Development of a Programming Learning System Based on a Question Generated strategy

Chih-Hung LAI, Pham-Duc THO *
*Department of Computer Science and Information Engineering, National Dong Hwa University,
Hualien, Taiwan, Republic of China*
*thopham@hvu.edu.vn

Abstract: In recent years, Computer programming has become a very important subject, and it is a basic literacy in the digital age. But learning programming skills is not an easy task as supported by many studies. On the other hand, many studies have concluded that student question generation has a positive effect on students' learning. However, few studies have geared toward supporting its use in programming classes. Therefore, this study aimed to develop a programming learning system named Peer-Interaction Programming Learning System based on a question generation strategy. The system was built by adding and intergrading learning assistance functions by system initiative into existing open-source Q&A system. In this paper, we reported the design of the system and its user interface, identified several related systems, discussed our motivation and underlying teaching philosophy.

Keywords: online learning system, student question generation, peer comment, question posing

1. Introduction

Programming is a complex activity and Jenkins (Jenkins, 2002) analyzed a number of factors that could contribute to the difficulty of programming. Some of the issues contributed to this obstacle such as : the teaching methods employed by the instructor, the difficult nature of computer programming (Matthíasdóttir, 2006), the study methods, abilities and attitudes employed by the student (Gomes & Mendes, 2007), the nature of the art of programming, the lack of prior knowledge of novice students, and the psychological influence that the student suffered from society (Jenkins, 2002). From past studies, the learning benefits of student question generation (SQG) have been well recognized and also attached with peer-comment feature (Denny, Luxton-Reilly, & Hamer, 2008; Denny, Luxton-Reilly, Tempero, & Hendrickx, 2011; Funabiki, Korenaga, Nakanishi, & Watanabe, 2013; Rhind & Pettigrew, 2012; Fu-Yun Yu, Liu, & Chan, 2005). There are many advantages of SQG based system, such as enhancing learning motivation, achievement, etc.

However, to the best of our knowledge, most of the existing researches have not yet focused on, or few to apply SQG and also peer comment in the support programming learning. Denny and Luxton-Reilly created CodeWrite (Denny et al., 2011) and StudySieve (Luxton-Reilly, Denny, Plimmer, & Bertinshaw, 2011) which is aimed at helping students learning to programming by using question posing and peer comment. Both CodeWrite and StudySieve have a significant effect in support students learn programming (Denny et al., 2011; Luxton-Reilly et al., 2011) but only focus on free-response domain which limit students who wanted to create other types of questions. In addition, only after the solution compiles and passes all the test cases are the solutions submitted by other students revealed (Denny et al., 2011) makes normal students harder to solve the difficult questions without any hints or supports.

That is, there are research gap for apply SQG for supporting its use in programming classes with more types of question and support mechanism to make students more convenience in questioning and answering in the system. Therefore, this study aims to develop a system to fulfill the gaps which remain in previous research.

2. Question generation and learning programming

Question generation is the activity in which students generate exam questions based on the reading content. The question types included multiple choice, matching, short answer, true-false, and fill-in-the-blank formats or word puzzles (Barak & Rafaeli, 2004; Wilson, 2004; F. Y. Yu, 2011). When students generate questions, students need to concentrate on the important information in their texts, and then provide correct answers and distracters (Yu, Liu & Chan, 2002). Students' abilities to explain why the answer options they create are correct or incorrect, reveal whether students really understand the reading materials or not (Fellenz, 2004). And through question generation, teachers could identify students' reading problems and thus provided adaptive instruction (Lan & Lin, 2011; Yeh & Lai, 2012).

