LEARNING DIFFICULTIES OF OBJECT ORIENTED PROGRAMMING (OOP) IN UNIVERSITY OF ILORIN - NIGERIA: STUDENTS' PERSPECTIVES

[1]LAWAL BASHIRU, [2]AGUNLEJIKA ADEROGBA JOSEPH

[1,2]Department of Computer Science, Federal College of Education (Technical) Gusau-Nigeria
E-mail: bash3119@yahoo.com, agunadejoseph@yahoo.com

Abstract- Many studies have described natures of difficulties facing students in learning programming. However, very little work has been done to identify the reasons that cause the difficulties students encounter in learning the subject. This study is a survey designed to determine natures and causes of difficulties facing students in learning of Object Oriented Programming (OOP). A mixed methodology was applied where both quantitative and qualitative methods were used to investigate the study under view. Second and third year computer science students of the University of Ilorin, Nigeria were used as population for the survey. The study was conducted by enquiring from students in the subject, what they struggle to learn, using questionnaires as the research instrument. Results obtained from the study show that students perceived as most difficult issues in learning OOP: understanding what exactly happens inside the computer when it executes a program, misunderstanding of memory operations, difficulty in building up and understanding how the program solve the given problem task. And these were caused from such reasons identified as: ineffective teaching pedagogy in OOP, poor programming planning and complexities in the available teaching and learning tools in OOP. Based on the findings, it was recommended that: educators and programming tools developer should be made to be aware of the data bank containing various difficulties facing programming students. This will give information on direction on how to adequately prepare and develop needed teaching materials in OOP and programming in general.

Keywords- Learning Difficulties, OOP, Students Perspectives, Misunderstanding, Teaching Materials

I. INTRODUCTION

The teaching of Object Oriented Programming (OOP) has become an exceedingly expanding preference at various international higher institutions. In the curricula of all universities, educating students in the field of computing, software design, information science and engineering, OOP is always presented (Donchev and Todorova 2008). Most universities programming courses have moved from Pascal or C to object-oriented languages such as C++, Eiffel, Smalltalk, Delphi and lately Java, and one of Visual Studio.Net languages such as VB.Net, C# or J#, as evidenced by popularity of most computer programming texts (Johnson & Moses 2008:95). The main reason for this transition according to Okur (2006:149) are related to "OOP having some conceptual edge like; Abstraction, Inheritance, Polymorphism and Encapsulation". This transition has in some ways, put extra tensions on both programming tutors and students of the subject.

Also, many studies have described natures of difficulties facing students in learning programming and OOP in particular (e.g. Gal-Ezer, et al. 2009; Sivasakthi and Rajendran 2011). However, very little work has been done to identify the reasons that cause the difficulties students encounter in learning the subject. The lack of such studies may result in programming tutors simply relying on experience and intuitions (Guo 2006). There is also a need to offer appropriate guidelines regarding appropriate activities to support the teaching and learning of OOP (Or-Bach & Lavy, 2004:82).

Results obtained from this study show that students perceived as most difficult issues in learning OOP: understanding what exactly happens inside the computer when it executes a program, misunderstanding of memory operations, difficulty in building up and understanding how the program solve the given problem task. And these were caused from such reasons identified as: ineffective teaching pedagogy in OOP, poor programming planning and complexities in the available teaching and learning tools in OOP.

II. STATEMENT OF THE PROBLEM

Students' performance in programming courses in Nigerian tertiary institutions is a matter of concern to all stakeholders in education and software industries in Nigeria.

There is also a very serious concern about why programming courses tend to have higher failure rates most especially in OOP. Unfortunately, there is no existing data or research suggesting reasons for these situations (Lawal 2012).

The researcher's experiences as a student and a tutor in Nigerian tertiary institution are the reasons for the making Nigerian Universities as the case study of this work. The aim was to investigate thoroughly and find reasons for the perennial difficulties facing students in learning OOP. This is with the aim of suggesting ways of improving the teaching and learning methods in OOP and helping students build programming skills.

III. PURPOSE OF THE STUDY

This study was designed to investigate and analyse the various natures of difficulties students encounter in learning OOP (from students' point of view). It was also designed to identify those reasons that cause the difficulties as well as possibilities of dealing with them.

IV. RESEARCH QUESTIONS

The following questions were raised to direct the study:

1. How is OOP being delivered at undergraduate level in Nigerian Universities?
2. What are the difficulties that students encounter in the learning process of OOP?
3. What are the causes of difficulties students encounter in their learning of OOP?
4. How can the teaching and learning methods in OOP be improved to provide an effective teaching and learning processes in OOP?

