

OLC, On-Line Compiler to Teach Programming Languages

Cayetano Guerra Artal, Maria Dolores Afonso Suarez
Idafen Santana Perez, Ruben Quesada Lopez

Abstract: The advance of Internet towards Web 2.0 conveys the potential it has in a wide range of scopes. The ongoing progress of the Web technology and its availability in teaching and learning, as well as a students' profile increasingly more used to managing an important amount of digital information, offers lecturers the opportunity and challenge of putting at students' disposal didactic tools making use of the Internet. Programming is one of the essential areas taught in university studies of Computer Science and other engineering degrees. At present, it is a knowledge acquired through tutorial classes and the practice with different tools for programming. This paper shows the acquired experience in the development and use of a simple compiler accessible through a Web page. In addition it presents a teaching proposal for its use in subjects that include programming languages lessons. OLC - On-Line Compiler - is an application which greatly lightens the student's workload at the initial stage of programming. During this initial period they will neither have to deal with the complexities of the installation and the configuration of these types of tools, nor with the understanding of multiple options which they present. Therefore students can concentrate on the comprehension of the programming structures and the programming language to be studied.

Keywords: Compiler, e-learning, interactivity, virtual laboratory, web 2.0, video tutorials.

1 Introduction

New Information and Communication Technologies (ICT's) still imply changes in a wide range of society scopes. A society in which the information, its use and distribution in digital format are part of a number of tasks carried out every day. New generations understand their environment in a natural way as well as the use of these technologies. This is reflected in university students who are increasingly getting used to managing this kind of information [1]. From the beginning Internet has meant a milestone in communication at a world wide level [2]. It provides Net users with an hypertext document system with a rich content and hyperlinks that allow to reference another resource: text document, image, animation, audio or video. In its beginning, information was contained in HTML pages that were not updated very frequently. From here its evolution was directed to dynamic HTML pages that make use of CMS with data bases. Next, it was directed to Web 2.0, a concept that has turned into reality in Internet, through which contents and the way of interaction with these contents in Internet have changed. Users, from sharing these contents, have turned their role into the edition and classification of them.

2 Web 2.0

This term, Web 2.0 [3], coined for the first time in 2004, makes reference to a new vision of the Web which has evolved towards users' communities, interaction and social networks. Amongst the technologies that bring the Web 2.0 into reality we find: the ones that use the Web platform to host desktop applications; the ones that separate content and design using style sheets; the ones that support RSS or the ones that allow the management of users and communities. In the same way, there exists a set of Web services that can be considered indicators of the advance towards Web 2.0 [4]. These services provide

communication mechanisms that make it possible to present information to users actively. Amongst these services we can find advertising services, shared photo-database, community-generated tag database, delivery of contents, encyclopaedias or personal web pages. Web navigators have experienced an important progress with regard to their function. They have been evolving from their beginnings, in which their function consisted in showing the contents of static web pages and their hyperlinks. Later on they went further, interpreting the interfaces that show one or various data bases' content. At present they support newest technologies that allow not only the interaction with dynamic web pages but also a wide range of web services. Amongst them social networks, that promotes information interchange between users. Internet has undergone a change which has made it become a relevant instrument for teaching and knowledge spreading, no matter the subject area. The presence of e-learning in Internet has allowed to increase education quality and its disposition. Its use to create teaching applications is fully interesting, being able to use, among other features, interactivity.