A number of web-based learning systems with a focus on student question-generation have been developed such as QPPA (F.-Y. Yu, Liu, & Chan, 2002), POP-B and POP-C (Nakano, Hirashima, & Takeuchi, 2002), Question Sharing, Information and Assessment system (QSIA) (Barak & Rafaeli, 2004), Multiple Choice Items Development Assignment (MCIDA) (Fellenz, 2004), Asking a Good Question (AGQ) (Chang, Huang, Tung, & Chan, 2005), Questionbank (Draaijer & Boter, 2005), ExamNet (Wilson, 2004), Concerto (Hazeyama & Hirai, 2007), PeerWise (Denny et al., 2008), Question-Authoring and Reasoning Knowledge System (QuARKS) (F. Y. Yu, 2009), CodeWrite (Denny et al., 2011), Question-Posing Indicators Service (QPIS) (Lan & Lin, 2011), StudySieve (Luxton-Reilly et al., 2011) and Active S-Quiz (Hayashi et al., 2015). While most such systems are domain-independent, allow students to generate different types of questions with multimedia, and support anonymous interactions, but few are geared toward supporting its use in programming classes. Denny et al. presented and evaluated a Web based on tool providing drill and practice supports for Java programming called CodeWrite (Denny et al., 2011), where students are responsible for developing exercises that are shared among classmates. Students develop associated test cases for the exercises they author, and receive immediate feedback on their own code when solving exercises created by their peers. CodeWrite and StudySieve (Luxton-Reilly et al., 2011) are actually extended version of PeerWise (Denny et al., 2008) which only allows students to generate multiple choice questions. Both developed to extend student-generated questions to the free-response domain and helped students to write, solve and assess computer programming codes with regard to the difficulty of multiple choice questions in addressing higher-order cognitive skills. StudySieve is now discontinued but CodeWrite is still being used in some Computer Science courses.

As we described in the introduction, very little research has been done on SQG in the area of programming. Current systems typically support a limited number of question types, are hard to interact with, and are not effective in helping students learn by themselves. Besides, they lack support for student anonymity. Ballantyne suggested that students should remain anonymous to alleviate student concerns over bias and unfair marking (Ballantyne, Hughes, & Mylonas, 2002). In some institutions, anonymity may be a statutory requirement (Bhalerao & Ward, 2001) when working with student information.

On the other hand, teachers need to use the systems with other e-learning systems to manage learning resources, grade students exercise, which are heavy work for teachers.

3. Description of the system

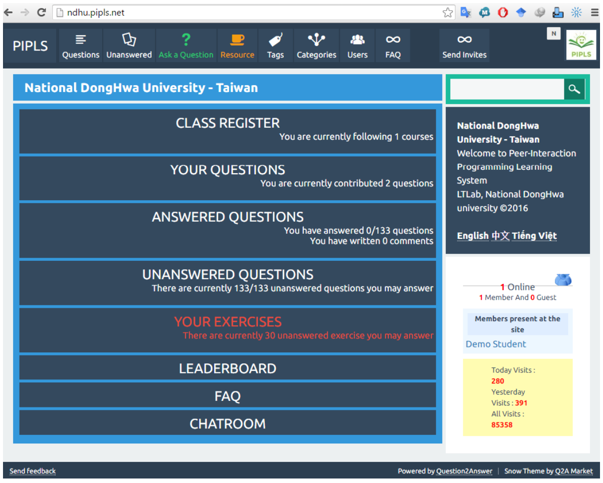
Peer-Interaction Programming Learning System (PIPLS) is a web-based system based on the open source Question2Answer system (Greenspan & Contributors, 2016) with further development and customization. PIPLS is designed aiming at fill some gaps which are remains from previous systems:

- Allow students to choose to use their real name, their nickname or anonymous.
- Support more question types: multiple choice, short answer, true-false, fill-in-the-blank, coding and essay with automatic judge or semi-automatic which is significant benefit for staff.
- Make the peer-interactive process more accurate and easy for students whom nowadays familiar with many social networking sites.
- Allow learning content to be integrated into courses.

PIPLS supports student-generated multiple types of questions, included free-response, multiple-choices, fill in the blanks, and true-false questions. In this system, the students can discuss with each other by asking and answering the questions. The teachers can set questions, share the resources of learning and develop effectiveness of class management.

After logging in, the main menu (Fig. 1) is divided into some sections entitled: "Class register", "Your questions", "Answered questions", "Unanswered questions", "Your exercises", "Leader board", "FAQ" and "Chatroom". On the main page, students can find some quick statistic information about their progress: courses they are following, contributed questions, answered questions, unanswered questions, and and remaining exercises..

Besides those sections, there are many features and functions that we developed to help students and teachers. The system was developed on the basis of the Question2Answer system, so it has inherited all features of Question2Answer (Greenspan & Contributors, 2016). Due to the limit of this paper, the role of some of these sections is described next.



