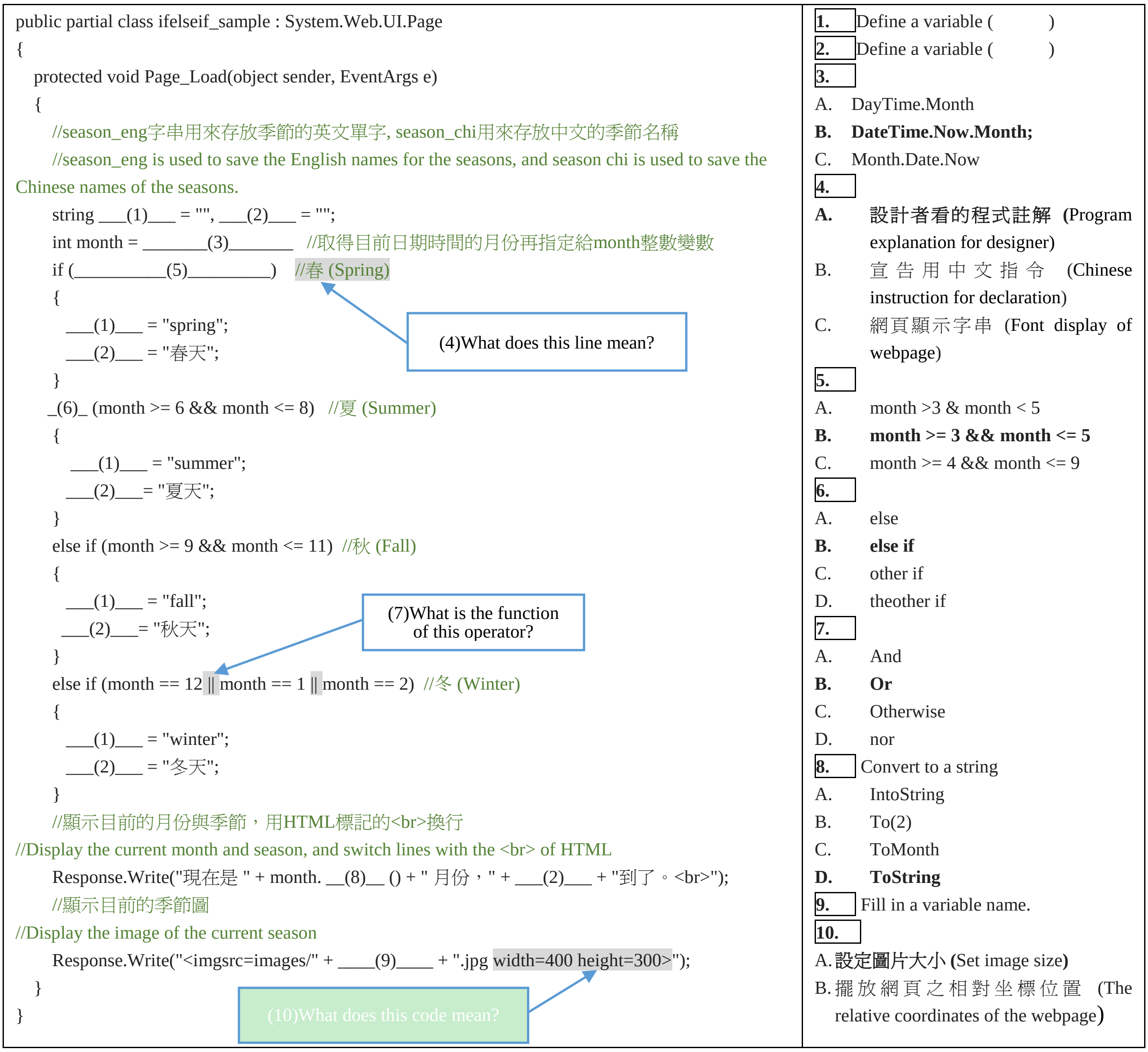
Regarding the curriculum of programming, the researcher hopes to break out of the behaviorism training module with repeated practices of the past, and value the cognitive process during learning. Therefore, the first three stages of the proposed teaching module are to lead students to engage in target oriented thinking and break down the logic of the programming mission, to teach the goal of meaningful learning. Also, this teaching module has assisting materials with result evaluation so students can complete the jobs on their own paste, achieving a certain degree of individualized teaching [13].

