Digital Competencies of the Future Programmer: SWIFT

Chernova E.A., Nazarov A.D.*

Ural State University of Economics, Ekaterinburg, Russia
**Corresponding author. Email: nazarov_usue@mail.ru*

ABSTRACT
Using programming tools is considered not enough for a fully functioning and development in transformation of the economy. The article discusses the innovative Swift programming language. Some programming languages are analyzed. The concept of multiparadigmality is revealed. The relevance of the use of a new language in the context of the digital economy is indicated.

Keywords: *digitalization of the economy, programming languages, Swift programming language*

1. INTRODUCTION

During the period of digitalization of the economy, there is a shortage of qualified personnel with modern competencies such as programmers, coders, Big Data specialists, and e-commerce specialists. In modern science, the problem of IT education is becoming especially acute. It is enough to have computer literacy and fundamental knowledge but not the application focus to go to university, therefore it is difficult for students to learn when it comes down to this sphere. Unfortunately, not everyone can cope with such a pace of mastering programming languages. Someone starts learning all the languages at once or chooses just by guessing, and someone completely reorganizes the process of learning. In our opinion, one of the most significant issues is the controversial question of what the programming starting points are? We suppose it is the choice of programming language. Programmers usually look for the most popular and highly paid ones. However, it is more important to choose not fashionable and familiar languages, but user friendly, functional and clear. That is why the difficult programming languages are often chosen. Programmers often consider Java, C ++ and similar ones without hesitation. This is understandable because they are in the top of the highest paid, demanded and popular languages. It is difficult for beginners to comprehend these languages, therefore, they use Basic, Pascal in most cases. Then it is possible to expand the knowledge.

2. COMPARATIVE ANALYSIS

There are hundreds of programming languages, that is why it is important to take seriously the choice of the appropriate one. Let us turn to the description of some languages.

Table 1 Analysis of the leading programming languages for initial learning

	Syntax and semantics	**Pradigm**	**Application**	**Acquisition**
Python	✓ Minimalistic ✓ Consistent ✓ Easy to read ✓ There are no operator brackets to reduce the number of lines and characters — Different assignment semantics	Multiparadigm	To create application programs For WEB development For training beginner programmers	Simple language, the ability to learn without any skills
JavaScript	✓ In many respects to C and Java syntax — Many errors	Multiparadigm	It is used to create interactive web pages and applications. A modern browser is not without JavaScript support	More advanced skill level. When mastering knowledge of HTML and CSS is also required
C#	✓ Practically does not differ from the C programming languages ✓ A small piece of the code ✓ Generates part of the code itself	Object-oriented Language	Create: Desktop applications Server applications and websites Mobile applications Console applications and utilities Universal applications for tablets, smartphone	Ability to study without specific skills

Continuation of Table 1.

Swift	✓ Easy to read ✓ Error-resistant ✓ Based on Objective-C ✓ Minimalistic	Multiparadigm	It can be used to create applications for desktop, mobile applications, servers	It is added to curriculum, easy to learn, easy to start professional development
Java	✓ There are not global functions or variables ✓ The code refers to classes, and all values are objects ✓ Based on C ++ but it is easier — Long code — Contains unnecessary information	Multiparadigm	Creating: Android applications Software and financial products Cash terminals Programs for working with big data	The actions are prescribed in the English language, it is easy for beginners A large amount of literature and courses
Pascal	✓ There is little ambiguity ✓ Intuitive — Outdated language — Limited functionality	Imperative Structured language	It is used in industrial programming, teaching programming in high school; It is the basis for a large amount of other languages	The basic language for teaching, one of the simplest languages

In addition to the presented programming languages, there is a significant amount of the relevant languages at the time of the information technologies development. However, of course, new options will be created with the use of the traditional languages over time.