National DongHwa University - Taiwan

CLASS REGISTER
You are currently following 1 courses

YOUR QUESTIONS
You are currently contributed 2 questions

ANSWERED QUESTIONS
You have answered 0/133 questions
You have written 0 comments

UNANSWERED QUESTIONS
There are currently 133/133 unanswered questions you may answer

YOUR EXERCISES
There are currently 30 unanswered exercise you may answer

LEADERBOARD

FAQ

CHATROOM

National DongHwa University - Taiwan
Welcome to Peer-Interaction Programming Learning System
LTLab, National DongHwa university ©2016

English 中文 Tiếng Việt

1 Online
1 Member And 0 Guest

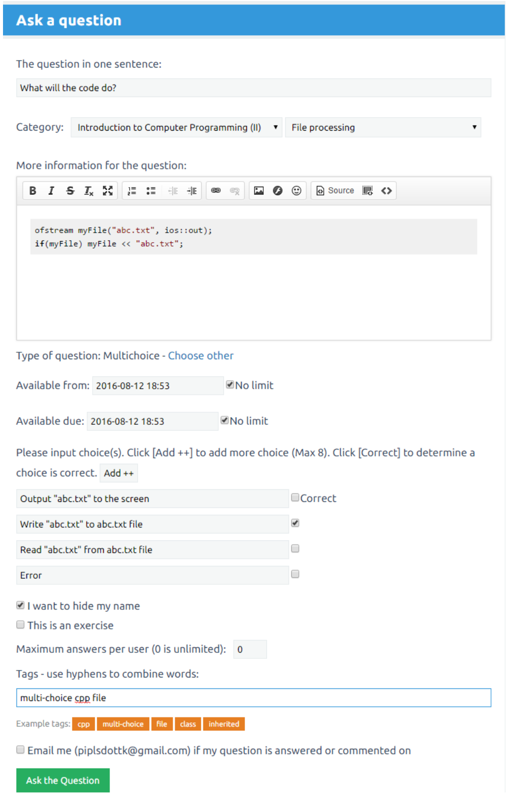
Members present at the site
Demo Student

Today Visits : 280
Yesterday Visits : 391
All Visits : 85358

Send feedback
Powered by Question2Answer | Snow Theme by Q2A Market

Figure 1. The main page of PIPLS.

3.1 *Composing a question*



Ask a question

The question in one sentence:
What will the code do?

Category: Introduction to Computer Programming (II) ▾ File processing ▾

More information for the question:

```
ofstream myFile("abc.txt", ios::out);
if(myFile) myFile << "abc.txt";
```

Type of question: Multichoice - Choose other

Available from: 2016-08-12 18:53 ☑No limit

Available due: 2016-08-12 18:53 ☑No limit

Please input choice(s). Click [Add ++] to add more choice (Max 8). Click [Correct] to determine a choice is correct. Add ++

Output "abc.txt" to the screen ☐Correct
Write "abc.txt" to abc.txt file ☑
Read "abc.txt" from abc.txt file ☐
Error ☐

☑ I want to hide my name
☐ This is an exercise

Maximum answers per user (0 is unlimited): 0

Tags - use hyphens to combine words:
multi-choice cpp file

Example tags: cpp multi-choice file class inherited

☐ Email me (piplsdottk@gmail.com) if my question is answered or commented on

Ask the Question

Figure 2. Composing a new question in PIPLS.

This section (fig. 2) allows a student to compose a question. The student was asked to chose the type of question before they can reach this section. PIPLS is designed to support multiple question types, including free-response, multiple-choices, fill in the blanks, and true-false questions.

