

# INTERACTIVE ASSISTANCE FOR TEACHING C++ ARRAY TOPIC TO NOVICE LEARNERS

Naemah Abdul Wahab<sup>1</sup>, Jamal Othman<sup>2</sup> and Saiful Nizam Warris<sup>3</sup>

<sup>1,2,3</sup> *Department of Computer and Mathematical Sciences, Universiti Teknologi MARA  
Cawangan Pulau Pinang, 13500 Permatang Pauh, Pulau Pinang, MALAYSIA.*

<sup>1</sup>*naema586@ppinang.uitm.edu.my;* <sup>2</sup>*jamalothman@ppinang.uitm.edu.my;*

<sup>3</sup>*saifulwar@ppinang.uitm.edu.my*

## ABSTRACT

*Previous research has suggested that teaching and learning computer programming to novice learners; particularly engineering students is a very challenging task. The complex nature of programming subject, lack of problem solving skills and misunderstanding of programming syntax and semantics are a few of the possible reasons of learner's negative stereotypes and difficulties with programming subject. Many researchers found that engineering students can be categorized under the "visual learner" side of the Felder-Silverman Learning Styles Model. However, the mismatches that exist between the learner's learning styles and traditional text-based programming language and auditory teaching approach contribute to misunderstanding, strong dislike and poor performance of students in programming subject. As programming novices prefer to comprehend visually, this study investigates the effectiveness of using interactive teaching aid as a tool in improving programming understanding specifically designed for novice learners, focused only in array topic in C++ introductory course. Initial results show that the learning performance and program comprehension on array C++ topic improved after using the teaching aid.*

**Keywords:** teaching aid in C++; C++ array topic; programming in C++ for novice learners.

## 1. INTRODUCTION

At present, computer science has been applied in various fields of study, including engineering area. Realizing its importance, many branches of engineering studies require their students to take introductory programming as one of the compulsory courses in their study plan. One of the fundamental programming languages that act as basis in learning advanced computer program is C++.

Basic programming languages are taught using text-based language that requires student to write programming statements in an editor. The completed program will be compiled and interpreted using programming software. However, learning programming language, specifically for novice learners is not an easy task. Rist (1996) as stated by Abid (2011) identified that novice learners are lack in professional and expert skills in programming language.

Traditional auditory lecturing and text-based teaching materials are the main reasons of learner's lack of ability to comprehend computer programming. Many previous research studies including Bucks and Oakes (2008) ruled out that engineering students fall on the "visual learners" category of Felder-Silverman Learning Styles Model. In addition, Myers (1988) pointed out that some difficult programming concepts are better explained visually through pictures, diagrams and etc. Chang (2011) mentioned that the application of animation, multimedia and visualization for the development of learning tools are used in programming courses to enhance programming comprehension.

The main problem of learning programming among engineering students is understanding the logic of the problem statements. Misunderstanding the problem statement leads to incorrect implementation of programming structures or methods. Moreover, the level of logical thinking among engineering students is still considered as slightly average. Other problems of learning programming as stated by Abdel Rahman, et. al. (2016), are multi different accent of English language among programming lecturers, lack of attractive techniques in teaching approaches and insufficient appropriate programming materials. Therefore, interactive teaching aid is used as a tool to comprehend programming understanding.

For that reason, this study created a teaching aid focused on array topic in C++ introductory course specially designed for novice learners as a tool to improve program understanding, specifically the array data structure concept. The array topic has been chosen for this procedure in order to measure the learning ability of engineering students. Array topic is a core or backbone of all the previous topics that have been taught by the lecturers. The basic control structures such as selection or repetition and structured programming such as function or procedural concept applies the implementation of array data structures as an important element in C++ programming. If the students are able to comprehend and grab easily the concepts of array topic, we can conclude that the students have successfully understand the entire topic of programming fundamentals. Failure to understand the array concepts can possibly imply that the students do not grab the understanding of the earlier topics such as selection, repetition and functions.

Figure 1 illustrate the framework of conducting the lectures in the programming classes. The framework in Figure 1 shows that the students should be able to understand clearly the previous topic before they can proceed to the next topics. Students are advisable to return to the prior topic if they failed to understand the current topic.

