Visual Teaching Objects for Programming Structures Using Presentation-Based Tools

Abdullah Ahmad Basuhail

Abstract— This paper presents an approach for the design and implementation of presentation-based visual animations that we have adopted during the teaching of the structures, techniques and algorithms in computer programming courses. Specifically, presentation-based animations and graphics are utilized to illustrate the concepts of programming to novice undergraduate science and engineering students. The developed components are considered as teaching objects for the programming topics and concepts. The contents can serve as hands-on experience for the learners and trainees that can help them in visualizing the flow of control and execution process of the programming structures. Using the developed teaching objects by this methodology does not require any external software, plug-ins, or add-in modules. The main features of our approach are effectiveness, attainability, and rapidness of the development process. It is also characterized by the reduction of the manpower and cost. The teaching objects that we developed here are also characterized by the reusability and integration into e-learning systems' environment. This technique could be extended to other fields and disciplines of science, technology and engineering.

Index Terms— Computer programming, Programming structures, Teaching objects, Animations, Computer graphics, e-teaching, e-learning, Presentation-based tools.

1 Introduction

USING technology in teaching is a predominant approach in contemporary education. It assists the instructors in conveying information and concepts in consistent, systematic, attractive and smooth manner. It also enhances the ability of the learners in grasping the concepts in more precisely and timely manner.

Animations are among the current tools that can be used for teaching enhancements. Educational animations are defined as "animations produced for the specific purpose of fostering learning [1], and assist in teaching and conveying the subject matter". The popularity of using animations to help instructors and authors of e-content to convey information and explanations has greatly increased recently as a consequence of the magnificent enhancements in the capabilities of the computer systems and the continuous development of the image and graphics oriented software packages.

Animations can be used to implement and develop e-teaching objects. A teaching object is defined as a small visual digital content, or practice items that can be reused to support courses' teaching. Instructors are able to implement teaching objects using electronic technology-based tools. Teaching objects can be fortified with animations and computer graphics.

Previously, implementation of educational animations have been constrained to only experienced and well trained authors. They have required the use of intensive techniques that have been exhausting the resources in terms of time and budget.

Nowadays, several computer software packages are existing for instructors and authors to assist them in conveying their expertise and explanations for the educational concepts and ideas. For the authors, these packages facilitate the process of converting the topics and concepts into animated teaching objects. Instructors are no longer limited to rely on static teaching objects related to their topics, but can readily convert them into more efficient, assisting, vivid, and dynamic electronic educational animations.

2 Problem Statement and Significance

During the teaching of the college-level introductory programming courses, we have been faced with the problem that most of the novice students have difficulties to understand and grasp the mechanism, logic, and flow of execution of the programming structures and their functionalities. Our observations from teaching those courses for several years revealed that a significant number of students could not visualize the logic and the functioning mechanism of the programming control structures.

We aimed to use modern technological tools to assist instructors in teaching and to help students to understand different topics in the programming courses and to simplify the topic matters of the programming structures specifically. A second goal we intended to achieve is to develop simple and effective teaching objects keeping in mind the consideration of the cost and time constraints. A third goal is to develop for in-session use teaching objects that can assist instructors in teaching instead of using the traditional tools such as markers, arrows, boards, etc. One more goal for implementing teaching objects is to convert them into learning objects that can be easily used by the students during the off-class sessions or they can be integrated into e-learning environments.

Traditional techniques used in the field to support in teaching, such as flowcharts, can help but it lacks the dynamics that

- *Dr. A. Basuhail is currently an assistant professor in the computer science department at King Abduaziz University, Jeddah, Saudi Arabia. E-mail: abasuhail@kau.edu.sa* can assist in following the flow of execution without a facilitator. Using educational animations saves time and efforts of the instructors who uses them during teaching sessions. The saved time can be exploited in more teaching elaboration and discussions. Moreover, the developed educational animation objects could serve as hands-on experience for the students and the trainers. Generally, teaching objects help students and learners to easily grasp the topics and allow replicating complex or expensive experiments in an easy and tangible manner.

We used to deliver lectures using technology tools such as computers, presentations' software, and data show devices, that nowadays become crucial in modern classroom and teaching environments. These tools could be exploited to enhance teaching and learning and to attract the attention of the students during in-class teaching sessions.