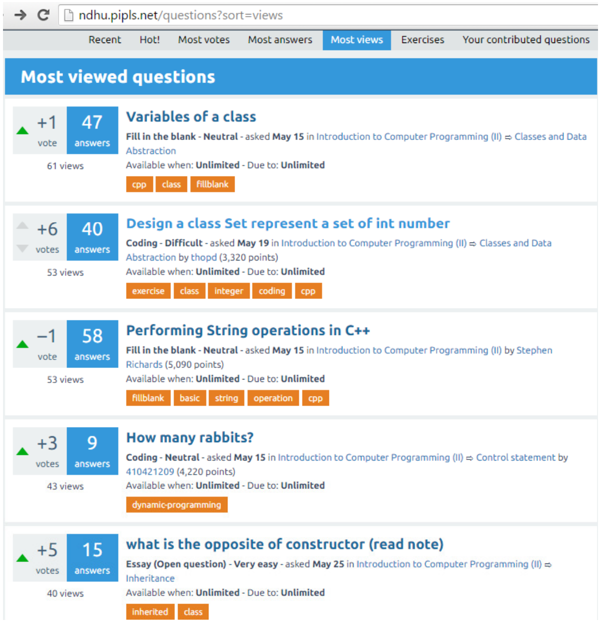
Inherited from Question2Answer (Greenspan & Contributors, 2016), the question title must be provided with detailed information and embed multimedia, links, … To help students find and organize relevant questions, the questions may be tagged with appropriate topics by the author.

Original Question2Answer only provided free-response question so we developed additional fields applied for different types of question. Fig.2 illustrates how a multiple-choice question is defined.

Besides important information come with types of question, we also developed available time for the question, anonymous feature, mark question as exercise specifically for teacher, and the maximum answer per user allowed for the question.

3.2 *Viewing all questions in course*

Figure 3 shows how the list of questions in the course is displayed to students. The design is inherited from Question2Answer, the questions contained in the course are displayed in a single paginated list that can be sorted and filtered by various criteria: Recent, hot, most votes, most answers, most views, and our extended filter include exercises, your contributed questions, difficulty (based on peer evaluations), unanswered or answered questions.


ndhu.pipls.net/questions?sort=views
Recent
Hot!
Most votes
Most answers
Most views
Exercises
Your contributed questions

Most viewed questions

+1 vote 47 answers 61 views
Variables of a class
Fill in the blank - Neutral - asked May 15 in Introduction to Computer Programming (II) ⇨ Classes and Data Abstraction
Available when: Unlimited - Due to: Unlimited
cpp class fillblank

+6 votes 40 answers 53 views
Design a class Set represent a set of int number
Coding - Difficult - asked May 19 in Introduction to Computer Programming (II) ⇨ Classes and Data Abstraction by thopd (3,320 points)
Available when: Unlimited - Due to: Unlimited
exercise class integer coding cpp

−1 vote 58 answers 53 views
Performing String operations in C++
Fill in the blank - Neutral - asked May 15 in Introduction to Computer Programming (II) by Stephen Richards (5,090 points)
Available when: Unlimited - Due to: Unlimited
fillblank basic string operation cpp

+3 votes 9 answers 43 views
How many rabbits?
Coding - Neutral - asked May 15 in Introduction to Computer Programming (II) ⇨ Control statement by 410421209 (4,220 points)
Available when: Unlimited - Due to: Unlimited
dynamic-programming

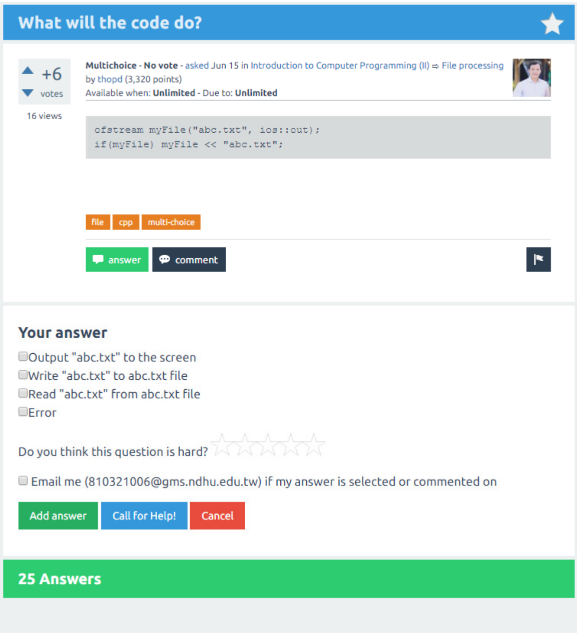
+5 votes 15 answers 40 views
what is the opposite of constructor (read note)
Essay (Open question) - Very easy - asked May 25 in Introduction to Computer Programming (II) ⇨ Inheritance
Available when: Unlimited - Due to: Unlimited
inherited class

Figure 3. The All Questions List (Sorted by Views) in PIPLS.

