

## Analysis of Learning Difficulties in Object Oriented Programming in Systems Engineering Students at UNTELS

Teodoro Diaz-Leyva\*, Omar Chamorro-Atalaya

Faculty of Engineering and Management, Universidad Nacional Tecnológica de Lima Sur Lima, Lima, 15047, Perú

### ARTICLE INFO

#### Article history:

Received: 01 September, 2020

Accepted: 01 December, 2020

Online: 28 December, 2020

#### Keywords:

Object Oriented Programming

Learning difficulty

Programming Language I

Systems Engineering Professional Career

### ABSTRACT

*The learning difficulties of Object-Oriented Programming can be one of the causes of failure in college students, which can even lead to student dropout. This article has the purpose of analyzing the main learning difficulties of object-oriented programming using the Java programming language in Systems Engineering students of the Universidad Nacional Tecnológica de Lima Sur (UNTELS), research that arises because the students reflect low academic performance in the subject of Programming Language I (object-oriented programming) that is developed in the fourth academic cycle with six hours of practice and zero hours of theory, because this is determined by the study curriculum. To collect data, first, the possible causes associated with learning difficulties have been identified through direct dialogue with the students of the second academic semester of 2019, then with these causes a questionnaire has been prepared to apply the survey technique. establishing indicators classified in the dimensions Teaching methodology, Previous knowledge of programming courses, reading materials, learning concepts and Process of learning to program. This research yielded as a result that students perceive with greater difficulty the indicators corresponding to the dimension "Learning concepts", in second place, the dimension Process of learning to program is presented, in third place the dimension Teaching methodology, in fourth place the Reading materials dimension and fifthly the dimension Previous knowledge of programming courses, the results of which will serve as a basis for the teachers of the subject to improve the strategies in the teaching / learning process in the students of the professional career of Engineering Systems.*

## 1. Introduction

In [1], the author expresses due to the generalized use of the object-oriented programming paradigm, teaching is not without difficulties, since it is based on a solid base of elementary concepts on which to add experiences of design of applications oriented to objects. Currently, one of the most widely used paradigms in software development is Object Oriented Programming (OOP), it implements programs in terms of objects, includes concepts of inheritance, cohesion, abstraction, polymorphism, coupling and encapsulation.

In [2], the author indicates, the academic programs of Engineering in Systems, the learning of the programming oriented to objects is seen in relation to the scene of the life of their professional tasks. Students are very concerned that their learning is an element that allows them to act effectively and efficiently

outside the university environment. Likewise, in [3] the author points out that it is necessary for the student to learn OOP and not the peculiarities of the chosen language, to present the theoretical contents in a way that facilitates their understanding and progressive learning.

In [4], the author points out, many times, even using languages that support OOP, the mistake is made of not programming in this methodology, because there are no clear concepts in this regard. Also in [5], the author indicates that, for many years, the emphasis in teaching was to transmit concepts and procedures whose meaning was often not understood. Also in [6], the author expresses the object-oriented paradigm includes a large number of concepts that allow the development of robust applications.

In [7], the author points out that currently there is a variety of programming languages, learning generally requires knowledge of flow diagrams, pseudo-code and the syntax of the programming language itself to build programs under the structured approach,

\*Corresponding Author: Teodoro Neri Diaz Leyva, Sector 3 Grupo 1A 03, Av Central, Villa El Salvador-UNTELS, 982716223 & is\_diazl@hotmail.com

[www.astesj.com](http://www.astesj.com)

<https://dx.doi.org/10.25046/aj0506203>

that is, using sequential, conditional and repetitive structures. Thus, learning to program at a professional level, regardless of the language used, is not a one-day task. There are students who fail to acquire the necessary programming skills, even after completing a Computer Science Programming Fundamentals course.