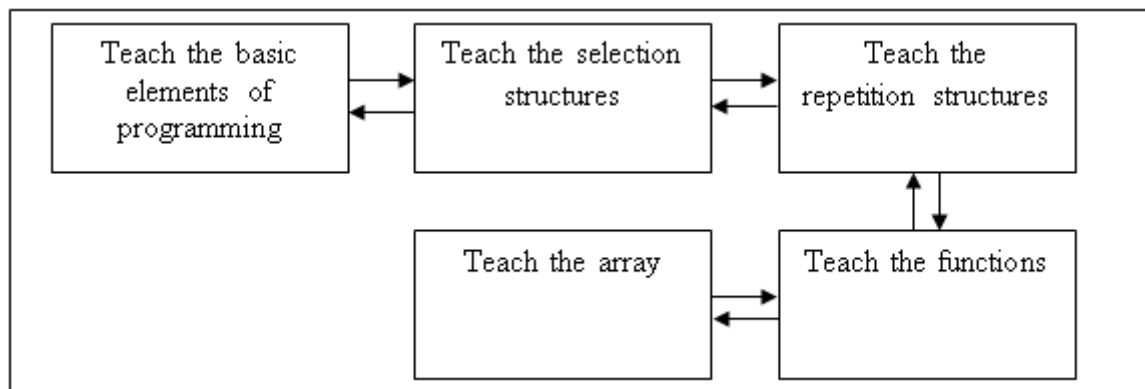


Figure 1 : Framework of Teaching Programming.

Based on all the motivation above, a simple teaching aid focused on array topic in C++ introductory course is developed using Java Script language. A questionnaire was designed and used as an evaluation procedure to assess on the benefits and effectiveness of the teaching aid. The following section discusses the details of the related work, evaluation procedure, results and findings of the study.

## **2. RELATED WORK**

As mentioned by Abid (2011) in his study, a C++ programming course usually includes basic programming syntax and semantics, programming concepts such as declarations, math's operations, control structures, arrays, functions, programming logic and application of problem-solving strategies using computer language.

In improving the understanding of computer program and programming languages, learners must be able to view a program code in different ways. Inability of visualizing a valid mental model of computer program contributes to student's low capability in programming comprehension and skills, lack of interest in the subject and poor grades. In addition, the traditional auditory lecturing and text-based teaching materials also contributed to the learner's lack of ability to comprehend computer programming.

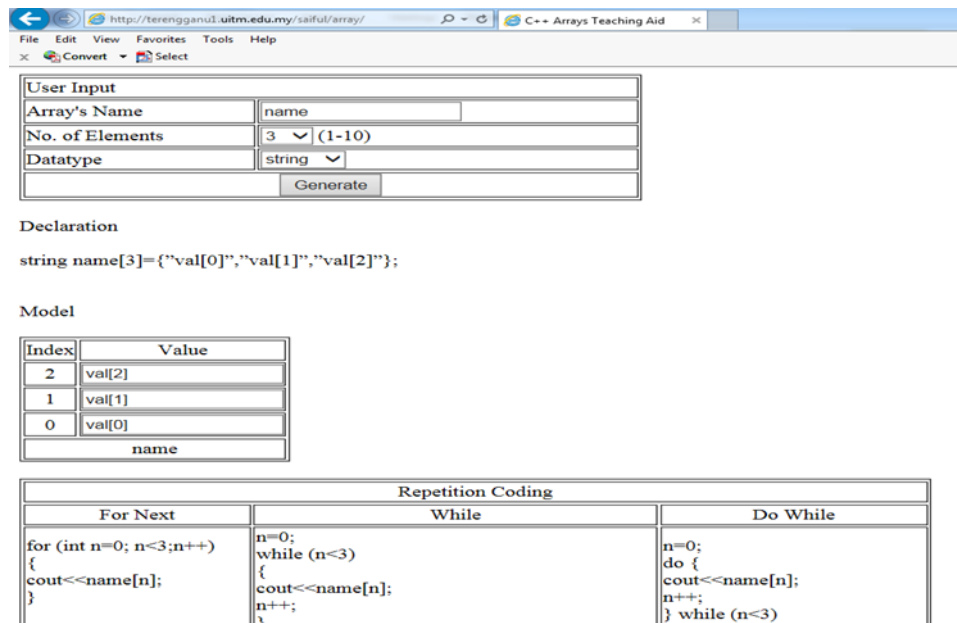
Many previous researches including Bucks and Oakes (2008) ruled out that engineering students fall on the "visual learners" category of Felder-Silverman Learning Styles Model. This is the reason why Myers (1988) recommended programming subject instructors to explain some difficult programming concepts visually through pictures, diagrams and etc. Furthermore, Chang (2011) has also suggested that the application of animation, multimedia and visualization for the development of learning tools be used in programming courses to enhance programming comprehension.