V. METHODOLOGY

In this study, a mixed methodology was applied where both quantitative and qualitative methods were used to investigate the study under view. In this regard, the researcher collected facts from respondents (students) through a survey method. The researcher wanted to study the present situation in the teaching and learning process in Object Oriented Programming, find out perceptions on the difficulties encountered by learners during the process and gather contributions from relevant sources on what can be done to balance situations in the process.

To achieve this goal, the researcher used the University of Ilorin - Nigerian as case study. Second and third year computer science students of the University were used as population for the survey. Purposive sampling technique was used. This sampling method restricted selection of respondents to only those students that offer OOP at undergraduate level. The questionnaire has an introductory letter attached with it. The researcher with the help and cooperation of two supporting staffs of the department of computer science of the university distributed the questionnaire to the concerned students. The student questionnaires were available for seven days. Students were given seven days to complete and return the questionnaire. The student questionnaire was built to collect and analyse students' perceptions on their learning of OOP.

To gain accurate understanding of their learning situation, the questionnaire contains four sections with five printed pages. Each section containing different questions asking about student views on the difficulties they encountered in the learning of OOP and what they considered could be their causes. A total of 155 questionnaires were completed and returned by the students. Statistical quantities such as Percentage, Mean and Grand Mean were constructed and used to analyse the facts gathered during the survey.

VI. RESULTS AND DISCUSSIONS

The findings of this survey are shown below:

Research Question 1:
How is OOP being delivered at undergraduate level in Nigerian Universities?

Table 1 presents students' self-estimation about the delivery and their learning in OOP. Results and analysis of their self estimation in learning OOP of Question 1 from student questionnaire: "Do you consider OOP difficult to learn?" shows that gross majority of the inquiring students consider OOP difficult to learn.

146(94.2%) of the enquiring students consider OOP very hard and difficult to learn. Majority of the students commented in the space provided that ineffective teaching methodology, lack of good help tools and high level of abstraction of OOP are their reasons.

Others complained about few and complex developmental environments that are available contribute to their difficulty.

Results from Question 2: "when do you feel that you learn issues about OOP?" show that students learn more issues about OOP while working in group than during lectures or working alone. 102(66%) feel that working in group on programming assignment and 93(60%) on exercise sessions in small group have helped to improve their learning in OOP.

Table 1. Students' self–estimation about their learning in OOP

Qn	Question Descriptions	Number of students who choose option(s)	%
1.	Do you consider OOP difficult to learn? (Please tick) • Yes • No • Neither	146 9 0	94.2 5.8 0

2.	When do you feel that you learn issues about OOP? Please tick option(s)	20	13
		55	35.5
	• During Lectures	93	60
	• During Practical Session	0	0
	• In Exercise Sessions in Small Group	23	15
	• In Surgery Classes	102	66
	• While Wor Alone Programming Coursework		
	• While Working Group Programming Coursework		
3.	Which OOP Language and Environment do you prefer for your learning of OOP?	22	
		138	14.2
	• OOP Language	12	89
	Java	0	8.4
	C++	0	0
	C#	7	0
	• OOP Environment	22	4.5
	BlueJ	140	14.1
	JGrasp	36	90.3
	NetBeans		23.2
	BorlandJ		
	BorlandC		
	Visual Studio		
4.	What nature of program examples is to be used for effective learning of course? Please tick appropriate:	145	93.6
		23	15
		6	12.8
		3	1.94
	• Real world exam	43	27.7
	• Pure mathema examples		
	• Half-ready project, enriched functionality		
	• Complex examples		
5.	What kind of materials helped you in learning OC Please tick as appropriate:		56.8
	• OOP course book	88	29
	• Lecture notes handout	45	27.7
	• Online materials	43	7.7
	• Interactive visualization	12	9.6
	• OOP language environment	15	14.1
	• Still pictures programming structure	22	29
	• Worksheets, Exercises ques and answers	45	

Results obtained from inquiring students on question which OOP language and environment they prefer for their learning of OOP show that C++ is the most preferred OOP language and Borland C environment is the most favourite OOP environment. 138(89%) of the inquiring students prefer C++ and 140(90.3%) prefer BorlandC as their favourite OOP development environment. This result shows that C++ is the dominant OOP language used in the University. The popularity is influenced by the OOP language in which they studied in their individual university. Analysis on students comment about their choice of language and environment shows that their transition from C language in their first year to C++ has highly influenced their decision. Java is the next favourite OOP language as shown from Table 1 (Q3). The reason could be that the language is taught at later stage of their undergraduate program.