For many students, learning difficulty occurs as they move from structured programming to object-oriented programming and integrate various software development technologies. Thus, in [8], the author points out that the problem of learning object-oriented programming is manifested since it is a complex issue that involves the integration of many elements such as the object-oriented paradigm, the programming language, the environment development, development methodology, modeling language, development patterns. and programming logic. Therefore, students are faced with an overwhelming number of concepts in a short period of time, which makes it difficult to assimilate and develop skills to generate lines of code.

When studying complex courses such as computer programming, one should begin by understanding the fundamental concepts of programming, in this sense, in [9], the author indicates the search for strategies that allow improving the teaching and learning of programming. an indispensable task. Among the causes of failure, it has been concluded that they do not reside in the student's difficulty in translating the solution of a problem into the phrases of a programming language, but rather have to do with a lack of methodology, habit and skill. to solve problems.

In [10], the author explains that the learning of the disciplines of Algorithmic and Algorithmic Programming presents, perhaps, one of the highest levels of difficulty in the careers of Computer Science and Computer Science. Historically, students have faced problems assimilating abstract mathematical notions, particularly when these include the dynamics of how algorithms manipulate data (Acm and Ieee-Cs, 2009). Object-oriented programming courses require prior knowledge of programming logic that is generally learned in algorithm courses, however, this is where students already have learning difficulties.

In [11], the author points out that the problem of university dropouts is complex and includes various causes; Institutional characteristics would be the first dropout factor, followed by vocational, academic and, finally, economic interests. Likewise in [12], the authors point out that dropout is associated with academic capital, the perception of performance, skills gaps to face studies, time management and the incorporation of learning techniques and habits that affect the decision to drop out of students.

At the Universidad Nacional Tecnológica de Lima Sur (UNTELS) the Programming Language I subject refers to Object-Oriented Programming with Java programming language, it is dictated in the fourth cycle of the Professional Career of Systems Engineering, which is develops in six (6) hours of practice and zero (0) hours of theory; After having taken two structured programming subjects, the academic performance of the students is not optimal, in the first academic semester of 2019 of 28 students enrolled 64% failed the course and in the second academic semester of 2019 of 25 students enrolled 40%.

In this context, the present research aims to describe the factors that hinder the learning of Object-Oriented Programming

with Java Language, since the student goes from studying programming with a structured approach to object-oriented and with little theory about the different topics addressed during its development. For this, a direct dialogue was established on the possible problems associated with learning difficulties, which have been ordered and reflected in a physical questionnaire regarding the teaching methodology, previous knowledge of programming courses, reading materials, learning concepts. and the process of learning to program with the purpose of improving the teaching and learning strategies in the students of the Programming I language course in the professional career of Systems Engineering at the National Technological University of Lima Sur.

## **2. Methodology**

### *2.1. Research level*

The research level of this article is descriptive / correlational. Descriptive because it specifies the characteristics of a group of students with respect to the learning difficulties of Object-Oriented Programming and is correlational since it seeks to know the degree of association between the dimensions Teaching methodology, Reading materials, Previous knowledge of courses Programming, Learning Concepts and Learning Process for Programming. In [13], the author describes the purpose of this type of study is to know the relationship or degree of association that exists between two or more concepts, categories or variables in a particular context.

To achieve these results, the following objectives have been set: to describe the main difficulties in learning object-oriented programming. Determine the dimension with the greatest learning difficulty of Object Oriented Programming.

### *2.2. Population and sample*

The population is made up of 25 students of the Programming Language I subject of the second academic semester of 2019 of the Systems Engineering professional career of the National Technological University of Lima Sur.

The sample is equal to the population, because it was only considered to apply the questionnaire to students in the second academic semester of 2019, which corresponds to a number of 25 and who take the subject of Programming Language I (Object-Oriented Programming).