We proposed the use of animations to help students to understand these topics and assisting instructors of the programming courses during conveying the topic matters. Due to the time and cost limitations, and software integration and compatibility of developing educational animations that serve our purposes, we decided to use presentation-based animations. The design methodology, presented in this paper, for developing animated teaching objects can help and assist in teaching. Using the proposed technique can help in time and cost reductions due to the simplicity of implementation and attainability of the used tools, but also not at the expense of the quality or the way of use.

There are some limitations for the presentation packages with regard of the flexibility and degree of control of the developed teaching objects, but generally for the most of the general instructors' use it can function properly to a satisfactory extent.

2.1 Literature Review

This research explains the use of presentation-based animations in the implementation of teaching objects that could be used by instructors in the class sessions as well as they can be converted into learning objects.

Educational animations have been used widely and attracted the attention of instructors, learners, and researchers in education and computer science fields. Implementing and developing of such objects could be intended for either the learning process or the teaching process.

Some researchers reported that using visualization tools such as animations could be used to give an accurate and rich picture of dynamic nature for some topics which are often very hard to grasp from text-based presentations of information [3], [4].

Researchers demonstrated an active way to use computer animation in molecular genetics class that has helped students to work interactively with a computer animation which deals with abstract concepts and processes in molecular biology [3]. They reported that the analysis of the post-test showed that the mean score of the experimental group was significantly higher than the mean score of the control group. This positive trend was also reflected in personal interviews [3].

Some researchers [4], explored the effect of computer animations on college chemistry students and found that instruction with animations may increase conceptual understanding by prompting the formation of dynamic mental models of the particulate nature of matter. In this type of instruction, animations provided more scientifically correct visual models for submicroscopic processes.

Some other researchers have argued that some of the main advantages of the use of computers for instruction are the possibility to represent phenomena as a sequence of animation, adding a temporal component; the possibility to manipulate data sets, and to test the impact of the manipulation [5], and increasing students' motivation [6].

Other researchers [7], have demonstrated that animated software can illustrate and elucidate biological concepts more clearly and effectively than more traditional means—lecture, discussion or even conventional laboratory activities.

Interactive models can address core ideas in a visually engaging way that makes them accessible to students with vastly differing learning styles [3]. As students work with the models, they are more likely to remember and transfer their learning to new situations [8].

Researchers concluded that simulations furnish suitable cognitive environments within which learners can exercise a range of inquiry skills. They have mentioned that many educators have found virtual learning environments to be powerful, in-dispensable tools for developing conceptual understanding [6]. Some other researchers conducted experimental study regarding the use of animation to learn vocabulary [9].

Learning objects have been developed and used in supporting students' learning of data structures [13] and technical drawing [14]. The multimedia animation has been used in physics experiment teaching [15], [16]. Moreover, 3D animations [17], [18] and algorithm animation and its applications in instruction [19] have been used in teaching.

In programming, visualization techniques for learning and teaching has been applied and used [20], [21], [22]. Another research has concentrated on the teaching of object-oriented programming with games [23] and video [24].

A study has found that animations and simulations in courses enabled students to grasp difficult concepts more easily and have improved the quality of course design [25].

3 Educational Animations

Animations can efficiently portray temporal changes, "sequential changes over time". For this reason, they seem ideally suited for the teaching of processes, events, procedures and the like. Using animation to present educational content can reflect two kind of changes; specifically, changes in the position and/or the changes in form for some or all of the components that are fundamental for implementing an electronic object for teaching the subject matter.

In contrast with static teaching objects, animations can show temporal change directly (rather than having to indicate it indirectly using auxiliary markings such as arrows and motion lines) [1]. Using animations instead of static graphics removes the need for these added markings so that displays can be not only simpler and less cluttered, but also more vivid, engaging, and more intuitively comprehended [1].

With animated depictions, information about the changes involved is available to be read straight from the display and it can assist the instructor during the presentation sessions. Well designed and implemented educational animations for teaching objects could be delivered to the learners for further self-study and independent use without a facilitator and without the learner's need to perform mental animation or interpretation of the topic in the off-class times.

Animations could be implemented in several ways. One way is the use of graphic-oriented animated diagrams by making the animations to occur over several frames. This can be used in constructing the proposed educational animation, but it requires an extra effort in the case of updates or modifications of the teaching object. Sometimes, such technique might produce an enormous file size due to the replication of images and graphics items, which is an undesirable trend in sharing and transferring environments.