Students' answer to the question what nature of programming example is to be used for effective learning in OOP show that majority prefer real world examples. 145(93.6%) of the inquiring students opined that real world examples are ideal for effective learning in OOP. Other students 43(27.7%) suggested simple task example gradually growing to complex is essential for effective learning in OOP.

Students' response to the question what kind of materials has helped you in learning OOP indicate that some materials have helped their understanding in OOP: 88(56.8%) of the inquiring students consider OOP course book, 45(29%) choose both lecture notes/handout and worksheets, exercises questions and answers and 43(27.7%) select online materials to have better their learning in OOP. The implications of these responses could be interpreted as several factors such as ineffective pedagogy, insufficient teaching and learning resources and aids have influenced their choices.

Research Question 2:
What are the difficulties students encounter in the learning process of OOP?

Results presented in Table 2 shows that students perceived as most difficult issues in learning OOP:

understanding what exactly happens inside the computer when it executes a program (UCP), misunderstanding of memory operations (MMO), difficulty in building up and underst-anding how the program solve the given problem task (UPT). Other issues that were perceived sometimes difficult by the inquired students in their learning of OOP were issues such as: detecting too many errors in my code that I don't know how to solve, so I stuck (ECD), Difficulties in testing and debugging of program, reading someone else code and detecting of logic errors (DBG) and difficulties in using heavy professional program development environment (HDE).

However, there are some other issues, which inquired students had underestimated and described as not at all difficult in their learning of OOP. That is the issue concerning Class-Object and the relationship between them.

Table 2. Students' self-estimation about the level of potential difficulties they encounter in learning OOP

Question2: Describe the nature of difficulties you encountered during your learning o OOP from the list below. Give a number on a scale:

(1-Very difficult, 2-Difficult, 3- Sometimes difficult , 4- Rarely difficult, 5- Not at all difficult)

Nature of Difficulties	Code	Mean	Rank
• Cannot understand what exactly happens inside the computer when it executes a program	*UCP*	*1.14*	*1*
• Difficulties in using heavy professional program development environment	*HDE*	*1.15*	*2*
• Misunderstanding of memory operations	*MMO*	*1.25*	*3*
• Lack of good help tools and reference materials	*LTM*	*1.24*	*4*
• Difficulties in testing and debugging of program, reading someone else code and detecting of logic errors	*DBG*	*2.17*	*5*
• Don't know where to start or how to go about completing a given task	*STR*	*2.25*	*6*
• Detecting too many errors in my code that I don't know how to solve, so I stuck	*ECD*	*2.25*	*7*
• Difficulties in building up and understanding how the program solve the given problem task	*UPT*	*2.26*	*8*
• Difficulties in designing a program and solving a complex task	*SCT*	*3.03*	*9*
• Difficulties in applying the correct logic	*LGC*	*3.35*	*10*
• Difficulties in understanding the basic OO and programming concepts	*BSC*	*4.18*	*11*
• Cannot differentiate between a class - object and the relationship between them	*COB*	*4.28*	*12*
• Lack of time and effort in doing practical programming exercise regularly	*TME*	*4.39*	*13*
• Other, please specify and grade_____			

Research Question 3:
What are the causes of difficulties students encounter in their learning of OOP?

Table 3 shows the summary of result of students' opinion on what they consider as the main causes of difficulties they encounter in learning OOP. The frequencies of the potential difficulty as ticked by the enquiring students are presented in the table in a descending order. Students' answer to the question what would you identify as the main causes of difficulties in learning OOP show that students suffer massively in learning OOP 147(94.88% each on) due to an ineffective teaching methodology and inability to mentally capture what is happening behind the scene as their program executes.

Other reasons for their trouble in learning the course according to the students are: 123(79.4%) consider reason caused due to their poor programming ability, 112(72.3%) choose lack of effective teaching and learning aids, 92(59.4%) consider rush to coding without much thinking, nor having any program structure in mind as contributing factor to their problems in OOP. Presentations of instructors and their attention on student are seen by 86(55.5%) of the inquiring students as reasons that have made OOP difficult for them to learn.