In [14], the authors in their article "The Proportionality of Women Graduated from the Professional Career of Mechanical and Electrical Engineering at UNTELS: Analysis of their Academic Performance and Field of Labor Action", point out that the sample will be equal to the population; Because the sample is less than 50, that is, the weighted average and the weighted average by specialty (Mechanical Subjects and Electrical Subjects) of the 38 women graduated from the Faculty of Mechanics and Electrical Engineering, until 2018 II. In this sense, in this research, the sample is equal to the population, since there is no other group of students studying the same Programming Language I subject in the second academic semester of 2019 that constitute a greater number in the population and therefore the sample is larger. Likewise, in [15], the authors highlight that the UNTELS University is an institution with a few years of operation, specifically it was founded in 2007, and its first promotion of graduates is registered

in 2011, to date the University does not have any report or information that allows relevant actions to be taken to improve the teaching and learning process of the subject Programming Language I.

2.3. Instrument used in data collection

The instrument used in data collection is the “survey”, with questions about difficulties associated with learning object-oriented programming at UNTELS, considering the dimensions Teaching methodology, Reading materials, Previous knowledge of programming courses, Learning of concepts and Process of learning to program.

3. Result

Next, we proceed to describe the results obtained from the processing of the data collected in the survey "Learning difficulties of the Programming Language I course", of the second semester of 2019.

In Figure 1, the graphical representation of the results obtained with respect to the indicators that hinder the learning of object-oriented programming of the dimension "Teaching methodology" is shown.

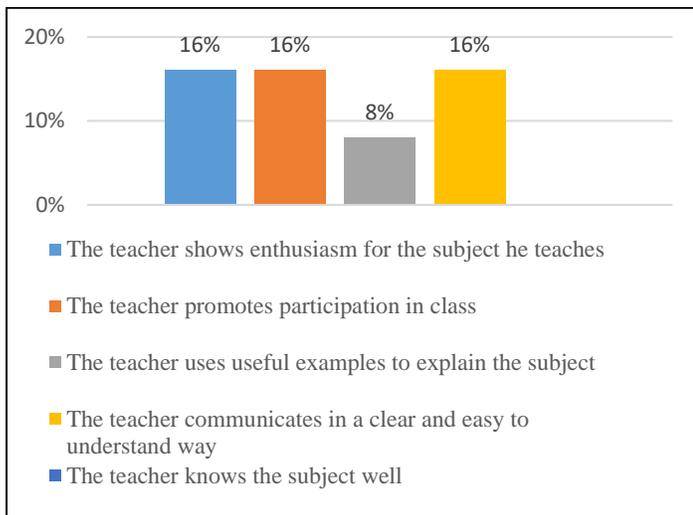


Figure 1: Higher indicators of the Teaching Methodology dimension that hinder the learning of OOP

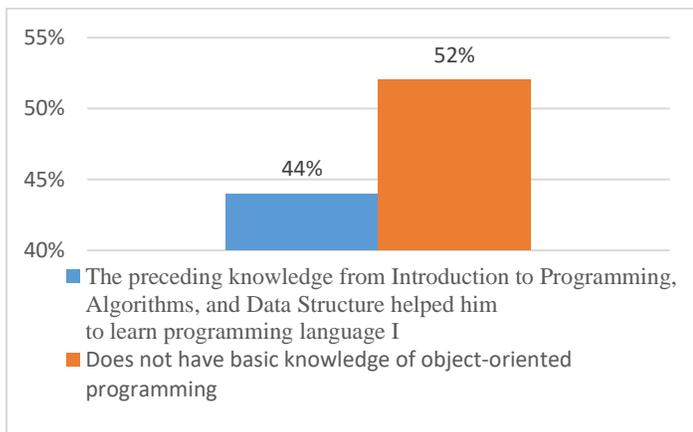


Figure 2: Indicator of the dimension Previous knowledge of programming courses that hinder the learning of OOP

As can be seen in the previous figure, the indicators that show the highest percentage are "The teacher shows enthusiasm for the subject he teaches", "The teacher promotes participation in class", "The teacher communicates in a clear and easy way understand "equivalent to 16% of the dimension" Teaching methodology "where students consider that it hinders learning OOP.

In Figure 2, the graphical representation of the result obtained with respect to the indicator that hinders the learning of object-oriented programming of the dimension "Previous knowledge of programming courses" is shown.

