

Informatics for Library and Information Science students with special focus on Python

UDC 004.43PYTHON

DOI 10.18485/infotheca.2018.18.1.4

ABSTRACT: Students of Library and Information Science (LIS students) at the Faculty of Philology in Belgrade attend thirteen different obligatory computing courses during the four-years period of their undergraduate studies. In this paper we highlight the content of the programming courses held during the fifth semester. Using Python programming language, students are introduced to basic programming concepts, data structures and their different applications. We also make a brief preview of other computing courses and suggest how the gained knowledge could be applied in the field of information technologies and natural language processing.

KEYWORDS: Python, programming, information technologies, natural language processing, information retrieval

PAPER SUBMITTED: 16 April 2018

PAPER ACCEPTED: 2 June 2018

Branislava Šandrih

branislava.sandrih@fil.bg.ac.rs

University of Belgrade, Faculty of Philology, Serbia

1 Introduction

The first two generations of librarians in Serbia were educated in the early sixties of the twentieth century, when the Department of Library Science was founded at the Faculty of Philology in Belgrade. After the education reform in 1978, the Faculty of Philology re-introduces a librarianship module, which includes four-semester studies with regular four-year studies in one of the groups of the Faculty of Philology. From 1990/91 the Department of Library and Information Science is being re-established, at the four-year study level. More about the history and the emergence of the Department as it is today is described in (Vraneš, 2008).

Teaching staff within the Department provides students with knowledge in various fields: librarianship, archivistics, museology, as well as mathemat-

ics and informatics. In this paper we will focus on courses that deal with the concepts of algorithmics and programming.

This paper is organized as follows. In Section 2, some of the discussions whether Python¹ is the most suitable as the first programming language for beginners of different age are listed and described. We also offer a brief insight in the current status of programming classes in elementary schools in Serbia, as well as the role of Python in this process. As an introduction to the content of the programming course itself, Section 3 contains a summary of computing courses that preceded it, in order to gain insight into the prior knowledge that students acquire, while the programming course itself is described in Section 4. Computing courses that follow are listed and briefly described in Section 5. In Section 6 a special analysis of the application of the acquired knowledge is discussed, both in further education and later, within in the labor market. We also analyze the overall knowledge gained during these studies, and present the possible work positions for which these students acquired competencies. We state the final conclusions in Section 7.

2 Python as a first programming language for LIS students

Discussions about which programming language represents the best candidate for beginners have been going on for many years now. It is obvious that the choice depends on several factors: the age of the person who begins to learn programming, prior knowledge and experience in the field of computing, level of English language usage, as well as the expected field of application.

Over time, different programming languages were considered preferable candidates for beginners. In the 1980s, it was BASIC and Pascal, and in the late 1990s and at the beginning of the 2000s it was the programming language Java (Hosch, 1996; King, 1997; Hadjerrouit, 1998). Authors in (Radenski, 2006) change the program of the first programming course in favor of Python and conclude that it is much easier for students to adopt the syntax of this programming language than Java. As the use of ready-made modules is more practical in Python, students develop even more complex applications than those developed by the preceding generations, who coded in Java. The same question still exists, after many years. Authors

¹ Python is a high level, interpretive programming language of general purpose. It was patented by Guido van Rossum in 1991

in citepAmbikesh2011 divided participants in the initial, high-school programming course in two groups. The first group learned Java as the first programming language, i.e. they adopted the object-oriented paradigm as their first paradigm. The second group was first introduced to the basic concepts (expressions, conditional clause, loops) in Python, and then joined Java students. Upon course completion, a comparison of the success of these two groups was made. It turned out that the second other approach (Python with Java) gave a better outcome. Another question these authors asked was: Should beginners be taught the most popular programming language at that moment, or rather the one preferred in the labor market? Over time, being not only objectively suitable for beginners, Python become the language of the industry. This is a consequence of the expansion of artificial intelligence in many fields and the role that Python had in its development. The mean incentive were the modules: *scikit-learn* (Pedregosa et al., 2011) for machine learning algorithms, *tensorflow* (Abadi et al., 2016) for handling multidimensional matrices, *nltk* (Bird and Loper, 2004) for natural language processing (NLP) and many others.