The latest research on learning tools development for introductory programming courses was done by Tuparov et. al (2014). He had developed the interactive simulation-based learning objects (LOs) in an introductory course of programming, focusing on sorting and searching algorithms. Other than that, Abid et. al (2011) proposed a computer aided language software as a teaching and learning tool for students to understand the fundamental concepts in programming. In addition, Major et. al (2011) used robots as tools in teaching of introductory programming to novice learners, whereas Bucks and Oakes (2008) explored graphical programming languages in teaching introduction to programming concepts to learners. Other research studies in the same field were also carried out by Cooper et. al (2000) and Hulls et. al (2005). Cooper et. al (2000) used a 3-D interactive animation environment called ALICE as a tool to actively engage students to comprehend fundamental of programming concepts while Hulls et. al (2005) created an interactive online tutorial assistance for a first year programming course.

Having taken the suggestions of previous studies into account, the researchers of the study have created an interactive assistance using web tool to address one of the difficult concepts in programming which was Array data structures.

### 3. EVALUATION PROCEDURE AND WEB TOOL SAMPLE

Fifty nine students in the second and third year Bachelor of Civil Engineering (Infrastructure) programme, Universiti Teknologi MARA, Pulau Pinang participated in the study towards the end of semester in 2014 after the C++ Array topic was taught. The respondents were given the opportunity to use the teaching aid as shown in Figure 2 before questionnaires were distributed.



The screenshot shows a web browser window titled "C++ Arrays Teaching Aid". The browser address bar shows the URL "http://terengganu1.uitm.edu.my/saiful/array/". The page content includes a "User Input" form with fields for "Array's Name" (containing "name"), "No. of Elements" (set to 3), and "Datatype" (set to string). Below the form is a "Generate" button. Underneath, the "Declaration" section shows the code: `string name[3]={"val[0]","val[1]","val[2]"};`. The "Model" section contains a table:

Index	Value
2	val[2]
1	val[1]
0	val[0]
name	

Below the model table is a "Repetition Coding" table with three columns: "For Next", "While", and "Do While".

For Next	While	Do While
<pre>for (int n=0; n&lt;3;n++) { cout&lt;&lt;name[n]; }</pre>	<pre>n=0; while (n&lt;3) { cout&lt;&lt;name[n]; n++; }</pre>	<pre>n=0; do { cout&lt;&lt;name[n]; n++; } while (n&lt;3)</pre>

Figure 2: C++ Array Teaching Aid Web Tools.

The questionnaire which comprised of 21 questions was uploaded on the web before conducting the survey. The questions in the study consisted of three parts: student's profile, student's perception on the topic of array after the tools was tested and student's perception on the tools features. All of the respondents completed the questionnaires. Therefore, the number of valid data maintained as 59. The subjects were categorized as novice learners because they had learned programming languages for the second time but it was their first experience learning array topic in C++. The respondents learned C++ programming in their fourth semester. While conducting the survey, the instructor clearly explained the instruction for each of the questions and allocate sufficient time for the respondents to answer the questions. It took about ten minutes for the students to complete the survey.

### 4. RESULT AND DISCUSSION

Generally, by referring to Figure 3, about 83% of the respondents were categorized as beginners with moderate skills in C++ programming. These students were from Matriculation and Diploma program.

The survey also indicates that almost all the students have basic knowledge and skills in C++ programming. However, Array topic was not covered in the syllabus during their matriculation and diploma program. Array is an introduction of basic data structure for most of the Bachelor program at tertiary level.

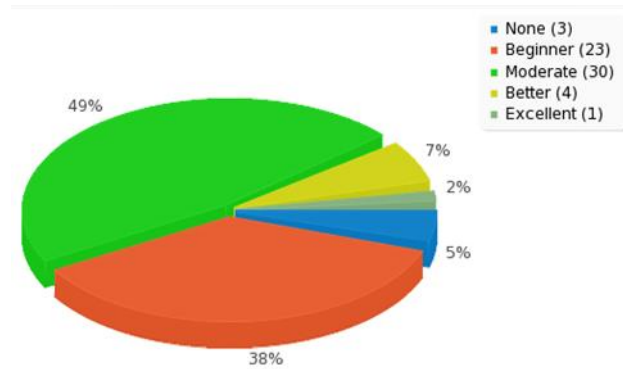


Figure 3: The Pie Chart of Students' Levels, Skills and Knowledge in C++ Programming.