As can be seen in the previous figure, the indicator that shows the highest percentage is "Does not have basic knowledge of object-oriented programming", equivalent to 52% of the dimension "Previous knowledge of programming courses" where students consider it difficult to learn of the OOP.

Figure 3 shows the graphical representation of the results obtained with respect to the indicators that hinder the learning of object-oriented programming of the dimension "Reading materials".

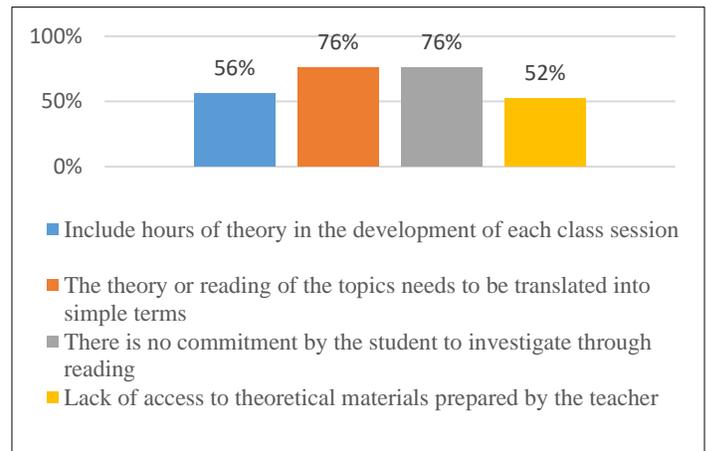


Figure 3: Higher indicators of the Reading Materials dimension that hinder the learning of OOP

As can be seen in the previous figure, the indicators that show the highest percentage are "The theory or reading of the topics needs to be translated in simple terms", "There is no commitment by the student to investigate through reading" equivalent to 76% of the dimension "Reading materials" where students consider that it hinders the learning of OOP.

Figure 4 shows the graphic representation of the results obtained with respect to the indicators that hinder the learning of object-oriented programming of the "Concept learning" dimension.

As can be seen in the previous figure, the indicator that shows the highest percentage is "Collections: Java Collection Framework, List and Array List, Map and Hash Map, Set and Hash Set, Iterator" equivalent to 60% of the dimension "Learning of concepts "where students consider it difficult to learn OOP.

Figure 5 shows the graphic representation of the results obtained with respect to the indicators that hinder the learning of object-oriented programming of the dimension "Process of learning to program".