Another active discussion is the one concerning the programming paradigm that will be initially introduced to beginners. Authors in (Goldwasser and Letscher, 2008) believe that this should be an object-oriented paradigm, but they also recommend Python as a language, which supports this paradigm in addition to procedural and functional. Basic concepts of programming and basic data structures are introduced to students at the University of Witwatersrand using the Scheme programming language (Sanders and Langford, 2008). This programming language supports the functional programming paradigm, which is somewhat less present in applications for everyday use. Deeming it thus unpractical, students who already had programming experience did not accept this programming language well. In the subsequent academic year, the same course was held using Python. It turned out that Python was now well adopted by both beginners and by students who already had previous programming experience.

In papers (Agarwal and Agarwal, 2005; Vasljević, 2013) the syntax of the Python programming language was analyzed, as well as the availability of tools for its application in teaching. The authors (Ranum et al., 2006) present their experiences with Python as part of an initial programming course. They conclude that Python has proved to be very convenient for this purpose, primarily because of its simple syntax and availability of development tools

Given the conclusions of the aforementioned authors, as well as their experiences, Python was selected as the initial programming course to be

taught within the studies of librarianship and informatics at the Department of Library and Information Science at the Faculty of Philology in Belgrade. One of the important factors that led influenced this decision was the existence of the module nltk for the processing of natural languages, which can be particularly useful for students of the Faculty of Philology.

Another reason for this decision is related to the content of informatics courses in prior education. Starting from 2017/18, a course named “Informatics and Computing” was for the first time introduced as obligatory in elementary schools in Serbia. Members of the “Petlja” Foundation² actively contribute to the modernization and popularization of informatics and computing in Serbia. The aim of the Foundation is to raise the level of awareness in the society regarding the general educational importance of algorithmic literacy, as well as to encourage the society to actively work towards the development of algorithmic literacy and programming. The contribution in the form of resources for teaching “Informatics and Computing” in the fifth and sixth grade is one of the prominent activities of this foundation. According to the curriculum, the course also includes compulsory programming lessons, which start already in the fifth grade, using Scratch³. For this purpose, an interactive handbook⁴ and practicum⁵ have been developed.

In the subsequent year, students switch to a textual programming language of teachers choice. In 2018/19 the first generation will attend programming classes in a textual programming language, according to the new program. Members of the “Petlja” Foundation believe that teachers should choose Python, and they offer a collection of resources helpful to both, the teachers and the six-graders. These materials include an interactive manual,⁶ an accompanying collection of algorithmic tasks,⁷ and a collection of short test questions.⁸

This also influenced our decision regarding the structure of the programming courses at the Faculty of Philology. Upcoming generations will already posses an algorithmic way of thinking, as well as a thorough knowledge of the syntax. With its curriculum, the study program of Library and Informatics

² Petlja Foundation (on-line)

³ Visual Programming Language, *Scratch* (on-line)

⁴ Programming in Scratch, a fifth grade manual (on-line)

⁵ Programming in Scratch, a fifth grade practicum (on-line)

⁶ Programming in Python, a sixth grade manual (on-line)

⁷ Programming in Python, a sixth-grade algorithmic assignment collections (on-line)

⁸ Programming in Python, a collection of short questions for the sixth grade (on-line)

is well prepared to welcome new generations, and continue to develop their existing programming skills.

3 Informatics courses in early semesters

In this section, we make a brief overview of computing courses that LIS students attend during the first four semesters. The list of courses and their curricula is available [on-line](#). From the very beginning of the studies, within an unique study program, computing subjects are adequately included (see (Krstev, 2002) for more details).