3 OLC: Learning Programming

Nowadays, programming is one of the essential areas taught in university studies of Computer Science and other engineering degrees, as well as in diplomas of Computer Science. At present, it is a knowledge acquired through theoretical classes and the practice with different tools for programming such as editors, compilers, linkers, debuggers or interpreters. In addition to this we find teacher tutorial classes. Being included in all programming subjects, compilers generate the executable programs made of the instructions written by the programmers. Compilation is a process carried out by means of the succession of a set of operations through which, using the instructions written in a programming language, we obtain the code written in another language that is understood by the computer. This is how we obtain executable programs. The Web 2.0 is at its very peak, this fact and the change in the web navigators' function allow to make use of them as production tools such as text editors, spreadsheets or, as we expound in the present work, a compiler. In this way we can move a multilanguage IDE from a computer to a Web page. OLC -On Line Compiler- is an application which greatly lightens the student's workload at the initial stage of programming. Its use is especially intended for subjects where the students must learn how to make programs. During this initial period they will neither have to deal with the complexities of the installation and the configuration of these types of tools, nor with the understanding of multiple options which they present. Therefore students can concentrate on the comprehension of programming structures and the programming language to be studied. The use made of the compiler in this initial stage is restricted to create programs from a set of instructions free of errors. The student will have written these instructions and the compiler will have helped to correct them. This way in later stages of the learning process and once the students have acquired the basic knowledge of the programming structures and the programming language, they can start using compilers with more complexities in the installation and configuration tasks as well as with a higher number of options. The knowledge acquired by the students in this first stage of the learning process helps them to get a better comprehension of the use and handling of the compilers.

3.1 Functionality

OLC interface is simple and intuitive. An interface defines the communication boundary between two entities, in this case between the students and the application. After the student identification, s/he must choose the programming language with which s/he is going to work. This can be performed from the selection language menu. Figure 1.

Once the programming language is chosen, the screen on Figure 2 will be displayed. The tabs menu located at the top of the screen, offers users the navigation through the editor, the files, the tutorials and other sections.



Figure 1: Welcome screen. Where the student chooses the programming language to work with.



Figure 2: OLC Work screen. With the tabs menu, the tools bar, and two text areas: one for the code and the other one for the results.

In the section "My files" a list with the user's files located on the server for that language is shown. These files are displayed by the last modified date. The students can manage their files using the available options. They can also load files in the server and work with them. Under the tabs menu, the tools bar can be found, whose options make it possible to work with different files and the code. Below the screen is divided into two text areas:

- One for the code, at the top, where students will write the code of their programs.
- One for the results, at the bottom, only for reading, that will give information about the different actions performed in the files.

Local files are shown as a list with a colour code used to identify them. Figure 3.



Figure 3: List of local files, with a colour code used to identify them.

Apart from the local files the application has a set of tutorials in video. By using them the information transmission can be carried out in a more efficient and interesting way for students. Through these videos Figure 4, a learning process can be established. These video tutorials have got a clear purpose: to assist users in learning how to manage the compiler and to follow the practical classes. These video tutorials are delivered through files embedded in the web page of OLC. This kind of material has got an increasingly didactic interest, since students will have the possibility of viewing again those chapters of the video where some doubts may have arisen. They could do this as many times as necessary [5] and from any device [6] with an Internet connection.

In the server we can find also a set of local files with examples. The teachers after the lessons can propose practical exercises using the different examples that can be found in the folders on-line. During the teaching period they can check the results of students' work, directing them in their learning process.

