

# The use of massive open online courses in teaching the fundamentals of programming to software engineers

Tetiana A. Vakaliuk<sup>1,2,3</sup>, Oleksii V. Chyzhmotria<sup>1</sup>, Olena H. Chyzhmotria<sup>1</sup>,  
Svitlana O. Didkivska<sup>4</sup> and Valerii V. Kontsedailo<sup>5</sup>

<sup>1</sup>Zhytomyr Polytechnic State University, 103 Chudnivska Str., Zhytomyr, 10005, Ukraine

<sup>2</sup>Institute for Digitalisation of Education of the National Academy of Educational Sciences of Ukraine,  
9 M. Berlynskoho Str., Kyiv, 04060, Ukraine

<sup>3</sup>Kryvyi Rih State Pedagogical University, 54 Gagarin Ave., Kryvyi Rih, 50086, Ukraine

<sup>4</sup>Cracow University of Economics, 27 Rakowicka, Kraków, 31-510, Poland

<sup>5</sup>Inner Circle, Nieuwendijk 40, 1012 MB Amsterdam, Netherlands

**Abstract.** The article describes the possibilities of using massive open online courses (MOOC) in teaching the discipline “Fundamentals of Programming” to students majoring in software engineering. The content of studying the discipline “Fundamentals of Programming” and the number of hours allotted for its study in institutions of higher education in Ukraine is analyzed. When studying the discipline “Fundamentals of Programming” at the Zhytomyr Polytechnic State University, the study of the corresponding course on the platform of massive open online courses Sololearn is submitted for independent work. A description of the possibilities of using this MOOC in teaching the discipline “Fundamentals of Programming” to future specialists in software engineering is presented. In addition, using a pedagogical experiment, the effectiveness of using the Sololearn in teaching the discipline “Fundamentals of Programming” to future software engineering specialists was tested. The results of the pedagogical experiment showed the effectiveness of using the MOOC Sololearn in teaching the “Fundamentals of Programming” to future software engineering specialists.

**Keywords:** massive open online courses, fundamentals of programming, software engineering specialists

## 1. Introduction

In the process of professional training of students of specialty 121 “Software Engineering” in Ukraine, some professional disciplines are required to study, in particular, “Fundamentals of Programming”, “Object-Oriented Programming”, “Web Technologies”, etc.

✉ [tetianavakaliuk@gmail.com](mailto:tetianavakaliuk@gmail.com) (T. A. Vakaliuk); [chov@ztu.edu.ua](mailto:chov@ztu.edu.ua) (O. V. Chyzhmotria); [ch-o-g@ztu.edu.ua](mailto:ch-o-g@ztu.edu.ua) (O. H. Chyzhmotria); [d2023@student.uek.krakow.pl](mailto:d2023@student.uek.krakow.pl) (S. O. Didkivska); [valerakontsedailo@gmail.com](mailto:valerakontsedailo@gmail.com) (V. V. Kontsedailo)

🌐 <https://sites.google.com/view/neota> (T. A. Vakaliuk); <https://scholar.google.com/citations?user=O9Ih4tEAAAAJ> (O. V. Chyzhmotria); <https://scholar.google.com/citations?user=0DYB2pcAAAAJ> (O. H. Chyzhmotria); <https://www.linkedin.com/in/svitlana-didkivska-b9b61616a> (S. O. Didkivska); <https://www.linkedin.com/in/kontsedaylo/> (V. V. Kontsedailo)

🆔 0000-0001-6825-4697 (T. A. Vakaliuk); 0000-0002-5515-6550 (O. V. Chyzhmotria); 0000-0001-8597-1292 (O. H. Chyzhmotria); 0000-0002-4004-6631 (S. O. Didkivska); 0000-0002-6463-370X (V. V. Kontsedailo)



© Copyright for this paper by its authors, published by Academy of Cognitive and Natural Sciences (ACNS). This is an Open Access article distributed under the terms of the Creative Commons License Attribution 4.0 International (CC BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

All disciplines are studied in the manner prescribed by the professional education program and provide a certain basic level of knowledge for the preliminary subject study.

The analysis of the curricula of higher education in Ukraine (the National University of Life and Environment Sciences of Ukraine, Zhytomyr Polytechnic State University, National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”, Odesa Polytechnic National University) showed that the study of the discipline “Fundamentals of Programming” is given from 150 to 330 hours (6 to 11 ECTS credits). In some universities, this discipline is divided into two, namely “Programming” and “Fundamentals of Software Engineering”.