During the first two semesters students attend a course named “Informatics for librarians”. At the beginning, it covers different numeral systems, as well as conversion algorithms between different numeral systems. Afterwards, basic concepts of a set, a relation, an operation, a function, and a sequence are being introduced. Finally, the last part of the course is dedicated to mathematical logic. This course establishes the necessary foundation required for the adoption of basic concepts in information science (starting from the binary system and its application in computational arithmetic, through functions and relations, required for understanding the concept of, e.g. relational databases). Slowly the process of designing a programmatic way of thinking among students begins.

Students simultaneously attend the course “Informatics practicum”, which, as its final outcome has the understanding of the design and principles of modern computers, familiarity with the basic hardware characteristics, as well as acquiring basic knowledge about different software types and training for their active and creative use in everyday life and subsequent professional work.

It is important to emphasize the relation and synchronization of these two subjects. In the practical part of the “Informatics Practicum”, students are trained for the advanced use of Word and Excel software from the Microsoft Office package. After the adoption of the concept of relations and mathematical operations within “Informatics for librarians” course, students can observe practical application of arithmetic and relational operators within Excel. Afterwards, the notions of function and composition of functions are being introduced, which are practically demonstrated through solving problems using built-in functions in Excel. Concepts of mathematical logic are also followed by their practical application using logical operators and built-in logical Excel functions.

In the digital era, representation and processing of written text in a computer are very attractive and important topics for philology, and especially LIS students. This is precisely the scope of the subject named “Digital Text”, taught during the third and fourth semester. The first topic of the course covers the lowest, bit-wise level of text representation in a computer. Subsequently, the representation of different alphabets and symbols, using various code schemes and standards is thoroughly studied.

Theoretical lectures are followed by practical classes, where students are trained for the advanced use of various text preparation and formatting systems. The first such system, L^AT_EX,⁹ has the greatest application for the preparation of scientific texts. Unlike Word, which is the what-you-see-is-what-you-get type of text processor, in L^AT_EX an author performs formatting separately from entering content. Easy formatting is accomplished using the already defined commands and environment, which provides a uniform and almost completely controlled layout of the document. This text processor encourages modularity, in the sense of separating different logical units in different files, by using existing or by writing new macros. Its use already represents some form of programming (Šandrih, 2016).

Students continue to practice the mechanism for entering plain text first, and dealing with its formatting afterwards, by learning mark up languages and languages for text formatting for display on the Web. The course covers the creation of content of web pages using HTML, and their styling using CSS. Within the course there are two obligatory practices that are realized at the University Library “Svetozar Markovic” in Belgrade. The first practice involves the use of various Google services, while within the second, students are trained to create pages in Wikipedia, which at the same time introduces the third mark up language taught in this course.

With such experience gained, it is safe to say that, after the first four semesters, students have a strong informatics foundation, but also acquire a basic programming intuition, which should be further encouraged.

4 Programming classes

During the fifth semester, programming is taught in two related courses: “Structure of Information 1” and “Informatics Practicum 3”. The curriculum for the first course is available [on-line](#) and includes: representing text and numbers in a computer, concepts of algorithms, operations, expressions,

⁹ [Text Editor L^AT_EX \(on-line\)](#)

if-clause, loops, strings, recursion, and search and sort algorithms. Beside pseudo-code, accompanying examples are also demonstrated in the code written in the programming language Python.

Simultaneously, the “Informatics Practicum 3” course covers all the above mentioned concepts, but in a somewhat more practical context. The course curriculum is available [on-line](#). This course covers the following topics: data types (primitive types, lists, sets, dictionaries), arithmetic and logical expressions, string formatting, if-clause, loops, various built-in functions, search and sort algorithms.

During the first thirty-six hours of the course, the so-called “literate paradigm” is applied, first introduced by Donald Knuth ([Knuth, 1984](#)). Parts of programming code are separated into smaller logical units, intertwined with explanations written in some natural language. This approach is especially practical when it comes to Python, thanks to [Anaconda¹⁰](#), which includes [Jupyter¹¹](#), an interactive development environment that contains executable code snippets along with explanations, comments, tables, images etc.