3.3 *Answering questions and viewing peer answers*

In question-generation systems designed for educational purposes, students can use the questions for drilling and practicing, and as such, any answers should remain hidden until the student attempts to answer the question (Luxton-Reilly et al., 2011). In PIPLS, answers are revealed only after the student submitted the correct answer for automatically judged questions, or the answer for an essay question.

We enhanced PIPLS by including many functions of traditional SQG systems, include “call for help” function. When student cannot figure out the answer, they don’t need to give up or require help by giving some comment and wait. They can keep thinking straight without losing time by “call for help”. This function will allow student reading the answers from other classmates. Then the student need to decide which answer is correct and complete their own answer.



What will the code do?

+6 votes

16 views

Multichoice - **No vote** - asked Jun 15 in Introduction to Computer Programming (II) ⇨ File processing
by thopd (3,320 points)
Available when: **Unlimited** - Due to: **Unlimited**

```
ofstream myFile("abc.txt", ios::out);
if(myFile) myFile << "abc.txt";
```

Your answer

□ Output "abc.txt" to the screen
□ Write "abc.txt" to abc.txt file
□ Read "abc.txt" from abc.txt file
□ Error

Do you think this question is hard? ☆☆☆☆☆

□ Email me (810321006@gms.ndhu.edu.tw) if my answer is selected or commented on

Figure 4. The Questions Answer interface (Multiple-choices type) in PIPLS.

3.4 *Evaluating questions and answer*

PIPLS support multiple-choice, true/false, fill in the blanks, and coding questions so it can automatically generate feedback for students who answer questions, by reporting whether the answer is correct (by percentage) or not. The student needs to submit the correct answer in order to see other answers. And numbers of answer are limited by author (or not) in the question composing interface (Fig. 2).

In the PIPLS, we have two types of free-response question: essay questions and coding question. Essay questions needs author or teacher to judge but coding question is automatically judged. PIPLS now supporting C, C++, Java and Pascal in auto-judge function. Coding questions are not only judged automatically, teacher also can re-judge the answer in case the machine cannot or if teacher want to give some bonus points for the good solution.

Currently, whenever a student submits an answer to a question, they can evaluate the question by assigning a rating (on a 1–5 scale) to the difficulty level of the question (Fig. 4).

Students also have the opportunity to write formative feedback to the question author, thanks to the comment feature of original Question2Answer which is visible to all users, and can agree or disagree with other feedback provided by their peers by voting feature. When the others' answer is visible, students can give comment and also voting in others' answer.

4. Conclusion and future work

We developed the system based on previous research, and focused on supporting students to learn Programming. We also extended the type of question generation (multiple-choice, true/false, fill in the blanks, essay and coding) and developed many additional feature. We hope to give more support to students when comparing with other systems which also support programming learning.

In future, we will plan to enhance the existed systems' functions and evaluate the impact of the tool on students' performance. We also intend to study the nature and quality of the artefacts (questions, answers and feedback) produced by students.

References

Ballantyne, R., Hughes, K., & Mylonas, A. (2002). Developing Procedures for Implementing Peer Assessment in Large Classes Using an Action Research Process. *Assessment & Evaluation in Higher Education, 27*(5), 427-441. doi:10.1080/0260293022000009302

Barak, M., & Rafaeli, S. (2004). On-line question-posing and peer-assessment as means for web-based knowledge sharing in learning. *International Journal of Human-Computer Studies, 61*(1), 84-103. doi:http://dx.doi.org/10.1016/j.ijhcs.2003.12.005

Bhalerao, A., & Ward, A. (2001). Towards electronically assisted a case study. *Research in Learning Technology, 9*(1).

Chang, S.-B., Huang, H.-M., Tung, K.-J., & Chan, T.-W. (2005). *AGQ: a model of student question generation supported by one-on-one educational computing.* Paper presented at the Proceedings of th 2005 conference on Computer support for collaborative learning: learning 2005: the next 10 years!

Denny, P., Luxton-Reilly, A., & Hamer, J. (2008). *The PeerWise system of student contributed assessment questions*. Paper presented at the Proceedings of the tenth conference on Australasian computing education - Volume 78, Wollongong, NSW, Australia.