A sufficiently large number of hours predetermines the need to submit part of the material for independent study (usually from 30% to 60% of the material is submitted for independent study, and universities themselves operate on this). A variety of additional learning tools becomes useful here for both teachers and students. One such tool is the Massive Open Online Course (MOOC) platform. There are two types of massive open online courses: connective Massive Open Online Courses (cMOOCs) and extended Massive Open Online Courses (xMOOCs). These MOOCs differ in that the first (cMOOCs) share digital educational environments and are also combined in various ways; the second (xMOOCs) are based on content provided by various educational institutions [8].

In the professional training of software engineers in the context of a pandemic and the Russian-Ukrainian war, a combination of traditional, distance, and blended learning techniques, as well as the use of various additional tools, including massive open online courses, are becoming increasingly important, which contributes to student motivation and interest.

Therefore, our paper is structured as follows. Section 2 contains a theoretical review of the scientific literature on the research issues. Section 3 describes the main methods for this study. Section 4 provides a description of the subject of the discipline “Fundamentals of programming”, which is studied by 1st year students majoring in software engineering, an overview of the independent work of students within the scope of studying this discipline, in particular, the use of the MOOC Sololearn within the scope of independent work of students with a detailed description of its possibilities, the advantages of using this tool are given, as well as an experimental verification of the feasibility of using the MOOC Sololearn when learning the basics of programming. Section 5 is the conclusion of this paper.

## 2. Theoretical background

MOOC has already gained popularity in the first quarter of the 21st century, so there are a lot of articles related to that topic, even more, there are books, that are devoted to the disclosure of this topic from different angles. For example, we can see a demonstration of the historical perspective on various stages of development of these courses and how their value changed, depending on the stage [17], advice on optimizing the process of their implementation and use according to the type and stage of development of the information infrastructure of the educational institution advice on optimizing the process of their implementation and use respectively the type and stage of development of the information infrastructure of the educational institution [7] or just perspective of its evolution from a higher education point of view [1]. Ethical issues for the study and the relations between science and society are described by Petousi and Sifaki [11].

However, speaking about the latest research results, which undoubtedly appeared more due to the growing popularity of distance learning as a result of the pandemic, it is worth paying attention to Scopus as one of the most reliable sources of verified information. There are currently 48 articles available in this source that focus on this topic part of them are concentrating on the adaptation of the course to the general type and specifics of the country's study conditions or just on solving some specific problems universities are forced to face too, like overcrowded universities on the impossibility of organizing real-time face-to-face lessons [9]. However, one of the most important and interesting aspects of the Massive Open Online Courses study is its use in teaching students a specific subject, for example, mathematics [14], international relations [3], geography [13], physics [2] or languages [4]. Several authors have studied the satisfaction of first-year students with a distance learning program in mathematics [6]. Still, we have to remember about digital barriers [5] that students might have while studying in that way. Those barriers could affect data results and lead to a distortion of information about the prospects for the development of the MOOC industry in education since every year students are better versed in digital technologies. The best way out of the situation is to test such courses on subjects that are studied by students of specialties related to information technology. A good example of such subjects could be computer networks [16] or software engineering.

There are many such MOOC platforms in the era of digitalization [10]. A review of various platforms of the MOOC, including Sololearn [12], showed that most of them contain theoretical material (for familiarization with the primary material of each topic), test tasks (for testing knowledge), and the possibility of communication among other participants (for communication and mutual consultation). In the case of learning programming languages (Sololearn also), MOOC also contains a built-in compiler (for training acquired skills from each programming language).

However, the problem of using platforms for massive open online courses in teaching certain professional disciplines in the process of preparing bachelor of software engineering requires a separate study.

Therefore, the purpose of the article is to describe the possibilities of using massive open online courses in teaching the basics of programming to software engineers, as well as to test the effectiveness of such use.

### 3. Research methods

To achieve the purpose, as well as to implement the objectives of this study, a set of methods was used:

- analysis, systematization, generalization of psychological and pedagogical, special and methodological literature on the problems of using MOOC in teaching software engineers in order to highlight relevant areas of research;
- observation – to identify the features of the use of MOOC in the educational process of future software engineers;
- testing – to determine the level of learning achievements in the basics of programming;

- pedagogical experiment – to experimentally test the pedagogical feasibility (efficiency) of introducing MOOC Sololearn into the study of the discipline “Fundamentals of Programming”;
- mathematical and statistical methods, with the help of which quantitative dependence between the indicators obtained as a result of the ascertaining and control stages of the study were determined, and a qualitative analysis was carried out.

To test the statistical equivalence of the experimental group (EG) and control group (CG), Pearson’s  $\chi^2$  test was used. For this criterion, some restrictions are important: sample size  $n \geq 30$ ; the frequency for each cell of the table should not be less than 5 (that is why this criterion could not be used when comparing the distributions of grades for the school course of informatics of students in the CG and the EG); the selected digits exhaust the entire distribution, that is, they cover the entire range of feature variability. All these conditions are satisfied for the obtained samples. Therefore, we are entitled to apply this criterion.