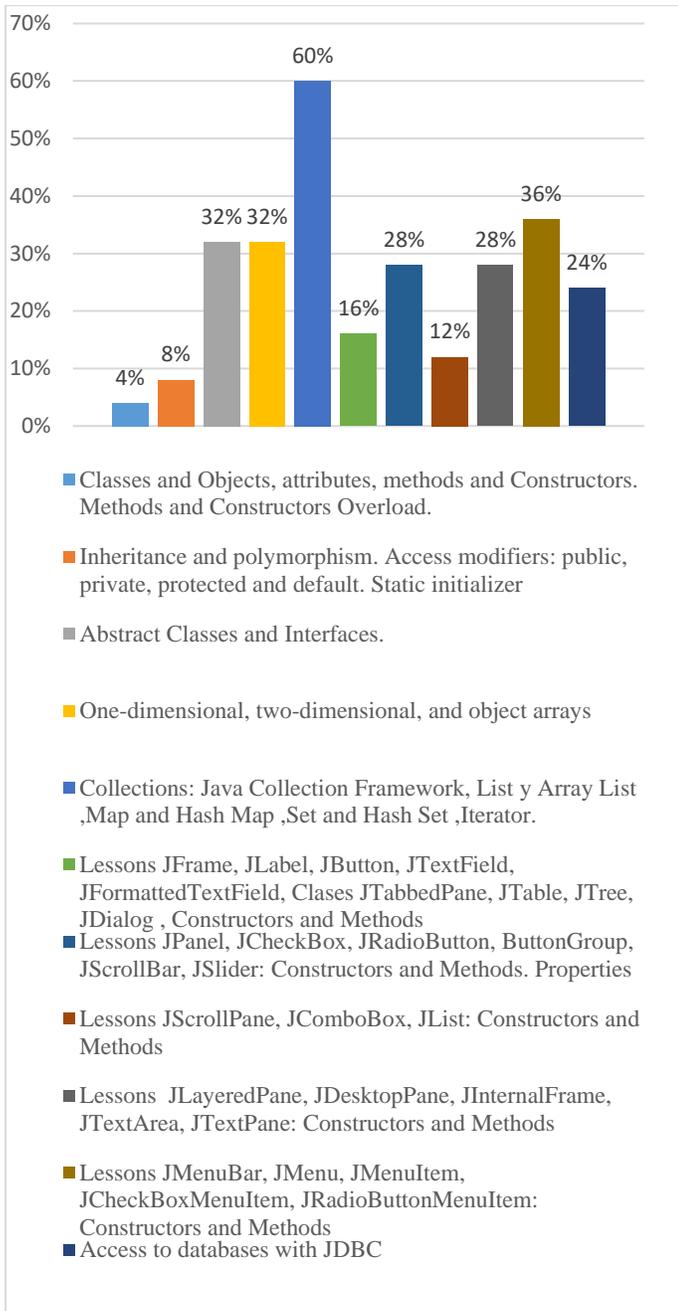


Figure 4: Highest indicator of the dimension "Learning concepts" that hinders the learning of OOP

As can be seen in the previous figure, the indicator that shows the highest percentage is "Develop programs using class libraries to create graphical user interfaces: AWT and Swing" equivalent to 48% of the dimension "Process of learning to program" where students consider it difficult to learn OOP.

To determine the dimension with the highest level of learning difficulty in OOP, we obtain the average for each dimension.

From the previous figure, we can determine that the highest percentage of difficulty associated with learning OOP is in the dimension "Learning concepts" equivalent to 8.20.

For the correlational analysis, we will classify the dimensions into two groups: Teaching Process Dimension and Program

Development Dimension. The following table shows this classification.

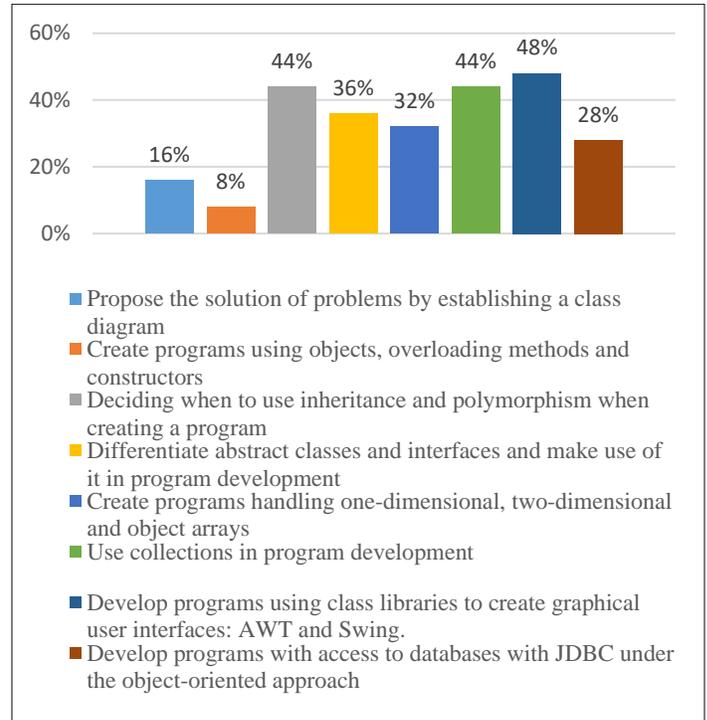


Figure 5: Highest indicator of the dimension "Process of learning to program" that hinders the learning of OOP