Interestingly, positive feedback was given by all respondents after the non-functionality testing was carried out on the interactive web tools.

Table 1 shows approximately more than 70% of the respondents acknowledged that the web tools had improved their knowledge and confidence level on the Array topic. Yet, almost 38% of the respondents were unable to relate the usefulness of this web tool in creating the awareness on the importance of learning Array topic as well as its application. This result significantly influenced by the scope of the problem-solving project developed by some of the students.

Group project is one of the assessments for most C++ programming courses offered. The project scope and requirements should include the application of basic control structure such as selection and repetition, variables declaration as well as the used of Array data structures. Unfortunately, there were groups of students who were incapable to accomplish some requirements of group project especially the application of Array data structures in their C++ program. These were the students who contributed to the 38% of the respondents that were unable to appreciate the importance of Array data structure application in problem-solving.

Table 1 : Non-functionality Testing on Interactive Web Tools.

NO.	QUESTION	*LEVEL OF SATISFACTION				
		1	2	3	4	5
1	The tools make the student easy to understand the basic declaration of Array.	-	-	26.23	57.38	16.39
2	The tools increase confidence level towards the topic of Array.	-	-	26.23	68.85	4.92
3	The tools improve my knowledge on the topic of Array.	-	-	21.31	70.49	8.20
4	The tools create the awareness on the importance of Array in my life.	-	1.64	36.07	55.74	6.56
5	The tools help the student on how to apply Array in the real problem.	-	1.64	27.87	63.93	6.56

\*1 – Extremely Disagree, 2 – Disagree, 3 – Neutral, 4 – Agree, 5 – Extremely Agree

Figure 4 show that approximately 73% of the respondents understand the basic declaration of Array in C++ programming.

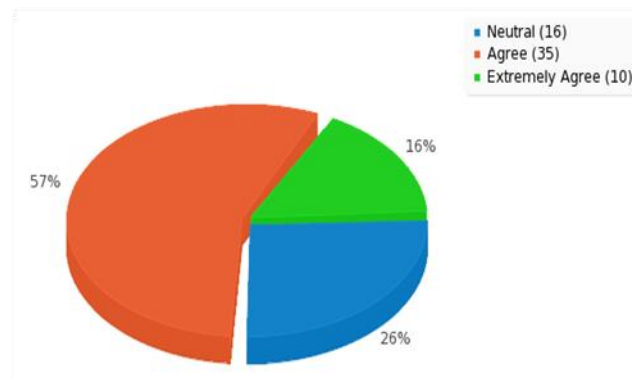


Figure 4: The Pie Chart of Students' Understanding on Basic Declaration of Array in C++.

The web tools assisted them to comprehend better on the concept of array declaration using different data types, through online interactive visual examples. The students were guided theoretically about 45 minutes by an instructor before the interactive web tools were used and tested by them. They were given 30 minutes to practise and familiarise themselves with the web tools. Towards the end of the class session, simple quiz on Array topic was given to the students and the student's marks obtained from the quiz were extremely high. This result indicates that the web tools aid the students in basic Array declaration and knowledge comprehension.

Functionality and usability testing on the web tools were also conducted by the same group of students. Figure 2 shows the interactive web tools that have been tested and evaluated. Table 2 shows the summary analysis table on the usability and functionality test that has been conducted.

Table 2: Analysis of Usability and Functionality Test on Web Tools.

USABILITY & FUNCTIONALITY TEST	AVERAGE	MODE	MEDIA N	STANDARD DEVIATION
1. The web tools are easy to use	<b>3.94</b>	<b>4</b>	<b>4</b>	<b>0.6327</b>
2. The interface is friendly	<b>3.94</b>	<b>4</b>	<b>4</b>	<b>0.5981</b>
3. The language is easy to understand	<b>4.02</b>	<b>4</b>	<b>4</b>	<b>0.4371</b>
4. The web tools are informative	<b>3.88</b>	<b>4</b>	<b>4</b>	<b>0.6058</b>
5. The web tools encourage the user practices more exercises	<b>3.85</b>	<b>4</b>	<b>4</b>	<b>0.6520</b>
6. I would like to introduce the web tools to others	<b>3.90</b>	<b>4</b>	<b>4</b>	<b>0.5921</b>

The analysis from Table 2 is illustrated in the Figure 5.