## 4. Results

**The purpose of studying** the discipline “Fundamentals of Programming” is to form students’ theoretical basis necessary for further work, obtaining theoretical knowledge and practical skills in the algorithmization of computational processes, the basic principles of software development in the C programming language.

**The objectives of** the academic discipline “Fundamentals of Programming” are:

- providing students with the necessary knowledge on the theory and practice of using algorithmic programming languages;
- to form an idea among students about the main stages of solving a computer problem;
- to gain a sequence of actions, skills, and abilities to work with modern software, program debugging;
- providing students with basic knowledge and skills with basic algorithmic structures;
- mastering basic knowledge and skills in working with arrays;
- mastering different methods of sorting arrays;
- providing students with basic knowledge and skills for working with functions, as well as the ability to implement recursive functions;
- mastering students ability to work with structures, string values, and pointers;
- mastering students basic skills of working with repositories.

Let us give an indicative topic for studying this discipline (as an example, the program given, according to which they study at the Zhytomyr Polytechnic State University).

Content module 1. **Introduction to programming. Basic construction methods.**

Topic 1. **Introduction to programming.** Basic concepts. The concept of a method. Ways of representing algorithms. The alphabet of the C programming language. Directives. Connecting libraries. Data types. Constants and variables. Declaring variables. Operations

of the C programming language. Data input-output operators. Formatting specifiers. Arithmetic operations. Features of working with them. Increment, decrement. Math functions. Block diagram. Basic elements of flowcharts. Basic algorithmic constructions.

Topic 2. **Branching.** Comparison operations. Logical operations. Boolean expressions. Simple and compound logical expressions. The if branching statement. Ternary operation. Switch statement.

Topic 3. **Introduction to the Git version control system.** The concept of a repository. Version control systems. Introduction to the Git version control system.

Topic 4. **Cycles.** The concept of a cycle. Cycle with while condition. Cycle with for parameter. Break and continue. Cycle with postcondition do...while. The comma operation. Nested cycles.

#### Content module 2. **Arrays.**

Topic 5. **Setting up the program. Testing.** Basic concepts. Setting. Error types. Test types.

Topic 6. **Arrays.** Array concept. One-dimensional arrays. Declaring and calling in one-dimensional arrays. Generation of pseudo-random numbers. Operations with one-dimensional arrays.

Topic 7. **Sorting arrays.** Sort by the exchange. Sorting by the selection method. Insertion sort. Shell sorting. Pyramid sort.

Topic 8. **Multidimensional arrays.** Declaring and accessing multidimensional arrays. Two-dimensional arrays. Operations with matrices.

#### Content module 3. **Functions, pointers, structures.**

Topic 9. **Functions.** The concept of a function. Function prototype. Passing parameters. Functions with a changing number of parameters. Passing an array to a function. Recursive functions.

Topic 10. **Pointers.** Local and global variables. Variable address. The concept of a pointer. Declaring a variable of pointer type. Basic operations on pointers. Features of the scanf function. Pointers and arrays.

Topic 11. **Character variables.** Working with string and character variables. String functions.

Topic 12. **Structures.** The concept of structure. Arrays of structures. Programming with structures.

Studying the discipline “Fundamentals of Programming” at Zhytomyr Polytechnic State University, students are recommended to study the materials of massive open online courses as part of their independent work. Students can take the course offered on the Massive Open Online Course (MOOC) platform: Sololearn C [12]. This course provides an opportunity to study individual theoretical blocks, and perform individual test tasks and certain final works.

In previous works, we studied the selection of MOOC for the training of future software engineers, and in the process of such selection, it was found that there are only special cases for learning the C language since such platforms are focused on more difficult programming