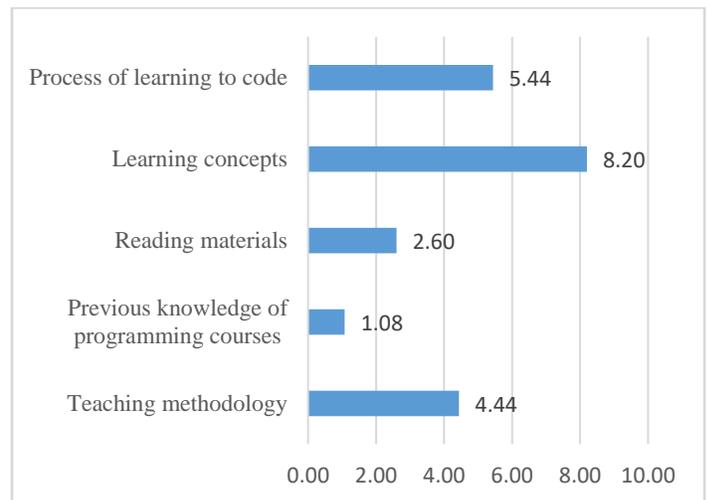


Figure 6: Average degree of difficulty for each dimension

Table 1: Classification of dimensions.

Teaching process	Program development
Teaching methodology	Process of learning to code
Reading materials	
Previous knowledge of programming courses	
Learning concepts	

In [16], the author points out to analyze the correlations, we have the interpretation of values expressed by various authors in scales, one of the most used being the Spearman rank correlation coefficient, which can score from -1.0 to +1.0 as detailed in table two

Table 2: Degree of relationship according to correlation coefficient.

Rank	Relationship
-0.91 a -1.00	Perfect negative correlation
-0.76 a -0.90	Very strong negative correlation
-0.51 a -0.75	Considerable negative correlation
-0.11 a -0.50	Mean negative correlation
-0.01 a -0.10	Weak negative correlation
0.00	There is no correlation
+0.01 a +0.10	Weak positive correlation
+0.11 a +0.50	Mean positive correlation
+0.51 a +0.75	Considerable positive correlation
+0.76 a +0.90	Very strong positive correlation
+0.91 a +1.00	Perfect positive correlation

In order to identify if there is a relationship or degree of association between the dimensions specified in Table 1, the Pearson Correlation coefficient is used, using the SPSS V25 statistical software.

Table 3: Pearson's correlation between the dimensions Learning concepts and Process of learning to program.

		Learning concepts	Process of learning to code
Learning concepts	Pearson correlation	1	0,659**
	Sig. (bilateral)		0.000
	N	25	25
Process of learning to code	Pearson correlation	0,659**	1
	Sig. (bilateral)	0.000	
	N	25	25

\*\* . The correlation is significant at the 0.01 level (bilateral).

In Table 3, we observe that the dimensions Learning concepts and Process of learning to program are directly related, that is, as there is more learning of the concepts of object-oriented programming with Java, there will be greater capacity in the process of learning to programming in Systems Engineering students, since a considerable positive correlation of 0.659 was obtained as a result.

Table 4: Pearson correlation between the dimensions Teaching methodology and Process of learning to program

		Process of learning to code	Teaching methodology
Process of learning to code	Pearson correlation	1	,235
	Sig. (bilateral)		,258
	N	25	25
Teaching methodology	Pearson correlation	,235	1
	Sig. (bilateral)	,258	
	N	25	25

In Table 4, we observe that the dimensions Teaching methodology and Process of learning to program present a mean positive correlation of 0.235.

Table 5: Pearson's correlation between the dimensions Reading materials and Process of learning to program.

		Process of learning to code	Reading Materials
Process of learning to code	Pearson correlation	1	-,014
	Sig. (bilateral)		,948
	N	25	25
Reading Materials	Pearson correlation	-,014	1
	Sig. (bilateral)	,948	
	N	25	25

In Table 5, we observe that the dimensions Reading materials and Process of learning to program are inversely related by presenting a mean negative correlation of -0.014, this implies that the number of reading materials on object-oriented programming can be increased, however, does not necessarily lead to increase their capacity of the process of learning to program in the students of Systems Engineering.