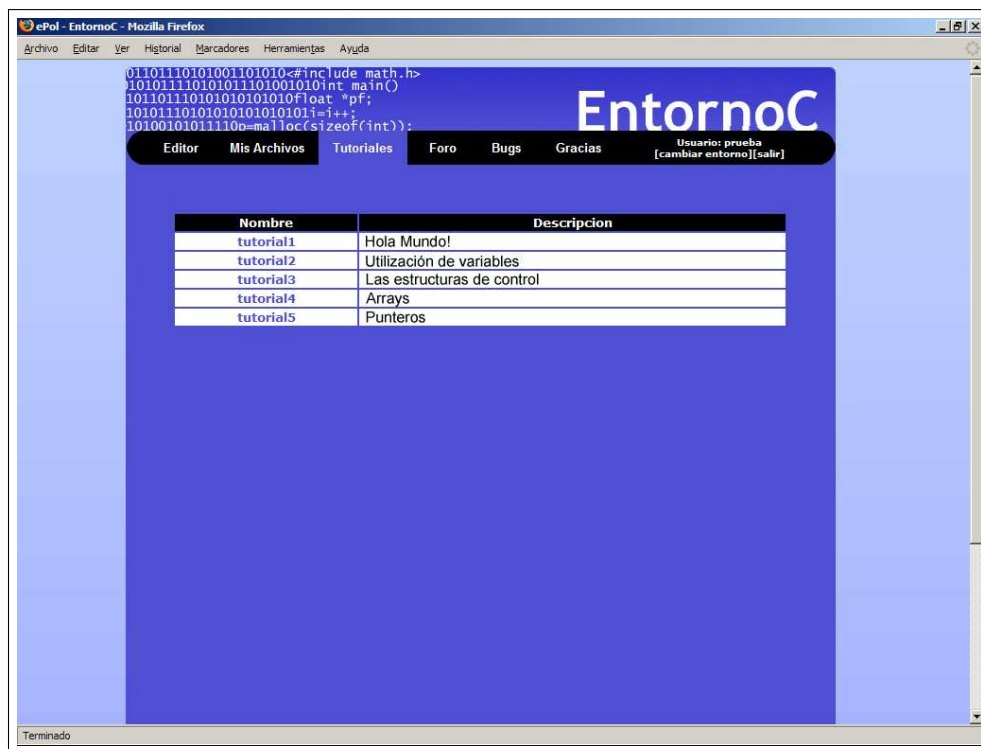


Figure 4: Multimedia Tutorials.

4 Teaching proposal with OLC

The subject "Fundamentals of Computer Science" is taught in the first year of various engineering degrees, amongst them the degree of "Chemical Engineering" of the "School of Industrial Engineering" that belongs to the University of Las Palmas de Gran Canaria. This subject, in this degree, is taught in the second term of the first academic year. Its aim is that students acquire basic knowledge of Computer Science in the theoretical classes and basic knowledge of programming in the practical classes. The theoretical hours are taught in the theory classrooms and the practical hours are taught in the laboratory. This is a classroom equipped with computers, one for each student, and connected to Internet. Programming knowledge is necessary for the use of certain tools that students will use in the following academic years, such as programmable calculators or software like Matlab. The number of teaching hours is three of theoretical lessons per week using traditional methods and one hour of practical lessons. This means fifteen practical hours during the term.

- In the first five hours of practical classes basic knowledge of programming, tools to use and C language are introduced. In the first hour of these five an introductory class of OLC will be taught. The following four hours the lessons will be taught through the video tutorials that will indicate how to make use of OLC and will ask the students to work with the examples proposed by the teachers. These first exercises will consist of simple programs that students will have to edit, modify and compile.
- In the following five hours, students will work in the modification and creation of computer programs. The proposal of modification made by the teachers will consist of making changes in programs that make use of a bit more complex structures than the ones used in the previous phase. A group of ten exercises could be completed, two for each practical hour, with an increasingly difficulty.

- For the last five hours the students must work in a practical exercise proposed by themselves. Teachers must give their approval to this proposal computer program. This will be the final exercise to evaluate and should be made with a more powerful compiler. At this stage of the learning process, students must already control the use of the compiler and its different options.

The independence acquired by the student after these stages using OLC and their control in the comprehension and use of other compiles should be noticeable. The assessment process will take into account the valuation of the supervision of their work with OLC and the final exercise proposed. According to this approach, using OLC compiler the role of teachers is greatly restated. They could dedicate time to transmitting motivation about the subject being studied and directing the students in their studies. Besides, the supervision of their work and the evaluation is a continuous process that could be done completely on-line. The role of students has also changed. The use of this kind of applications gives them more autonomy than in previous academic courses.

5 Didactic perspective

The work developed has been exposed to a group of five teachers [7] whose subjects deal with programming. All these subjects are included in the first academic year of several engineering studies. Different opinions about the exposition have been found. The features that the teachers have taken into account to give their viewpoint about OLC have been: the professional profile their students will have in the future; the use they will make of programming both in the academic and professional field, and the level of knowledge they must achieve during the academic course. To a lesser extent they have considered the percentage that programming has in the final evaluation and the number of hours they must dedicate to the practical part of the subject.