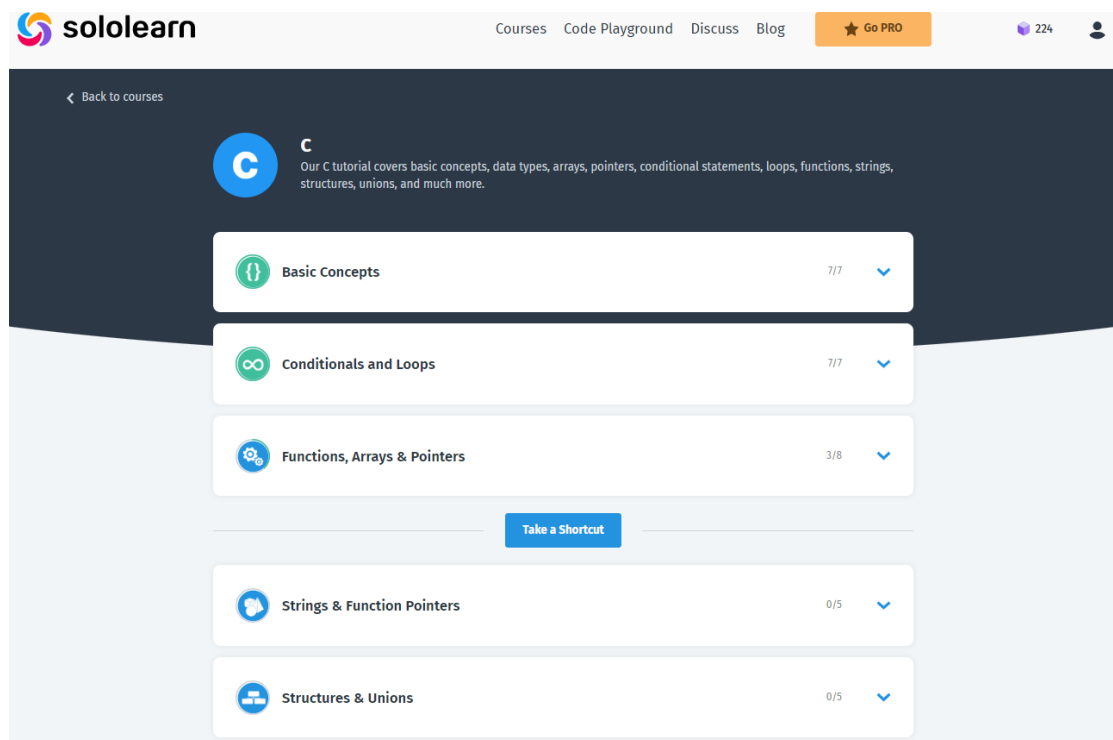
languages to learn [15]. Therefore, the authors chose this MOOC on the Sololearn platform to study the basics of programming in the C language.

Let us consider the possibility of using MOOC Sololearn in the course “Fundamentals of Programming” as a part of additional independent work. The first thing to point out is that this course is in English, and since this platform does not provide any actions of teachers or instructors, it is not possible to localize in another language.

This course involves studying the material and improving skills in the basics of programming in the following sections: basic concepts of the C programming language; conditional expressions and loops; functions, arrays, pointers; strings and function pointers; structures; memory management; file and error handling; preprocessor.

These topics fully cover the topics for study in the main course “Fundamentals of Programming” and are their logical addition. Another feature of this MOOC platform is that the courses are freely available and anyone can take them. Yes, not all features of the platform are freely available, but if you wish, you can purchase the full version for an in-depth study of the material. Another feature of this platform is that each course is provided with a separate address where you can enter the course without wasting time searching for the course.

When you log into your account at the address of a specific course [12], the user opens his profile of passing (studying) the course (figure 1).



**Figure 1:** Personal account for studying the course.

Each section (study topic) in this MOOC includes theoretical information (figure 2), containing material for repeating what was learned in the classroom.

**sololearn** Courses Code Playground Discuss Blog **Go PRO** 224

Two-Dimensional Arrays 90 Comments

A **two-dimensional array** is an array of arrays and can be thought of as a table. You can also think of a two-dimensional array as a grid for representing a chess board, city blocks, and much more.

A two-dimensional array declaration indicates the number of number rows and the number of columns.

**For example:**

```
int a[2][3]; /* A 2 x 3 array */
```

Nested curly braces are used to initialize elements row by row, as in the following statement:

```
int a[2][3] = {
    {3, 2, 6},
    {4, 5, 20}
};
```

The same statement can also take the form:

```
int a[2][3] = { {3, 2, 6}, {4, 5, 20} };
```

The former statement offers more clarity with visualizing the structure of the array.

An array can have more than two dimensions. For example, `a[5][3][4]` is an array that has 5 elements that each store 3 elements that each store 4 elements.

**Figure 2:** Theoretical material in the MOOC Sololearn.

This MOOC allows you to comment on the activities for each topic. Such a section is located in the upper right corner, by entering which a registered user at the stage of viewing theoretical material can communicate with other registered users through comments (figure 3).