Table 6: Pearson correlation between the dimensions Previous knowledge of programming courses and Process of learning to program.

		Previous knowledge of programming courses	Process of learning to code
Previous knowledge of programming courses	Pearson correlation	1	0,036
	Sig. (bilateral)		0,863
	N	25	25
Process of learning to code	Pearson correlation	,036	1
	Sig. (bilateral)	0,863	
	N	25	25

In Table 6, we observe that the dimensions Previous knowledge of programming courses and Process of learning to program present a mean positive correlation of 0.036.

#### 4. Discussion

In relation to the results of the present investigation, the following discussions are held below:

In [17], the author points out that starting in the world of object-oriented programming is a complex task for many students, the teacher plays an important role in student motivation, as evidenced by this research in its indicator "The teacher promotes participation in class", where 16% of the students consider that it hinders their learning of OOP, it can be said that this result is consistent with the research entitled "Initiatives to motivate Programming students" in which it is concluded that motivation is low due to the novelty of the subject, the difficulty of students in learning abstract concepts and the lack of capacity for continuous work.

In [1], the author describes regarding the indicator "The theory or reading of the topics needs to be translated into simple terms",

76% of the students consider that it hinders the learning of Object-Oriented Programming, coincides with the thesis doctoral degree entitled "Virtual environments based on active learning for the teaching of object orientation" in which he concludes that it is necessary for those who are starting in this paradigm to reach a clear understanding of its concepts, abstract in most cases, which represent in software development.

Regarding the indicator that students "do not have basic knowledge of object-oriented programming" before taking the Programming Language I course, 52% indicate that it makes learning difficult for them, this evidence agrees with the study entitled "Methodology of teaching by learning object-oriented programming logic" pointed out in [18] by the author, in which he concludes that there is a tendency to use Java as the first language directly with the object-oriented concept, without deepening the development of the bases logic of programming, this can be very harmful, because we are going to generate good programmers to code using languages, but without logical bases, that is, programmers who do not know how to program.

Regarding the indicator "Collections: Java Collection Framework, List and Array List, Map and Hash Map, Set and Hash Set, Iterator", 60% of the students consider a difficult topic to understand since it is related to classes, interfaces, inheritance, polymorphism for its understanding can be considered consistent since it coincides with the study entitled "Learning object-oriented programming: Experiences in technical and upper secondary education" indicated in [19] by the author, in which it is concluded that the subject of greater difficulty of understanding for the control groups is inheritance, while for experimental groups it is polymorphism.

## 5. Conclusions

It has been possible to determine the main difficulties associated with the learning of object-oriented programming (Programming Language I) in the students of the School of Systems Engineering of the National Technological University of Lima Sur, which are "The teacher shows enthusiasm for the subject he teaches", "The teacher promotes participation in class", "The teacher communicates in a clear and easy to understand way" with 16% indicators related to the teaching methodology". The indicators "Lack of translating the theory or reading of the topics into simple terms", "There is no commitment by the student to investigate through reading" with 76% related to the reading materials. The indicators "Does not have basic knowledge of object-oriented programming" with 52% related to Have prior knowledge. The indicator "Collections: Java Collection Framework, List and Array List, Map and Hash Map, Set and Hash Set, Iterator" with 60% in relation to Learning concepts. The indicator "Develop programs using class libraries to create graphical user interfaces: AWT and Swing" with 48% related to the Learn to program process.

It has been possible to determine the dimension that has the greatest learning difficulty of Object-Oriented Programming, which is "Learning concepts" with an average of 8.20, we can mention that this expresses that currently the curricular plan related to programming courses at school Systems Engineering do not have hours of theory to help manage fundamental concepts of Object Oriented Programming.