- *Teacher one.* He teaches "Basic Knowledge of Computer Science" belonging to the first course of "Chemical Engineering". To me the teaching proposal is the right one. The students find many difficulties in the understanding of compilers at the initial period of learning programming, he thinks. Furthermore, sometimes even the use of computer is complex for them, although they learn quickly. Professionally they will unlikely need to make computer programs. Nevertheless, in the following academic years, surely, they will need to understand what variables are and how to make use of them in small computer programs as well as using them in small functions created for MatLab. According to the teacher in their senior thesis, it is very probable that they will only need to manage statistics information and perhaps make some update or modification in a small computer program in order to obtain some of this information.
- *Teacher two.* She is the responsible for the practical part of the subject "Fundamentals of Computer Science". This subject corresponds to the group of training complements in the degree in Industrial Organization Engineering. In my opinion the teaching proposal with OLC is very well structured. Maybe the level of knowledge shown in the video tutorials is quite high, since generally the students who apply for this degree do not have a high interest in programming. The number of students that choose the subject is low. They try to pass the subject without paying attention to the tools they are using. Professionally it is not probable that they will make use of these programming skills in the future. That is why the percentage of the final evaluation is low, although the use of multimedia material could be proposed for other subjects, since these multimedia tools are ideal for teaching.
- *Teacher three.* He is the coordinator of the subject "Programming Methodology" belonging to the first course of "Technical Computer Science Engineering". He thinks it is a good proposal. In his department they have thought to develop a similar tool. However, they have not considered using

multimedia tutorials, just only help files. We should take advantage of these multimedia capacities. His students will surely use the programming knowledge both at an academic level and in their professional life.

- *Teacher four.* She has taught "Fundamentals of Computers" in the first year of "Computer Science Systems Engineering". She finds the teaching proposal with OLC interesting. In fact in "Fundamentals of Computer" she uses a program developed for students to use in order to create computer programs in assembly language . Thus they will not have to make use of the machine compiler. According to her, the students are well trained in Computer Science; therefore they do not need to begin with a tool so simple. However there exists a fundamental aspect of this tool that should be exploited: to have production tools in the Web. Undoubtedly they mean a great advantage, she thinks. Proposals with more functions and accessible from the Web would be very interesting for her students.
- *Teacher five.* She teaches "Basic Knowledge of Computer Science" belonging to the first course of "Public Works Technical Engineering" in all its three specialities: "Hydrology", "Civil Building", and "Urban Transport and Services". She thinks that the proposal is of no interest for what she needs, bearing in mind the number of practical lessons per week and the teaching content, which do not match together. She believes that in the future students will not make much use of this programming knowledge. It is quite complex to make them understand some concepts. We are just in a period in which not all the students who reach university studies are equally trained in Computer Science, and only some of them have got a computer at home. Maybe the practical lessons should be concentrated on removing programming knowledge from the contents of the subject.

Some students have also given their opinion about the environment. Their profile extends from students of computer science engineering in their second year at the university to some professional that already have finished the bachelor of "Computer Science".

- *Student one.* He studies the 2nd year of the degree "Industrial Engineering" (basic knowledge). The possibility of make my programs without having to install any software in my computer (I have had problems trying to install some C compilers before), makes me save time and allows me to work in the same conditions independently of the place -laboratory or home-. Furthermore with no knowledge about Linux, I could compile my first program in that operating system.
- *Student two.* He studies the 2nd year of the bachelor degrees "Computer Science Engineering" (learning C programming language) The use of OLC is ideal to learn a programming language. Firstly, it hides tiresome options that do not add anything to a beginner user, as path specifications or debug options. Moreover the availability of the video tutorials while using the editor is much handy that making programs in a conventional programming environment, where you should read the tutorial in a web page or using paper format. Finally, the operations I can carry out with "my files" option are the necessary ones for a beginner (save, compile, indent, and download the .exe file), and can be made in an easy way.
- *Student three.* He has finished the bachelor degree "Computer Science Engineering" and in a few weeks will defend his senior thesis (high level of programming language knowledge). I like the idea implemented, to have the compiler in the web allow access from any computer with an Internet connection and it is useful to compile in a specific moment with no need to download and install a compiler. For students that just make a brief immersion in the programming studies is ideal in fact, but if you study Computer Science at a professional level you find limitations. Apart of this, to download the .exe file with the result of the compilation every time is not much useful cause it is more handy the characteristic exe button of an IDE.