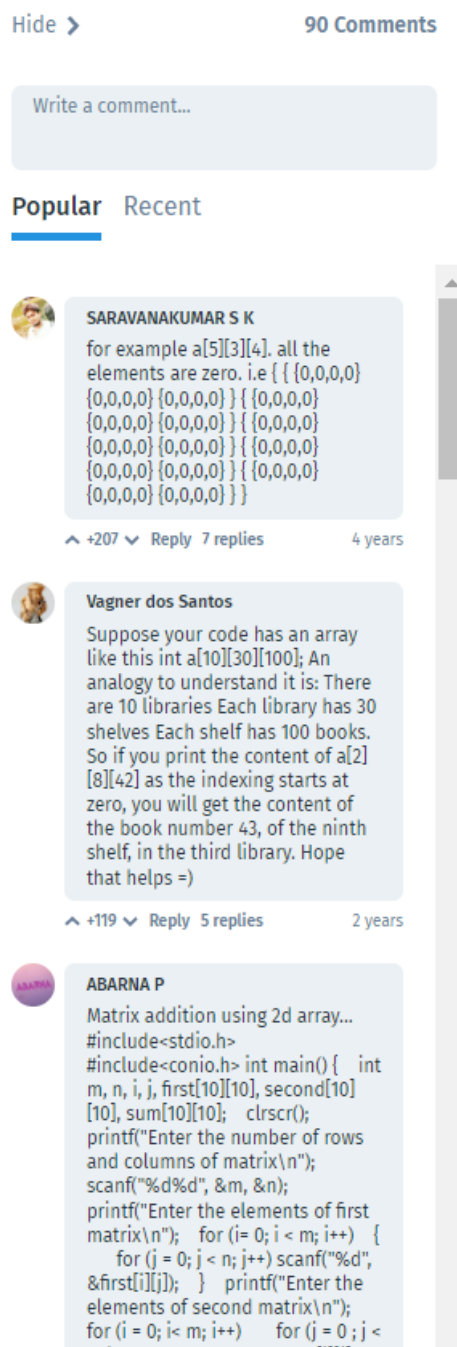
However, it should be noted that not all sections of the MOOC data, such an opportunity opens immediately. In addition to theoretical material, users are allowed to view examples of the program execution and try to work with them on their own, making certain changes (figure 4 and figure 5).

This platform MOOC also provides a task for testing knowledge of various forms, for example, choosing 1 answer from several proposed ones (figure 6) and entering your answer (figure 7). At the same time, it should be noted that the section of communication through comments in such tasks becomes available only after checking the answer. For each topic, up to 10 tasks of this type are given to consolidating knowledge.

After passing the module, the user is presented with a final test (figure 8), which consists similarly of several questions (from 5 to 10), and according to the result, it is set whether the module was enrolled to him.

It should also be noted here that the next module will not be opened to the user until he completely completes the previous one (figure 1).

If we consider each topic offered in this MOOC, then in section 1 “Basic concepts” there is a repetition of the base of the C programming language, namely: data types, arithmetic operators, comments in the code, working with the console (input and output). In section 2 “Conditional statements and loops”, conditional statements were considered, such as: if, else,



**Figure 3:** The ability to communicate through comments in the MOOC Sololearn.

switch, and three types of loops: for, while, do... while. The section also included theory on logical operators, which are &&, which means “i”, and ||, which means “or”.



## Accessing Two-Dimensional Arrays

To access an element of a two-dimensional array, both the row index and column index are required.

For example, the following statements display the value of an element and then assign a new value:

```
int a[2][3] = {
    {3, 2, 6},
    {4, 5, 20}
};
printf("Element 3 in row 2 is %d\n", a[1][2]); /* 20 */
a[1][2] = 25;
printf("Element 3 in row 2 is %d\n", a[1][2]); /* 25
*/
```

Try it Yourself

Just as a **for** loop is used to iterate through a one-dimensional array, nested **for** loops are used to traverse a two-dimensional array:

```
int a[2][3] = {
    {3, 2, 6},
    {4, 5, 20}
};
int k, j;
/* display array contents */
for (k = 0; k < 2; k++) {
    for (j = 0; j < 3; j++) {
        printf("%d", a[k][j]);
    }
    printf("\n");
}
```

Try it Yourself

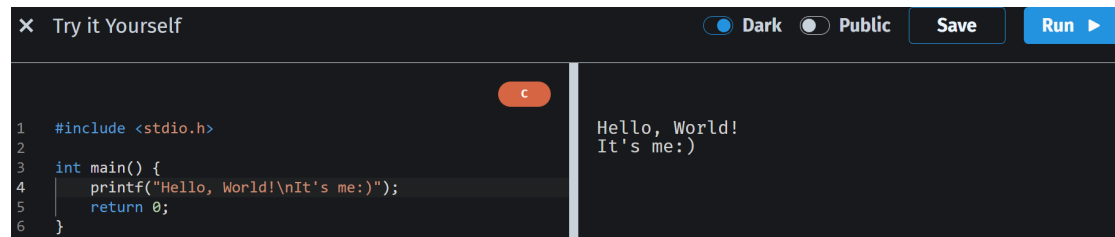
Back

Continue

**Figure 4:** Ability to work with program code in Sololearn.

Section 3 “Functions, Arrays, and Pointers” provides material that is completely repetitive to the classroom material. As a result, in this section, there is only a repetition of the studied material, as well as the consolidation of the necessary skills for working with arrays, functions, and pointers.