It is important to mention a related graduate study program at the University of Belgrade, “Social Sciences and Computing”.¹² The aim of these studies is to provide bachelors of social sciences with knowledge that will enable them to apply modern computing technologies in their area of expertise. Within the course “Programming for linguists”, the literate paradigm is applied for teaching Python and its application to NLP. Teaching staff ([Stanković et al., 2017](#)) points out the good sides of this approach and records successful outcome.

During the remaining sixteen teaching hours, students make stand-alone desktop applications using the PAGE module.¹³ These Desktop application are mostly aimed at text processing.

At the Faculty of Philology, Python was taught for the first time in the winter semester 2017/18.¹⁴ This teaching plan can be considered experimental and subject to changes. Based on the experience we gained, we state our goals for the upcoming years in the next chapters.

¹⁰ [Python distribution, *Anaconda* \(on-line\)](#)

¹¹ [Interactive Python development environment, *Jupyter* \(on-line\)](#)

¹² [Social Sciences and Computing \(on-line\)](#)

¹³ [Development module for desktop applications, *Page* \(on-line\)](#)

¹⁴ Students have the opportunity to take an exam in both, Python and Visual Basic

5 Informatics courses in latter semesters

Along with the above mentioned subjects, in the same semester, students attend the “Statistics in libraries” course. The aim of this course is acquiring basic theoretical knowledge and skills necessary for statistical data analysis. Within this course, the following topics are covered: basic concepts of statistics (notions of population, variables, measurements, measuring instruments, measures, measuring scales, samples, distribution and sample size), measures of average and variability, percentiles and percentile ranges, introduction to theory of probability (concepts of experiment, outcome, probability of an event, concept of distribution of discrete and continuous variables, standard normal distribution, confidence intervals), correlation coefficients, determination of statistical significance and reliability of statistical measures.

In the next, sixth semester, students take compulsory course “Informatics Practicum 4”, which covers the following topics: markup languages XML and JSON, DTD and XML Schema for XML documents structure manipulation, Bib \TeX format for bibliography management in L \TeX , as well as regular expressions (Утвић, 2003). The course is supported by a variety of additional activities — besides the plethora of practical tasks that should be completed by students during classes, there are also two seminar papers. With the acquired skills, various types of documents are prepared and annotated in accordance with a given DTD scheme, which represents the task for the first seminar paper. For the second seminar paper, the task is to prepare a bibliography a the scientific paper of a particular journal or conference proceedings using Bib \TeX , which is a moderately light challenge to LIS students, thanks to the knowledge gained in courses related to librarianship.

Along with this course, students also attend the course of “Information Structure 2”. It covers the following topics: abstract data structures (lists, trees and graphs), XML as the general standard meta-language for various documents and other types of information markup, UNIMARC as a specific library standard for presenting catalog and bibliographic information, a universal digital documents description language known as the Dublin Core, the TEI standard for marking digital text, semantic web and RDF.

In the next, seventh semester, students attend the course “Database and Library Information Systems”, which covers basic models in databases development, detailed insight in the relational model, the use of the Access program within the MS Office package, the design of databases and complex queries .

In the last semester of undergraduate studies, there are two more obligatory computing courses: "Multimedia documents" and "Information retrieval". Knowledge acquired in the field of librarianship and informatics is creatively expressed within a joint project of the former course. The concept and structure of this course are described in (Krstev and Trtovac, 2014), and more about some of the previous students' projects can be read in (Lazić and Poklopić, 2011; Kovrlija et al., 2012; Obradović et al., 2016) and (Stošić et al., 2018).

A lot of high-quality and important knowledge in the field of information retrieval (IR), whose specialists are today highly sought-after, is provided by the course with the same name, which covers the following topics: use of inverted indexes, Boolean and vector queries and their extensions, optimization of operations on inverted-indexes, evaluation of information retrieval systems, string matching algorithms, automatic indexing of individual terms, automatic calculation of term weights, use of thesaurus, etc.