B. Implications for Constructivism Learning

It is clearly indicated in past literature that the biggest difference between experts and novices is that experts obtain conditionalized knowledge including the fields of application for the knowledge [17]. In other words, when the structure of program codes is meaningful, students with better learning abilities in programming can remember the program codes faster and easier than novices [18]. The proposed teaching framework coordinates with the cognition building process of the students, starting from setting goals, thinking process to finally completing the mission, and values the learner-centered spirit. Through this framework, we hope to effectively lower the threshold for novices entering the field of programming in actual classes, so that students can try to understand how to solve programming problems when they face them, instead of going straight the problem solving strategies. In this module, we stress that students can acquire fluency or be automatic through the learning process, and as a result, the requirements on concentrations of learning individuals can be lowered. This makes some aspects in the learning process easier, and they will have the ability to pay attention to other aspects of the leaning task. Abilities such as metacognition and self-directed learning are expected to be established and developed through this teaching framework.

Appendix

Appendix 1. A c# program demonstrates a webpage with the current month, season, and a corresponding picture.



```
public partial class ifelseif_sample : System.Web.UI.Page
{
   protected void Page_Load(object sender, EventArgs e)
   {
      //season_eng字串用來存放季節的英文單字, season_chi用來存放中文的季節名稱
      //season_eng is used to save the English names for the seasons, and season chi is used to save the
Chinese names of the seasons.
      string ___(1)___ = "", ___(2)___ = "";
      int month = _______(3)_______   //取得目前日期時間的月份再指定給month整數變數
      if (__________(5)_________)   //春 (Spring)
      {
        ___(1)___ = "spring";
        ___(2)___ = "春天";
      }
     _(6)_ (month >= 6 && month <= 8)   //夏 (Summer)
      {
         ___(1)___ = "summer";
        ___(2)___= "夏天";
      }
      else if (month >= 9 && month <= 11)  //秋 (Fall)
      {
        ___(1)___ = "fall";
       ___(2)___= "秋天";
      }
      else if (month == 12 || month == 1 || month == 2)  //冬 (Winter)
      {
        ___(1)___ = "winter";
        ___(2)___ = "冬天";
      }
      //顯示目前的月份與季節，用HTML標記的<br>換行
//Display the current month and season, and switch lines with the <br> of HTML
      Response.Write("現在是 " + month. __(8)__ () + " 月份，" + ___(2)___ + "到了。<br>");
      //顯示目前的季節圖
//Display the image of the current season
      Response.Write("<imgsrc=images/" + ____(9)____ + ".jpg width=400 height=300>");
   }
}
```

(4)What does this line mean?
(7)What is the function of this operator?
(10)What does this code mean?

1. Define a variable (   )

2. Define a variable (   )

3.

A. DayTime.Month

B. DateTime.Now.Month;

C. Month.Date.Now

4.

A. 設計者看的程式註解 (Program explanation for designer)

B. 宣告用中文指令 (Chinese instruction for declaration)

C. 網頁顯示字串 (Font display of webpage)

5.

A. month >3 & month < 5

B. month >= 3 && month <= 5

C. month >= 4 && month <= 9

6.

A. else

B. else if

C. other if

D. theother if

7.

A. And

B. Or

C. Otherwise

D. nor

8. Convert to a string

A. IntoString

B. To(2)

C. ToMonth

D. ToString

9. Fill in a variable name.

10.

A. 設定圖片大小 (Set image size)

B. 擺放網頁之相對坐標位置 (The relative coordinates of the webpage)

ACKNOWLEDGMENT

We wish to thank the National Science Council of the Republic of China for financially supporting this research under Contract No. NSC.

REFERENCES

[1] E. Costelloe, "Teaching programming the state of the art," CRITE Technical Report, Department of Computing, Institute of Technology Tallaght, Dublin, Ireland, 2004.

[2] M. Schollmeyer, "Computer programming in high school vs. college," *ACM SIGCSE Bulletin*, vol. 28, no. 1, pp. 378-382, March 1996.

[3] A. Kaplan and C. Midgley, "The effect of achievement goals: Does level of perceived academic competence make a difference?" *Contemporary Educational Psychology*, vol. 22, pp. 415-435, 1997.

[4] A. J. Elliot, "Approach and avoidance motivation and achievement goals," *Educational Psychologist*, vol. 34, no. 3, pp. 169-189, 1999.

[5] Y. J. Lai, "Online knowledge diagnosis and remedial system using concept maps model：A case study of program design of java," M. S. thesis, Dept. Info. Manage., Chaoyang Univ. of Tech., Taichung, Taiwan, 2009.