Section 4 “Strings and Function Pointers” has a new topic that doesn’t overlap with the classroom material. This is the “Pointers to Functions” topic, which explains in detail how



```

1 #include <stdio.h>
2
3 int main() {
4     printf("Hello, World!\nIt's me:);
5     return 0;
6 }

```

Output: Hello, World!  
It's me:)

**Figure 5:** Working with program code in Sololearn.

### Data Types

Which of the following are valid data types in C?

int, double, char, boolean

int, bool, string

int, float, double, char

int, float, string, char

Well done!

**Figure 6:** Choice of 1 correct answer.

to declare and access functions using pointers. The next section is “Structures and Unions”. It should be noted here that students were offered a new topic for study - this is the topic “Union”, in which students can learn about unions, which allows storing different types of data in the same place in memory, accessed through the dot operator located between the name of a variable and the name of a particular union member.

In Section 6 “Memory Management”, the following functions were considered for memory allocation on dynamic array memory (malloc, calloc), array expansion (realloc), and freeing (free). This material is not new to the student, since this topic completely overlaps with the topic for studying in the classroom, so students in this section only consolidated their knowledge and skills.

Section 7 “File and error handling” is the largest section in terms of the amount of new material that is submitted for independent processing. By studying this section, students can learn how files are handled in the C language, which is used for this function, what binary files are, and how to handle errors.

The free Function	Recursive Functions
Fill in the blanks to allocate memory and then free the memory allocated by malloc().	Fill in the blanks to define a recursive function for calculating the factorial of n:
<pre>int * ptr = malloc(10); free(ptr);</pre>	<pre>int fact(int n) {     if (n == 1) return 1;     return n * fact(n - 1); }</pre>

Figure 7: Entering your answer.

Which choice indicates a single-line comment?	Fill in the blanks to print the numbers 0 through 5 using the do-while loop:
<input type="text" value="##single line comment"/> <input type="text" value="**single line comment"/> <input checked="" type="text" value="//single line comment"/>	<pre>int c = 0; do {     printf("%d", c);     c++; } while (c &lt;= 5);</pre>

Figure 8: Module test.

Section 8 “Preprocessor” contained some of the material that students studied in the classroom and some of the new ones. The topics covered in this section are preprocessor directives, conditional compilation directives, and preprocessor statements.

Note that the advantage of such MOOC is that all the material is structured, and for each topic, there is theoretical material, attempts to work with the program code and knowledge testing.

The use of this MOOC for self-processing by students within the framework of the course “Fundamentals of Programming” contributes to the consolidation of the mastered theoretical material, since the course contains a repetition of the theory presented to students in lectures on this discipline, as well as the development and improvement of skills in programming in the C language.

During the course, students can view the progress bar (figure 9), and after successful completion of the course, students receive a certificate (figure 10).

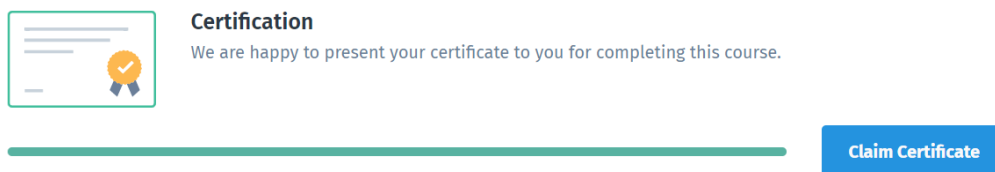


Figure 9: Course completion scale.



Figure 10: Certificate.

Note that the MOOC Sololearn is adaptive for various systems and, accordingly, devices, so the user has the opportunity to use it from any device.

The use of MOOCs in the educational process of higher education contributes to the assimilation of the material through one's own practical experience, mostly independently. After all, students can get acquainted with additional theoretical information, view various examples with solutions in the C language, and at the same time they can also try to change a fragment of the program code using the knowledge gained; perform a separate task by writing code fragments in the answer, and, of course, take a test to check their knowledge.

The use of such a MOOC provides 100% coverage of the topics provided by the training program for studying the discipline "Fundamentals of Programming". Note that MOOC is not the main one in the study of the course "Fundamentals of Programming". It is only an auxiliary tool for consolidating knowledge and skills during lectures and practical exercises.

To test the effectiveness of the implementation of the MOOC Sololearn in the study of the discipline "Fundamentals of Programming", a pedagogical experiment was conducted at

the Zhytomyr Polytechnic State University, which meant comparing the learning outcomes according to the traditional method of teaching the basics of programming and the author’s using the MOOC Sololearn.