Denny, P., Luxton-Reilly, A., Tempero, E., & Hendrickx, J. (2011). *CodeWrite: supporting student-driven practice of java*. Paper presented at the Proceedings of the 42nd ACM technical symposium on Computer science education, Dallas, TX, USA.

Draaijer, S., & Boter, J. (2005). Questionbank: computer supported self-questioning.

Fellenz, M. R. (2004). Using assessment to support higher level learning: the multiple choice item development assignment. *Assessment & Evaluation in Higher Education, 29*(6), 703-719. doi:10.1080/0260293042000227245

Funabiki, N., Korenaga, Y., Nakanishi, T., & Watanabe, K. (2013, 26-29 Aug. 2013). *An extension of fill-in-the-blank problem function in Java programming learning assistant system.* Paper presented at the Humanitarian Technology Conference (R10-HTC), 2013 IEEE Region 10.

Gomes, A., & Mendes, A. J. (2007). *Learning to program-difficulties and solutions*.

Greenspan, G., & Contributors. (2016). Question2Answer (Version 1.7.4). Retrieved from http://www.question2answer.org/

Hayashi, T., Nakagawa, S., Kishimoto, T., Hirai, Y., Ura, K., Yaegashi, R., . . . Tarumi, H. (2015, 1-3 June 2015). *Active S-quiz: An intelligent educational system for basic knowledge learning by question-posing.* Paper presented at the Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), 2015 16th IEEE/ACIS International Conference on.

Hazeyama, A., & Hirai, Y. (2007). *Concerto II: A learning community support system based on question-posing.* Paper presented at the Seventh IEEE International Conference on Advanced Learning Technologies (ICALT 2007).

Jenkins, T. (2002). On the difficulty of learning to program. *Proceedings of the 3rd Annual HEA Conference for the ICS Learning and Teaching Support Network, 1-8.*

Lan, Y. F., & Lin, P. C. (2011). Evaluation and improvement of student's question-posing ability in a web-based learning environment. *Australasian Journal of Educational Technology, 27*(4), 581-599.

Luxton-Reilly, A., Denny, P., Plimmer, B., & Bertinshaw, D. (2011). *Supporting student-generated free-response questions*. Paper presented at the Proceedings of the 16th annual joint conference on Innovation and technology in computer science education, Darmstadt, Germany.

Matthíasdóttir, Á. (2006). *How to teach programming languages to novice students? Lecturing or not.* Paper presented at the International Conference on Computer Systems and Technologies-CompSysTech.

Nakano, A., Hirashima, T., & Takeuchi, A. (2002, 3-6 Dec. 2002). *A support environment for learning by describing problem map.* Paper presented at the Computers in Education, 2002. Proceedings. International Conference on.

Rhind, S. M., & Pettigrew, G. W. (2012). Peer Generation of Multiple-Choice Questions: Student Engagement and Experiences. *Journal of Veterinary Medical Education, 39*(4), 375-379. doi:10.3138/jvme.0512-043R

Wilson, E. V. (2004). ExamNet asynchronous learning network: augmenting face-to-face courses with student-developed exam questions. *Computers & Education, 42*(1), 87-107.

Yeh, H. C., & Lai, P. Y. (2012). Implementing online question generation to foster reading comprehension. *Australasian Journal of Educational Technology, 28*(7), 1152-1175.

Yu, F.-Y., Liu, Y.-H., & Chan, T.-W. (2002, 3-6 Dec. 2002). *The efficacy of a web-based domain independent question-posing and peer assessment learning system.* Paper presented at the Computers in Education, 2002. Proceedings. International Conference on.

Yu, F. Y. (2009). Scaffolding student-generated questions: Design and development of a customizable online learning system. *Computers in Human Behavior, 25*(5), 1129-1138. doi:10.1016/j.chb.2009.05.002

Yu, F. Y. (2011). Multiple peer-assessment modes to augment online student question-generation processes. *Computers & Education, 56*(2), 484-494. doi:10.1016/j.compedu.2010.08.025

Yu, F. Y., Liu, Y. H., & Chan, T. W. (2005). A web-based learning system for question-posing and peer assessment. *Innovations in Education and Teaching International, 42*(4), 337-348. doi:10.1080/14703290500062557