The Department of Library and Information Science also provides non-mandatory "Language Technologies", another two-semester undergraduate computer science course. The aim of this course is to familiarize students with language technologies, especially with language tools and resources, the current international standards for their implementation, and to train them for their active use. The focus of this course is the usage of existing language tools for the Serbian language.

We also briefly describe computer science courses that can be chosen by Master students.

The aim of the course "Advanced Language Technologies" is to review the basic methods in the processing of natural languages, concluding with syntax parsing. The course focuses on the usability and applicability of these methods into the Serbian language. The classes of the course "Electronic Publishing and Digital Libraries" are designed to introduce students to forms of electronic publishing and its implications on the work of modern libraries. The aim of the course is to familiarize students with the types of digital libraries, the ways of their creation, as well as with the possibilities of connecting with traditional collections. "Designing and Maintaining Web Content" introduces students to the basic forms and characteristics of Web content preparation and maintenance, Wiki-technologies, and the use of content management systems. The course "Advanced Methods in Information Retrieval", building up on the knowledge acquired the undergraduate course, outlines the concepts of advanced information retrieval systems, as well as of systems on which they are based: indexing, classification, cluster-

ing, summarization and information extraction. Finally, the content of the “Bibliometrics” is suitable for students who wish to continue doing research professionally, as well as to librarians from scientific research organizations whose task is to assist researchers in evaluating their own work.

6 Application of Python

The joint agreement of the subject teachers is that, starting from winter semester 2018, the course “Informatics Practicum 3” should tackle more practical problems from statistical processing mentioned in “Statistics in libraries” classes. Beside built-in functions for solving basic statistical tasks in Python (mean, variation, deviation, different correlation coefficients, etc.), there is also *scipy*,¹⁵ an ecosystem of free, open-source mathematical modules that enable more advanced statistical processing.

Usage of existing Python modules for the processing of already annotated texts would be a novelty for the “Informatics practicum 4” course. One of the interesting modules for working with XML documents is *BeautifulSoup*,¹⁶ which helps manipulating and extracting the content of marked elements. Another application of newly acquired knowledge within the same course could be the extraction of data from Web pages. Students interested in computational linguistics could compile corpora in this way, and then process that content using the already mentioned *nltk* module.¹⁷

Presentation of the document as a Bag-of-Words (BoW), as well as automatic calculation of term weights using *tf-idf* (term frequency - inverse documents frequency), are options supported by the *scikit-learn* module.¹⁸ The first generation of students who enrolled in the “Information retrieval” course with previously acquired programming skills in Python, could make, for example, a comparative analysis of the various literary works of an author. Every story or a novel (i.e. text document) can be presented as a B-o-W, and then the terms present could be weighted using *tf-idf* measure in order to determine the most discriminate terms per document. The terms obtained could be compared for different literary works, and the results could then be discussed. A similar analysis can be also applied to the Web - for

¹⁵ Python statistical modules, *scipy* (on-line)

¹⁶ Module for working with structured text documents, *BeautifulSoup* (on-line)

¹⁷ Python module for NLP, *nltk* (on-line)

¹⁸ Machine learning module, *scikit-learn* (on-line)

example, specific terms that users can use in different social networks could be determined.

The knowledge of Python programming language, markup languages and statistical concepts, combined with information retrieval and three foreign languages, offer these students numerous opportunities for employment in the information technology field. This wide range of work positions includes: software testing, database maintenance, statistical tools usage, maintenance of indexes in information retrieval systems, creation of diverse textual material from the web, web applications development, big data analysis etc.

7 Conclusion

Even after a very brief overview of the computer science courses curricula given in this paper, it is obvious that LIS students acquire multidisciplinary knowledge and skills that can be applied in different contexts. The Python programming course is just one part of a modern program that LIS Department offers, along with various application opportunities. The task of the teachers is to encourage students to develop their interests in the direction that is attractive, but also prosperous. Thus, the contribution of LIS students in the development of resources and Web tools for Serbian language could be very significant.