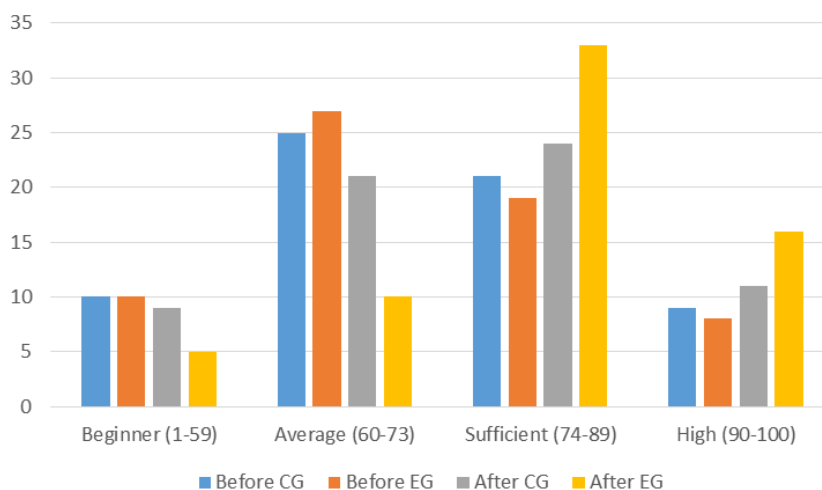
For this, the applicants for the education of 1 year of study were divided into the EG and the CG (experimental and control groups, respectively). In the CG, training was carried out according to the traditional method, and in the EG – using the MOOC Sololearn.

Statistical data of the EG and CG before and after the experiment are presented in table 1 and figure 11.

**Table 1**

Comparative distribution of students from the CG and the EG by the level of educational achievements in the fundamentals of programming at the beginning and end of the pedagogical experiment.

Level of academic achievement	Before		After	
	CG	EG	CG	EG
Beginner (1-59)	10	10	9	5
Average (60-73)	25	27	21	10
Sufficient (74-89)	21	19	24	33
High (90-100)	9	8	11	16
Total	65	64	65	64



**Figure 11:** Comparative distribution of students in CG and EG according to the level of academic achievements in the fundamentals of programming at the beginning and end of the pedagogical experiment.

To test the statistical equivalence (to the experiment) and statistical non-equivalence (after the experiment), the Pearson test was applied, the result of which was found:

- at the beginning of the pedagogical experiment, it was found that  $\chi_{emp}^2 = 0.36$ ,  $\chi_{emp}^2 < \chi_{0.05}^2$ , therefore, the samples do not have statistically significant differences, and this means that the composition of students in the EG and CG is approximately equivalent;

- at the end of the pedagogical experiment, it was found that  $\chi_{emp}^2 = 8.48$ ,  $\chi_{emp}^2 > \chi_{0.01}^2$  and this indicates that the samples have statistically significant differences at this stage.

Since the statistical data testify to the positive dynamics of a sufficient and high level of educational achievements in the experimental group at the end of the pedagogical experiment, which is confirmed experimentally, we can conclude that the experimental methodological system of teaching “Fundamentals of Programming” using the MOOC Sololearn is more effective than the traditional one.

## 5. Conclusions

When teaching the basics of programming to future software engineers, changes should be made to the curriculum of the discipline with the addition of MOOC for independent work. As the conducted pedagogical experiment showed, the use of MOOC Sololearn in teaching the basics of programming to future bachelors in software engineering helps to increase the level of students' educational achievements. In addition, as the students themselves point out, the use of various tools in the educational process contributes to their self-development, motivation, and interest, and helps to consolidate the acquired knowledge and skills. The prospects for further research include the study of the possibilities of using other ICT tools in the independent work of future software engineers.