Table 3. Summary of result of students' opinion on causes of difficulties they encounter in learning OOP

Question: What would you identify as main causes of difficulties in learning OOP? Please tick as appropriate from the list provided below

Nature of Difficulties	Number of student that choose option(s)	%
• Teaching methodology is less effective	147	94.8
• While typing or reading executable code, don't mentally connect to what happens inside the computer with regards to changes in both computer memory and in objects' state should the program executes	147	94.8
• Poor program planning ability	123	79.4
• Lack of effective teaching and learning aids	112	72.3
• Some of the available OOP environment are too complex to use	110	71
• Rush to coding without much thinking, nor having any program structure in mind	92	59.4
• Presentations of instructors and their attention on student	86	55.5
• Lack of interest to learn OOP	21	13.6
• Syllabi coverage is too wide	20	13
• Less examples in practical use	20	13
• Other, please specify and grade	2	1.3

Research Question 4:

How can the teaching and learning methods in OOP be improved to provide an effective teaching and learning processes in OOP?

A lot of contributions were gathered from the enquiring students on what they consider could reduce the potential difficulties they encounter in learning OOP and ultimately better their

understanding of the course. It is very difficult to analyse their submission but the researchers have tried their best to summarise their major submission as follows:

1. An effective pedagogy in teaching and learning of OOP
2. Teaching should focus on program decomposition and planning attitude
3. Provision of many practical and real life exercises as possible
4. Lectures need to be more visual and interactive
5. Use of modern IDEs and the use of source control tools such as GIT

Recommendations

The following recommendations for future work are made:

1. A similar investigation can be conducted to analyse programming tutors' perception on the same issue to be able to draw a valid conclusion on the study under view.
2. A data bank containing various difficulty issues facing novice programmers can be created. An educational environment can as well be simulated in which certain strategies can be evolved to tackle the problems that were found. This will go a long distance to assist novices and better their learning of OOP and other programming paradigms.
3. Educators and programming tools developer should be made to be aware of the data bank created. This will give information on direction on how to adequately prepare and develop needed teaching materials in OOP and programming in general.

CONCLUSION

Responses gathered from the survey show that students encounter difficulties in their learning of OOP and other issues that are related to their learning instincts. For example, the concept of difficulties evolved from issues such as: abstract nature without proper grounding in programming, IDEs are troublesome, misunderstanding of memory operations and misconceptions between a class and an object. These findings can also have direct bearing on how the course curriculum in OOP could be reviewed. An in-depth understanding of potential difficulties students encountered during the learning process in OOP from the learners' perspectives should lead to designing effective teaching tools in OOP and subsequently resulting into an effective delivery of OOP. However, since this is an initial investigation, further investigation can be conducted about other difficulties that are facing students in learning OOP from the tutors, point of view and action research can serve as follow up to implement a reviewed teaching methods.

Identifying reasons for students difficulties will provide educators and curriculum planners a source of data which can give information for direction on how syllabi and curriculum for OOP could be developed. This will also assist programming teachers as well as developers of programming teaching tools to be aware of varying degree of problems and potential difficulties facing students that were found, so that they can adequately prepare, plan and design an effective teaching and learning process and programming teaching tools in OOP.

REFERENCES

[1]. Donchev, T. & Todorova, E. (2008) 'Object oriented programming in Bulgarian Universities informatics and Computer Science curricula Informatics in Education, 7(2), pp.159-172

[2]. Gal-Ezer, J., Vilner, T. & Zur, E. (2009) 'Has the aradigmshift in CS1 harmful effect data structures: a case study. SIGCSE Bulletin, 41(1), pp. 67-78

[3]. Guo, H. (2006). Visual Materials and Learning aids in Teaching Object Oriented Programming. Proceedings of the 7th Annual Conference of the Higher Education Academy Information and Computer Sciences (29th – 31st August, 2006), Trinity College, Dublin, pp.64-69

[4]. Johnson, R.A. & Moses, D.R. (2008). Object-first vs. Structures-first approachesto OO Programming education: an empirical study. Academy of Information and Management Sciences Journal,11(2), pp.95-103

[5]. Lawal B. (2011) Study of Teaching and Learning Methods in Object Oriented Programming.Unpublished MSc thesis submitted to Coventry University United Kingdom Okur, M. C. (2006). Teaching object oriented programming at the introductory level Journal of Yasar university, 1(1), pp.149-157 [online] available from http://joy.yasar.edu.tr/makale/no2_nol1/05_m_cudi_okur.pdf [12 September 2012]

[6]. Or-Bach, R. & Lavy, I. (2004). Cognitive activities of abstraction in object orientation: an empirical study' SIGCSE Bulletin, 36(2), pp.82-86

[7]. Sivasakthi, M. & Rajendran, R. (2011) 'Learning difficulties of 'Object- Oriented

[8]. Programming using Java: students' perspective'. Indian Journal of Science an Technology, 4(8), pp.983-985[online] available from http://indjst.org/archive/vol.4.issue.8/24-aug11sivasakthi.pdf [22 Sep. 2012]

★★★