References

- Abadi, Martín, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis et al.. “TensorFlow: A System for Large-Scale Machine Learning”. In *Operating Systems Design and Implementation* Vol. 16 (2016): 265–283. Accessed March 1, 2018, <https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf>
- Agarwal, Krishna K. and Achla Agarwal. “Python for CS1, CS2 and Beyond”. *Journal of Computing Sciences in Colleges* Vol. 20, no. 4 (2005): 262–270. Accessed March 1, 2018, <http://dl.acm.org/citation.cfm?id=1047846.1047887>
- Bird, Steven and Edward Loper. “NLTK: the Natural Language Toolkit”. In *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*. Association for Computational Linguistics, 2004, 31. Accessed March 1, 2018, <http://www.aclweb.org/anthology/P04-3031>

- Goldwasser, Michael H. and David Letscher. “Teaching an Object-oriented CS1 -: With Python”. In *Proceedings of the 13th Annual Conference on Innovation and Technology in Computer Science Education*. Conference on Innovation and Technology in Computer Science Education '08, New York, NY, USA: ACM, 2008, 42–46. Accessed March 1, 2018, <http://doi.acm.org/10.1145/1384271.1384285>
- Hadjerrouit, Said. “Java as First Programming Language: a Critical Evaluation”. *ACM Special Interest Group on Computer Science Education Bulletin* Vol. 30, no. 2 (1998): 43–47.
- Hosch, Frederick. “Java as a First Language: an Evaluation”. *Special Interest Group on Computer Science Education* Vol. 28, no. 3 (1996): 45–50.
- Jayal, Ambikesh, Stasha Lauria, Allan Tucker and Stephen Swift. “Python for Teaching Introductory Programming: A Quantitative Evaluation”. *Innovation in Teaching and Learning in Information and Computer Sciences* Vol. 10, no. 1 (2011): 86–90. Accessed March 1, 2018, <https://www.tandfonline.com/doi/abs/10.11120/ital.2011.10010086>
- King, K. N. “The Case for Java as a First Language”. In *Proceedings of the 35th Annual Southeast Regional Conference*, ACM, 1997, 124–131. Accessed March 1, 2018, <https://pdfs.semanticscholar.org/e8b0/e9cbbff598706136925cf8f25f66bbeed89f.pdf>
- Knuth, Donald Ervin. “Literate Programming”. *The Computer Journal* Vol. 27, no. 2 (1984): 97–111.
- Kovrlija, Darja and Valentina Tasić and Suzana Topalović. ““Priest Ćira and and priest Spria” – the united knowledge on the common student’s project”. *INFOtheca - Journal for Digital Humanities* Vol. 13, no. 2 (2012): 80–83. Accessed March 1, 2018, http://infoteka.bg.ac.rs/pdf/Eng/2012-2/INFOTHECA_XIII_2_December2012_80-83.pdf
- Krstev, Cvetana. “Information Science Curriculum at the Undergraduate Studies of Library and Information Science”. In *Skup bibliotekara balkanskih zemalja: Saradnja obrazovanje kvalitet*, Beograd, Srbija: Narodna biblioteka Srbije, 2002, 117–122. Приступљено 01.03.2018, <http://poincare.matf.bg.ac.rs/~cvetana/biblio/BalkanBiblio.pdf>
- Krstev, Cvetana and Aleksandra Trtovac. “Teaching Multimedia Documents to LIS Students”. *The Journal of Academic Librarianship* Vol. 40, no. 2 (2014): 152–162. Accessed March 1, 2018, <http://iranarze.ir/wp-content/uploads/2017/07/7262-English-IranArze.pdf>
- Lazić, Biljana and Jelica Poklopić, “Multimedia Document “Cult Radio Programs” - An Insight into the Status of the Archives of Radio Stations in Serbia”, *INFOtheca - Journal for Digital Humanities* Vol. 12, no. 2