- *Student four.* He still has to study three subjects to finish the degree "Computer Science Engineering" (high level of programming language knowledge). For me it wasn't a trouble to install the C compiler, this tool is not the compiler I would have liked to use at the beginning. I think that when I began to make programs in C language I installed the compiler in my computer at home so I had not used OLC at all. On the other hand I cannot execute the programs made on-line and this is something that I do not like. When you compile a program the first thing you want to prove is the result, and using OLC you must download the .exe file, I prefer to compile with my IDE. The great advantage could be to compile and execute on-line.
- *Student five.* He has finished the bachelor degree "Computer Science Engineering" and is developing his senior thesis (high level of programming language knowledge). Advantages: - Easy to use. - You do not have to install any software, just only an Internet connection is necessary. - You can work from everywhere: home, laboratories, any place with an Internet connection. Disadvantages: - To download the .exe file. - Just a few options of debug - I do not like that my programs be in the Web, for security reasons. In resume: It is interesting at an initial stage of programming or for easy programs in computers where you do not have a compiler installed but apart from that it does not offer much more. It could be more attractive if you could execute your programs on -line.

6 Conclusions

This work proposes an interesting initiative in the field of e-learning: to portray interactive applications on-line for didactic use [8]. A simple compiler accessible from the Web provides students with an easy way to learn at the initial stage of programming studies. OLC makes use of multimedia contents to support teachers in the teaching process. These contents show how to work with the on-line compiler and with the programming language [9]. Apart from these there are further advantages, such as having at students and teachers' disposal [10] the compiler through the Web and the possibility of making use of it at any time and from any computer with an Internet connection. Thus barriers of time and space are eliminated. Students will have access to the compiler, the contents and the files with which they have been working. Besides, there is no need to save the work in store devices nor to install software. To have remote virtual laboratories increases their efficiency [11] and reduces the costs of the resources used in learning [12]. OLC could not compete with the powerful traditional IDE's due to its limitations for the expert software development, but undoubtedly it fills the void that exists in the area of teaching programming at the initial stage. This experience has already passed from the development phase to the experimental one. At present we are working on a research of those subjects that include programming languages lessons, and the use of OLC could be suitable. As many other initiatives, to promote its acceptance and to achieve its use depend on Programming teachers, who have already valued the application positively and favourably. According to their opinion, to introduce the students in the field of programming is hard in the initial stage. Not only for the complexity of the control and data structures and the programming language to learn, but also for the complexity of the tools they need to use, such as compilers, bearing in mind that students only need to make use of a reduced number of functions at their initial period of their programming studies. Using this kind of applications with multimedia contents allow teachers to dedicate time to activities which improve the quality of teaching, transmitting a further motivation about the subject to be studied and the results they can obtain, as well as directing students closely in their studies. [13]. In the near future, we will continue this line of work. Our new projects involve the development of other applications accessible from the Web [14], used as OLC, in e-learning field. Some of these applications will count on a key feature: interactivity [15]. Thus students will develop new working methods [16] in order to ensure participation [17].

7 Acknowledgment

This article is an extended version of our paper [18], published in Proceedings of the 2nd International Conference on Virtual Learning Ū ICVL 2007 (M. Vlada, G. Albeanu, D.M. Popovici, eds).

We wish to thank the program committee of ICVL 2007 that select and recommended the publication of our extended work in IJCCC.