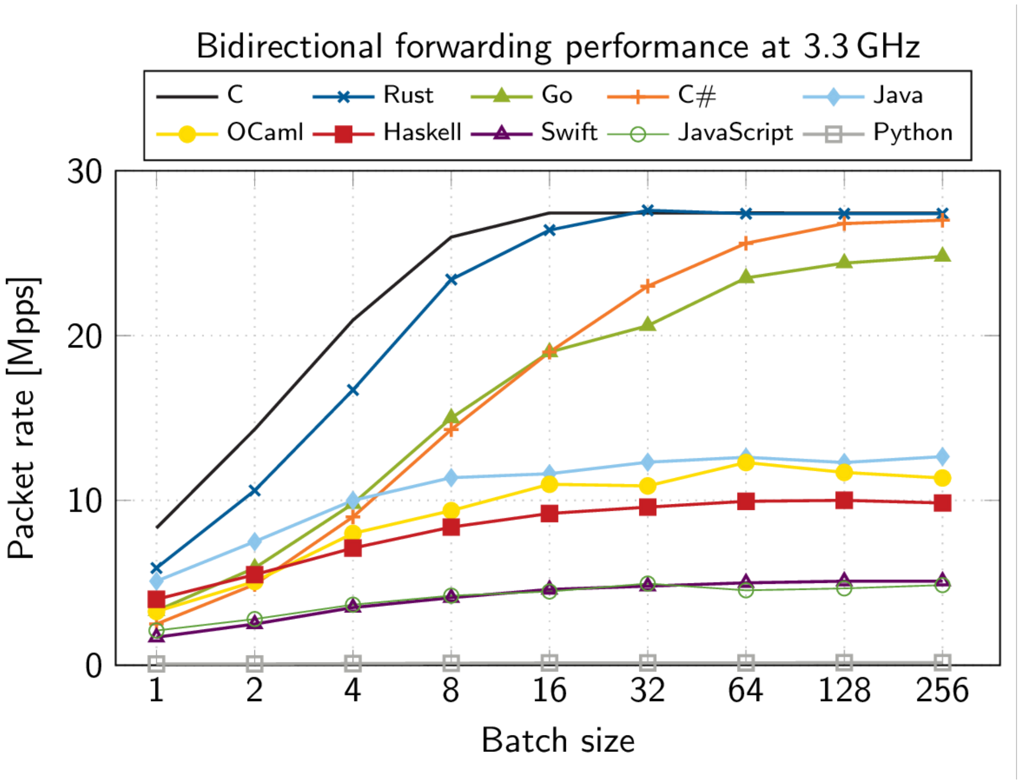

Figure 1 Biderctional forwarding performance at 3.3 GHz

3. RESULTS AND DISCUSSIONS

There are particular perspectives in the study of the multi-paradigm programming languages. The visual capabilities that were originally taken from several, most often unrelated languages are the basis of the multi-paradigm programming language. There is using the multiple paradigms in a single application. In our opinion, it is worth focusing on the newest language Swift now.

Swift is a general purpose language created in 2014 by Apple and is gaining momentum. In the context of the dynamic development of IT technologies, it was necessary to improve some aspects, in this regard, secure programming patterns are used and modern functions have been added. Performance with new features exceeds many other programming languages. Swift functions are designed to work together to create the powerful, interesting in term of exploit language. Its code is cleaner and less error-prone, and the modules remove headers and provide namespaces.

Figure 2 Logo “SWIFT”

The memory is automated and there is no need to enter a semicolon. Swift also borrows from other languages, for example, named parameters carried over from Objective-C are expressed in clean syntax that makes Swift API easy to read and maintain. The Swift code may work together with the code written in the C and Objective-C programming languages within the same project. Swift is designed to make it easier for a developer to write and maintain correct programs as well as for novice programmers to use. In 2015, the helper libraries appeared, and Swift.org was created for this project. Despite the fact that Apple focuses on wealthy consumers and has recognized high-tech prestige items, it is distributed under an open license. Anyone can use this programming language; there is a significant community of programmers who support beginners to develop this language. According to Stack Overflow, Swift is the 4th most favorite language among expert developers. At the same time, it has reached the top 10 most popular programming languages in just four years.