## 6. Recommendations

While developing the Programming Language I (OOP) course in its entirety in a practical way using the Java language, it is recommended to develop manuals or self-instructional videos that help the student understand concepts of object-oriented programming.

Since the curricular plans at the Universidad Nacional Tecnológicas de Lima Sur are updated every three years, it is recommended to include hours of theory in programming courses or, failing that, develop intelligent applications that help understand object-oriented programming concepts.

## References

- [1] G. Jiménez, Virtual environments based on active learning techniques for teaching object-orientation, doctoral thesis, Complutense University of Madrid, Spain, 2018.
- [2] I. Friss de Kereki, What is object-oriented programming? A phenomenographic approach. Educational Research Notebooks, 1998.
- [3] D. Gayo, A. Cernuda del Rio, J. Cueva, M. Diaz, M. Garcia, J. Redondo, Reflections and experiences on the teaching of OOP as the only paradigm, University of Oviedo, Spain, 2003.
- [4] J. Rios, Object-oriented programming in non-object-oriented languages: C, An Experience. Scientia Et Technica, **21**(29), 107-111. Technological University of Pereira, 2005.
- [5] S. Rueda, I. Señas, A problem-solving approach to teaching OOP. National University of the South - Bahía Blanca, 2008.
- [6] H. Flórez, Object-oriented programming using Java. Bogotá: Ecoe Ediciones, 2012.
- [7] J. Insuasti, Teaching and learning problems of programming fundamentals. Education and Social Development Magazine, **10**(2), 234-246, 2016. DOI: org / 10/18359 / reds.1701.
- [8] J. Sánchez-García, M. Urías-Ruiz, B. Gutiérrez-Herrera, Analysis of the Learning Problems of Object-Oriented Programming, Scientific article in Ra Ximhai, **11**(4): 289-304, 2015.
- [9] J. Angulo & O. Palomar, LM1: a study methodology for the subject "Programming 1". Educere: Venezuelan Journal of Education, **73**, 635-648, 2018.
- [10] J. Fuentes-Rosado & M. Moo-Medina, Difficulties of learning to program. Engineering education magazine, **12**(24), 76-82, 2017. <https://doi.org/10.1093/ptj/79.4.371>
- [11] M. Sánchez, University dropout in students from a private university in Iquitos. Digital Journal of Research in University Teaching, 60-83, 2012.
- [12] R. Carvajal & C. Cervantes. Approaches to university dropout in Chile. Educação e Pesquisa, 44, 2018.
- [13] R. Hernández, C. Fernández, M. Baptista, Research Methodology, Mexico: Mcgraw-Hill / Interamericana editores, 2010.
- [14] D. Arce, O. Chamorro, Y. Leandro, J. Castro, E. Martínez, The Proportionality of Women Graduated from the Professional Career of Mechanical and Electrical Engineering at UNTELS: Analysis of their Academic Performance and Labor Field of Action . Advances in Science, Technology and Engineering Systems Journal, **5**(1), 368-372, 2020. <https://dx.doi.org/10.25046/aj0501273>
- [15] O. Chamorro, T. Diaz, D. Arce, J. Castro, D. Barrientos, A. Roman-Gonzalez, Satisfaction of the graduate for the continuous improvement of educational quality in unteles. Advances in Science, Technology and Engineering Systems Journal, **4**(5), 151-157, 2019. <https://dx.doi.org/10.25046/aj040503>
- [16] M. Mondragón, Use of Spearman's correlation in an intervention study in physiotherapy. Mov.cient.**8**(1): 98-104, 2014. <https://doi.org/10.1093/ptj/79.4.371>
- [17] M. Salamó, J. Camps, C. Vallespí, D. Vernet, X. Llorà, E. Bernadó & X. González, Initiatives to motivate Programming students, Proceedings of the VII Conference on University Teaching of Informatics (JENUI 2001), 329, 2001.
- [18] L. Román, Methodology for teaching and learning the logic of object-oriented programming.
- [19] R. Tabares & H. Gómez, Learning object-oriented programming: Experiences in technical and higher secondary education.