[6] S. Pepper, "Topic maps," *Encyclopedia of Library and Information Sciences*, 3rd ed., London: Taylor and Francis, 2010, pp. 5247-5259.

[7] T. C. Huang, "Creating a knowledge development model for blog-based learning," *International Journal of Information and Education on Technology*, vol. 1, no. 3, pp. 261-267, August. 2011.

[8] T. C. Huang and C. C. Chen, "Animating civic education: developing a knowledge navigation system using blogging and topic map technology," *Educational Technology and Society*, vol. 16, no. 1, pp. 79-92, 2013.

[9] J. Dewey, *Experience and education*, New York: Collier, 1963.

[10] E. Kline, D. W. Moore, and S. A. Moore, "Colonel Francis Parker and beginning reading instruction," *Reading Research and Instruction*, vol. 26, no. 3, pp. 141-150, Jan. 1987.

[11] W. Jeynes, "The relationship between parental involvement and urban secondary school student academic achievement: A meta-analysis," *Urban Education*, vol. 42, no. 1, pp. 82-110, 2007.

[12] D. A. Kolb, *Experiential learning: Experience as the source of learning and development*, Englewood Cliffs, NJ: Prentice Hall, pp. 21, 1984.

[13] B. J. Zimmerman, "Becoming a self-regulated learner: An overview," *Theory into Practice*, vol. 41, no. 2, pp. 64-70, 2002.

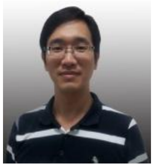
[14] A. Newell and H. A. Simon, *Human Problem Solving*, Prentice-Hall, 1972.

[15] G. M. Olson, S. A. Duffy, and R. L. Mack, "Thinking-Outloud as a Method for Studying Real-Time Comprehension Processes," in *New methods in the study of immediate processes in comprehension. Hillsdale*, D. E. Kieras and M. Just, Eds., NJ: Erlbaum, 1984, pp. 253-286.

[16] R. E. Mayer, "The promise of multimedia learning: using the same instructional design methods across different media," *Learning and Instruction*, vol. 13, pp. 125-139, 2003.

[17] R. Glaser, "Expert Knowledge and Process of Thinking," in *Enhancing Thinking Skills in Science and Mathematics*, *Hillsdale*, D. F. Halpern, Ed, NJ: Erlbaum, pp. 63-75, 1992.

[18] K. Ehrlich and E. Soloway, "An Empirical Investigation of the Tacit Plan Knowledge in Programming," in *Human Factors in Computer Systems*, J. C. Thomas and M. L. Schneider, Eds., Norwood, NJ: Ablex, 1984, pp. 113-134.



Tien-Chi Huang is currently an assistant professor at the Department of Information Management, National Taichung Institute of Technology in Taichung, Taiwan. He received the M.S. and Ph.D. degrees in computer engineering from National Sun Yat-Sen University and National Cheng Kung Universityin Taiwan in 2005 and 2009 respectively. Dr. Huang hosts several National Science Council projects about science education in Taiwan and his research interests focus on e-learning system development, knowledge management, user behavior analysis, and artificial intelligence.

Shu Yu received her master degree in educational psychology from National Cheng Kung University in July 2010, and begun her career as a citizenship teacher in August 2010 at Shinmin Senior high school in Taichung City, Taiwan. With interests in information technology and teaching, she does apply theories to her practical teaching and devote to integrate IT into instructions. She also believes that students and teachers are the leading role in class, even if the IT is widely adopted in class.

Chia-Chen Chen is an assistant professor of Management Information Systems at National Chung Hsing University, Taiwan. Her current research interests include RFID, context awareness, wireless and sensor network, e-learning, and smart living. Dr. Chen’s research is published or is forthcoming in Information Sciences, Computer and Education, Journal of Educational Technology & Society, The Electronic Library, International Journal of Mobile Communications, Expert Systems with Applications, International Journal of Information Technology and Management, and a number of national and international conference proceedings.

Mu-Yen Chen is an associate professor of Information Management at National Taichung University of Science and Technology, Taiwan. His current research interests include e-learning, e-Portfolio, RFID, artificial intelligent, financial engineering, and knowledge management. Dr. Chen’s research is published or is forthcoming in Information Sciences, Journal of Educational Technology & Society, Journal of Information Science, Computers and Mathematics with Applications, Neural Network World, Quantitative Finance, Expert Systems with Applications, Soft Computing, and a number of national and international conference proceedings.