4. CONCLUSION

The future of the Swift programming language has not been determined. It will be able to either spread around the world and become the basis of information development, or it will specialize in iOS development depending on changes in the digitalization of the economy. In any case, dealing with this programming language is relevant now. On average, the income of Swift developers is more than other operating systems developers generate due to the need to motivate specialists who create programs for Apple devices and strict quality requirements.

REFERENCES

[1] F. Bergenti, E. Iotti, S. Monica, A. Poggi, Agent-oriented model-driven development for JADE with the JADEL programming language, Computer Languages, Systems and Structures, 50 (2017) 142-158. DOI:10.1016/j.cl.2017.06.001

[2] R.Z. Cabada, M.L.B. Estrada, F.G. Hernández, R.O. Bustillos, C.A. Reyes-García, An affective and web 3.0-based learning environment for a programming language, Telematics and Informatics, 35(3) (2018) 611-628. DOI:10.1016/j.tele.2017.03.005

[3] B. Carpenter, A. Gelman, M.D. Hoffman, D. Lee, B. Goodrich, M. Betancourt, A. Riddell, Stan: A probabilistic programming language, Journal of Statistical Software, 76(1) (2017). DOI: 10.18637/jss.v076.i01

[4] F.T. Chong, D. Franklin, M. Martonosi, Programming languages and compiler design for realistic quantum hardware, Nature, 549(7671) (2017) 180-187. DOI:10.1038/nature23459

[5] M. Felleisen, R.B. Findler, M. Flatt, S. Krishnamurthi, E. Barzilay, J. McCarthy, S. Tobin-Hochstadt, A programmable programming language, Communications of the ACM, 61(3) (2018) 62-71. DOI:10.1145/3127323

[6] J. Good, K. Howland, Programming language, natural language? supporting the diverse computational activities of novice programmers, J. of Visual Languages and Computing, 39 (2017) 78-92. DOI:10.1016/j.jvlc.2016.10.008

[7] J. Hao, T.K. Ho, Machine learning made easy: A review of scikit-learn package in python programming language, J. of Educational and Behavioral Statistics, 44(3) (2019) 348-361. doi:10.3102/1076998619832248

[8] N. Mehta, M.V. Devarakonda, Machine learning, natural language programming, and electronic health records: The next step in the artificial intelligence journey? J. of Allergy and Clinical Immunology, 141(6) (2018) 2019-2021. DOI:10.1016/j.jaci.2018.02.025

[9] M. Mladenović, I. Boljat, Ž. Žanko, Comparing loops misconceptions in block-based and text-based programming languages at the K-12 level, Educ. and Information Technologies, 23(4), (2018) 1483-1500. DOI:10.1007/s10639-017-9673-3

[10] O.O. Ortiz, J.A.P. Franco, P.M.A. Garau, R.H. Martin, Innovative mobile robot method: Improving the learning of programming languages in engineering degrees, IEEE Transactions on Education, 60(2) (2017) 143-148. DOI:10.1109/TE.2016.2608779

[11] B. Ray, D. Posnett, P. Devanbu, V. Filkov, A large-scale study of programming languages and code

quality in GitHub, Communications of the ACM, 60(10) (2017) 91-100. DOI:10.1145/3126905

[12] P.P. Ray, A survey on visual programming languages in internet of things, Scientific Programming, (2017). DOI:10.1155/2017/1231430

[13] S. Steiner, J. Wolf, S. Glatzel, A. Andreou, J.M. Granda, G. Keenan, L. Cronin, Organic synthesis in a modular robotic system driven by a chemical programming language, Science, 363(6423) (2019) DOI: 10.1126/science.aav2211

[14] C. Teng, J. Chen, Z. Chen, Impact of augmented reality on programming language learning: Efficiency and perception, J. of Educational Computing Research, 56(2) (2018) 254-271. DOI: 10.1177/0735633117706109

[15] X. Wang, C. Tian, Z. Duan, L. Zhao, MSVL: A typed language for temporal logic programming, Frontiers of Computer Science, 11(5) (2017) 762-785. DOI:10.1007/s11704-016-6059-4