Bibliography

- [1] P. Brey. "Evaluating the social and cultural implications of the internet". Volume 35, Issue 3 (September 2005). ACM Press.
- [2] C. McHugo, T. Hall. "Enhancing ICT learning with Real-life problems". EMRC University of Limerik."
- [3] T. O'Reilly. "What Is Web 2.0 - Design Patterns and Business". Models for the Next Generation of Software
- [4] A. Yong-yeol, H. Seungyeop, K. Haewoon, M. Sue, J. Hawoong. "Semantic web and web 2.0: Analysis of topological characteristics of huge online social networking services". Proceedings of the 16th international conference on World Wide Web WWW'07. ACM Press
- [5] T. Hartsell, S. Chi-Yin Yuen. "Video Streaming in Online Learning". AACE Journal. 14(1), pp. 31-43. Chesapeake, VA: AACE.
- [6] L. Neal. "Predictions for 2006: e-learning experts map the road ahead". Volume 2006 Issue 1 (January 2006). ACM Press. New York.
- [7] S. Anderson. "Working together to develop a professional learning community". HERDSA 2002
- [8] G. Lautenbach. "Engaging with e-learning: emerging epistemologies and increased pedagogic vision in higher education". IADIS Inter national Conference e-Learning 2007
- [9] R. Rheeder, R. Diseko, G. Lautenbach. "The design of interactivity for a web based learning environment at a higher education institution." IADIS International Conference e-Learning 2007
- [10] H. Samaras, T. Giouvanakis, D Bousiou, K. Tarabanis. "Towards a New Generation of Multimedia Learning Research". AACE Journal. 14 (1), pp. 3-30. Chesapeake, VA: AACE
- [11] Lawson, E.A. Stackpole, W. (2006). IT education - online education: Does a virtual networking laboratory result in similar student achievement and satisfaction?. Proceedings of the 7th conference on Information technology education SIGITE'06. (October 2006). ACM Press.W. NejdI, M. Wolpers "European e-Learning: Important research issues and application scenarios". L3S Research Center and University of Hannover.)
- [12] W. NejdI, M. Wolpers "European e-Learning: Important research issues and application scenarios". L3S Research Center and University of Hannover.
- [13] K. Fraser, "A blended Learning Approach to Teaching" Introduction to Multimedia, The E-Bit!. AISHE Conference 2006
- [14] A. Leene. "The MicroWeb". Using Content in theory and practice MicroLearning 2006, Innsbruck - 7 June 2006

- [15] P. Valente, P. Sampaio. "Defining Goals for the design of Interactive Multimedia Documents". In Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2006 (pp. 955-962). Chesapeake, VA: AACE.
- [16] N. Sala. "Web Based Learning Environment: an Example". In Proceedings of 2000 (pp. 807-808). Chesapeake, VA: AACE.
- [17] C. Gráinne, "Making sense of today's technology-enhanced environment for learning: rethinking student and teacher roles" IADIS International Conference e-Learning 2007
- [18] Cayetano Guerra Artal, Maria Dolores Afonso Suarez, Idafen Santana Perez, Ruben Quesada Lopez, "OLC, On-Line Compiler to teach programming languages", In (M. Vlada, G. Albeanu, D.M. Popovici, eds) Proceedings of the 2nd International Conference on Virtual Learning - ICVL 2007, October 26-28, 2007, Constanta, Romania, pp. 283-290, 2007.

Cayetano Guerra Artal, Maria Dolores Afonso Suarez
Idafen Santana Perez, Ruben Quesada Lopez
Instituto Universitario de Sistemas Inteligentes y
Aplicaciones Numéricas en Ingeniería IUSIANI
Universidad de Las Palmas de Gran Canaria
Departamento de Informatica y Sistemas
Edif. de Informatica y Matematicas.
Campus Universitario de Tafira. 35017. Las Palmas. Spain.
E-mail: cguerra@iusiani.ulpgc.es, mafonso@dis.ulpgc.es
idafensp@gmail.com, rubenql@gmail.com

Received: November 4, 2007



Cayetano Guerra Artal, graduated at the University of Las Palmas de Gran Canaria in Computer Science. He has been teaching Multimedia for seven years at the Computer Science and Systems Department of the University of Las Palmas de Gran Canaria. He received his Ph.D. in Computer Vision. His main research fields are Computer Vision, Computer Graphics, Multimedia and e-Learning.



Maria Dolores Afonso Suarez, graduated at the University of Las Palmas de Gran Canaria in Computer Science. She has worked in private enterprises developing software and as associated professor at the Computer Science and Systems Department of the University of Las Palmas de Gran Canaria. Her main research fields are Software Engineering, Multimedia, and e-Learning.



Idafen Santana Pérez, graduated at the School of Computer Science of the University of Las Palmas de Gran Canaria. He is a pre-graduated student of the Faculty of Computer Science of this university. His current research interests are in e-learning applications and natural language processing.



Ruben Quesada Lopez, graduated at the School of Computer Science of the University of Las Palmas de Gran Canaria. He is a pre-graduated student of the Faculty of Computer Science of this university. His current research interests are in e-learning applications and natural language processing.