- (2011): 48–52, Accessed March 1, 2018, http://infoteka.bg.ac.rs/pdf/Eng/2011-2/INF0theca-JournalforDigitalHumanities_XII_2_Decembar_48-52.pdf
- Obradović, Milena, Aleksandra Arsenijević and Mihailo Škorić. “Preparation of Multimedia Document “YU Rock Scene””. *INFOtheca - Journal for Digital Humanities* Vol. 16, no. 1-2 (2016): 111–127. Accessed March 1, 2018, https://infoteka.bg.ac.rs/ojs/index.php/Infoteka/article/view/2016.16.1_2.6_en
- Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion et al.. “Scikit-learn: Machine learning in Python”. *Journal of machine learning research* Vol. 12 (2011): 2825–2830. Accessed March 1, 2018, <http://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf>
- Radenski, Atanas. ““Python First”: A Lab-based Digital Introduction to Computer Science”. *Special Interest Group on Computer Science Education Bull* Vol. 38, no. 3 (2006): 197–201. Accessed March 1, 2018, <http://doi.acm.org/10.1145/1140123.1140177>
- Ranum, David, Bradley Miller, John Zelle and Mark Guzdial. “Successful Approaches to Teaching Introductory Computer Science Courses with Python”. *Special Interest Group on Computer Science Education Bull*. Vol. 38, no. 1 (2006): 396–397. Accessed March 1, 2018, <http://doi.acm.org/10.1145/1124706.1121465>
- Sanders, Ian D. and Sasha Langford. “Students’ Perceptions of Python As a First Programming Language at Wits”. *Special Interest Group on Computer Science Education Bull* Vol. 40, no. 3 (2008): 365–365. Accessed March 1, 2018, <http://doi.acm.org/10.1145/1597849.1384407>
- Stanković, Ranka, Branislava Šandrih, Olivera Kitanović, Ivan Obradović and Miloš Manić. “An E-Learning Approach to Social Sciences”. In *The 8th International Conference on eLearning (eLearning-2017)*, Belgrade, Serbia: Belgrade Metropolitan University, 2017, 26–29. Accessed March 1, 2018, <http://econference.metropolitan.ac.rs/files/Elearning-2017-final.pdf>
- Stošić, Anđela, Marina Milošević and Sandra Spasić. “Working on the Multimedia Document “Al’ se nekad dobro jelo””. *INFOtheca - Journal for Digital Humanities* Vol. 17, no. 2 (2018): 97–118. Accessed March 1, 2018, https://infoteka.bg.ac.rs/ojs/index.php/Infoteka/article/view/2017.17.2.5_en
- Vasljević, Nebojša. “Why Python is a Good Programming Language for Learning to Program”. *INFOtheca - Journal for Dig-*

- ital Humanities* Vol. 14, no. 1 (2013): 58–70. Приступљено 01.03.2018, http://infoteka.bg.ac.rs/pdf/Eng/2013-1/INFOtheca-JournalforDigitalHumanities_XIV_1_2014_58a-70a.pdf
- Vraneš, Aleksandra. “From the History of the Library and Information Science Department of the Faculty of Philology of the University of Belgrade”. *INFOtheca - Journal for Digital Humanities* Vol. 9, no. 1-2 (2008): 5–9. Accessed March 1, 2018, http://infoteka.bg.ac.rs/pdf/Eng/2008/INFOtheca-JournalforDigitalHumanities_IX_1-2_May2008_5a-9a.pdf
- Šandrih, Branislava. “Mogući doprinos L^AT_EX-a u obrazovnom procesu”. In *Slobodan softver u obrazovanju*, Sremski Karlovci, Srbija: Udruženje profesora informatike Srbije, 2016, 63–66. Accessed March 1, 2018, <https://www.slideshare.net/pdisrbije/slobodan-softver-2016-zbornik-radova-30012016>
- Утвић, Милош. “Регуларни изрази”. *Свет речи* Год. 7, бр. 15-16 (2003): 66–68.