## References

- [1] Bennett, R. and Kent, M., eds, 2017. *Massive open online courses and higher education: What went right, what went wrong and where to next?* Taylor & Francis.
- [2] Huangqing, L. and Yanping, X., 2021. Online teaching research on college physics by using different teaching tools. *Physics Education*, 56(3), p.035007. Available from: <https://doi.org/10.1088/1361-6552/abdee1>.
- [3] Kaempf, S. and Finn, C., 2019. Teaching International Relations through the Format of a Massive Open Online Course (MOOC). *International Studies Perspectives*, 22(1), pp.1–24. Available from: <https://doi.org/10.1093/isp/ekz025>.
- [4] Lebedeva, M., 2021. Instructional Design of Skill-Balanced LMOOC: a Case of the Russian Language MOOC for Beginners. *Journal of Universal Computer Science*, 27(5), pp.485–497. Available from: <https://doi.org/10.3897/jucs.67136>.
- [5] Ling, V., Sotnikova, L., Rodionova, I., Vasilets, I., Zavjalova, O., Fedorovskaya, V. and Datkova, E., 2020. Online educational resources for students and digital barrier. *TEM Journal*, 9(1), pp.373–379. Available from: <https://doi.org/10.18421/TEM91-51>.
- [6] Matzakos, N.M. and Kalogiannakis, M., 2018. An analysis of first year engineering students' satisfaction with a support distance learning program in mathematics. *Education and Information Technologies*, 23(2), pp.869–891. Available from: <https://doi.org/10.1007/s10639-017-9641-y>.
- [7] Mesquita, A. and Peres, P., eds, 2015. *Furthering higher education possibilities through massive open online courses*. IGI Global. Available from: <https://doi.org/10.4018/978-1-4666-8279-5>.

- [8] MOOCS, cMOOCS and xMOOCS: Definition and explanation, 2021. Available from: <https://oxford-review.com/oxford-review-encyclopaedia-terms/moocs-cmoocs-and-xmoocs-definition-and-explanation/>.
- [9] Naji, K., Ibriz, A. and Mourdi, Y., 2020. Adoption of MOOCs by Emerging Countries Seeking Solutions to University Overcrowding: Literature Review and Feedback from the First Scientific MOOC Held by Sidi Mohammed Ben Abdullah University – Fez, Morocco. *International Journal of Emerging Technologies in Learning (ijET)*, 15(22), p.78–103. Available from: <https://doi.org/10.3991/ijet.v15i22.16945>.
- [10] Panchenko, L., 2021. Digital storytelling in adult education: barriers and ways to overcome them. *Educational Technology Quarterly*, 2021(4), p.673–688. Available from: <https://doi.org/10.55056/etq.41>.
- [11] Petousi, V. and Sifaki, E., 2020. Contextualizing harm in the framework of research misconduct. findings from discourse analysis of scientific publications. *International Journal of Sustainable Development*, 23(3-4), pp.149–174. Available from: <https://doi.org/10.1504/IJSD.2020.10037655>.
- [12] Sololearn, 2022. C. Available from: <http://web.archive.org/web/20220924155455/https://www.sololearn.com/learning/1089>.
- [13] Syvyi, M.J., Mazbayev, O.B., Varakuta, O.M., Panteleeva, N.B. and Bondarenko, O.V., 2020. Distance learning as innovation technology of school geographical education. In: O.Y. Burov and A.E. Kiv, eds. *Proceedings of the 3rd International Workshop on Augmented Reality in Education, Kryvyi Rih, Ukraine, May 13, 2020*. CEUR-WS.org, *CEUR Workshop Proceedings*, vol. 2731, pp.369–382. Available from: <http://ceur-ws.org/Vol-2731/paper22.pdf>.
- [14] Vagaeva, O.A., Galimullina, N.M., Liksina, E.V., Efremkina, I.N. and Lomakin, D.E., 2021. Role of MOOCs in teaching Mathematics to students majoring in Engineering. *Journal of Physics: Conference Series*, 1889(2), p.022043. Available from: <https://doi.org/10.1088/1742-6596/1889/2/022043>.
- [15] Vakaliuk, T., Spirin, O. and Kontsedailo, V., 2021. Criteria for selecting open web-oriented technologies for teaching the basics of programming to future software engineers. *Educational Technology Quarterly*, 2021(1), p.73–86. Available from: <https://doi.org/10.55056/etq.16>.
- [16] Vakaliuk, T.A., Yefimenko, A., Bolotina, V., Bailiuk, Y., Pokotylo, O. and Didkivska, S., 2020. Using Massive Open Online Courses in Teaching the Subject "Computer Networks" to the Future IT Specialists. In: O. Sokolov, G. Zholtkevych, V. Yakovyna, Y. Tarasich, V. Kharchenko, V. Kobets, O. Burov, S. Semerikov and H. Kravtsov, eds. *Proceedings of the 16th International Conference on ICT in Education, Research and Industrial Applications. Integration, Harmonization and Knowledge Transfer. Volume II: Workshops, Kharkiv, Ukraine, October 06-10, 2020*. CEUR-WS.org, *CEUR Workshop Proceedings*, vol. 2732, pp.665–676. Available from: <http://ceur-ws.org/Vol-2732/20200665.pdf>.
- [17] Waks, L.J., 2016. *The Evolution and Evaluation of Massive Open Online Courses: MOOCs in Motion*, The Cultural and Social Foundations of Education. New York: Palgrave Pivot. Available from: <https://doi.org/10.1057/978-1-349-85204-8>.