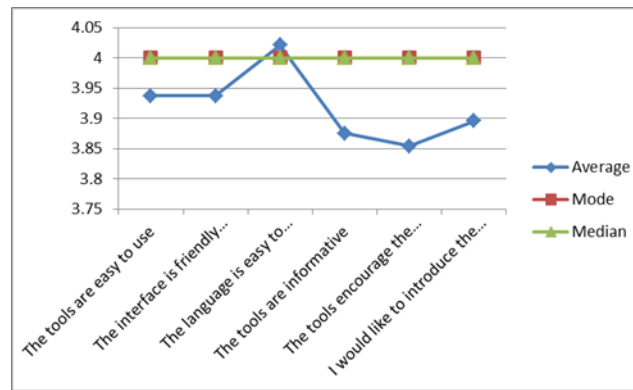


Figure 5 : The Line Chart Shows the Result Patterns for the Usability and Functionality Test.

The Likert scale from 1 to 5 was used to evaluate the usability and functionality of the web tools. Only six (6) questions that were relatively connected to non-functionality testing were selected to be discussed further.

Positive responses were achieved from the functionality testing that was conducted. More than 80% of the respondents were satisfied with the web tools functionality. The respondents claimed that the web tools were easy to use and decided to recommend this tool to other students in UiTM branches and faculties. 75% of the respondents agreed that the web tools were informative and facilitated the students in mastering Array topic since it encouraged the students to practise more examples.

## 5. CONCLUSION

The engineering learners' visual learning styles that are not align with programming language text-based structure and in addition to the textual and auditory teaching methods contribute to misunderstanding, lack of program comprehension, inattentiveness and poor performance of students in programming subject. Many earlier and current research studies explored on computer visualization approaches and tools to create effective and enhance the understanding of programming. In order to address this issue, a teaching aid was developed, focusing on array topic in C++, as a tool to assist the teaching and learning process of programming concepts. Functional, non-functionality as well as usability testing on this web tool were also conducted through questionnaires distributed to the engineering students. The initial results show a positive response from the respondents of this study and optimistic impact on students' programming comprehension and learning skills. In addition, this teaching aid should also serve as a resourceful assistance to educators in the process of teaching basic concepts in introductory programming course.

## REFERENCES

- Abdel Rahman S.M., Al-Syabi B., & Kaabi S.S.A. (2016). The Reasons behind the Weakness of some Students in Programming Courses in the College of Applied Science, Ibr. *Modern Education and Computer Science Press*, 1, 48-54.
- Abid, S.H., Zehra, S., & Iftikhar, H. (2011). Using Computer Aided Language Software for Teaching and Self-learning. *14th International Conference on Interactive Collaborative Learning (ICL2011)*, 102-106.

- Bucks, G., & Oakes, W. (2008). Work in Progress – Impact of Graphical Programming Environments on Learning and Understanding Programming Concepts. *38th ASEE/IEEE Frontiers in Education Conference*, 23-24.
- Chang, W., & Zhan, R. (2011). Cognitive Knowledge status of Learning Path in C++ Programming Language based on Rule Space Model for College Students. *7th International Conference on Networked Computing and Advanced Information Management (NCIM)*, 307-312.
- Cooper, S., Dann, W., & Pausch, R. (2000). Alice: a 3-D Tool for Introductory Programming Concepts. *Journal of Computing Sciences in Colleges*, 15(5), 108-117.
- Hulls, C.C.W., Neale, A.J., Komalo, B.N., Petrov, V., & Brush, D.J. (2005). Interactive Online Tutorial Assistance for a First Programming Course. *Journal of IEEE Transactions on Education*, 48(4), 719-728.
- Major, L., Kyriacou, T., & Brereton, O.P. (2011). Systematic literature review: teaching novices programming using robots. *15th Annual Conference on Evaluation and Assessment in Software Engineering (EASE 2011)*, 21-30.
- Myers, B.A., Chandhok, R., & Sareen, A. (1988). Automatic Data Visualization for Novice Pascal Programmers. *IEEE Workshop on Visual Languages*, 1988. 192-198.
- Rist, R. (1996). Teaching Eiffel as a first language. *Journal of Object-Oriented Programming*, 9, 30-41.
- Tuparov, G., Tuparova, D., & Jordanov, V. (2014). Teaching sorting and searching algorithms through simulation-based learning objects in an introductory programming course. *5<sup>th</sup> World Conference on Educational Sciences (WCES 2013)*, 2962-2966.