Other sophisticated animation software, such as Flash, can be used to build sophisticated teaching objects, but it imposes the use of these objects externally or installing plug-ins to use and embed into presentations with some experienced problems of integration functioning and compatibility. Another obstacle with the implementation using such packages is the need for the authors to learn and get experienced with the use of new software. Moreover, it requires dedicating a considerable time for the process of development and implementation of objects due to the relative difficulty in the use compared with other user-friendly and cumulative experience software packages.

The presentation software packages have an easy-to-use and straightforward animation capability that, in the right hands, can produce very effective and informative educational animations in an easy manner.

Using presentation-oriented animations and graphics software can save the time and effort because it exploits the cumulative knowledge and experience that the instructors gained from familiarity and earlier use of such software packages and build upon them.

Presentation software is usually used by most of the instructors to assist them in lecturing and delivering the teaching topics. As a consequence, using presentation-based software packages for implanting teaching objects have the advantages of rapid implementations and assurance of functioning, integration and compatibility without the use of any extensional or external software packages.

4 Design and Implementation Model

The first step in the implementation of a presentation-based teaching object for an educational concept, procedure or task is to design and construct a model for the desired topic. The design can be developed and implemented using the tools which are comprised of computer systems and proper software packages.

The developing of an electronic teaching object starts by designating a topic matter. This can be considered as a small chunk of information, process, or procedure to be and demonstrated to the students. To have better and easier control over the implementation phase, the selected topic matter has to be as small as possible "divide-and-conquer".

Figure 1 illustrates the design methodology cycle and its stages that can be followed in the development and implementation of an animated presentation-based teaching object. After completing the implementation cycle, the stages have to be revisited for future modifications and enhancements. This revision guarantees integrity of the content and precise functionality of the final working product.

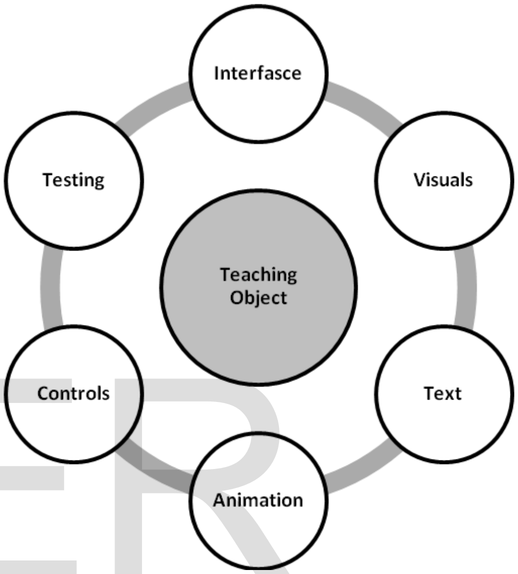

Fig. 1. Stages of implementing presentation-based teaching object.

The subsequent paragraphs explain the proposed design methodology for implementing a teaching object using simple and attainable tools taking into consideration instructors who are the non-specialized or non-experienced in developing teaching objects.

4.1 Interface Design

In this stage, a user interface for a presentation-based educational teaching object is set. This is done by crafting a certain template to be used for all the subsequent implementations. This stage may also comprise selecting proper colors and backgrounds to be used and unified in the entire design process. Proper labeling are inserted next to each used item in order to aid the users to use the teaching object. The overall view can be improved by adding explanatory titles. Adding other features and setting properties, such as borders and shadows, enhance the visualization of the designed teaching object.

4.2 Adding Visuals

This stage in the development process deals with selecting and adding proper topic related drawings, graphics, clipart and images. Ready-made graphics collected from outside resources such as web-sites, (see [11] and [12]) can be used for faster implementations. Another way to develop images and drawings is by using imaging and drawing tools and/or computer software.

For better topic understanding and concentration, it is more desirable to use simple and meaningful visuals. An overwhelmed teaching object might distracts and confuses the learners.

4.3 Text Entry

In this stage, the author adds theoretical information related to the topic matter, such as: text, captions and/or instructions. This step makes the teaching object to be more significant and straightforward. A well designed and self-explained teaching object can be used as an independent learning object by the learners.

4.4 Animation Sequencing

The animation stage is the most important phase in the implementation. It might consume the longest time and effort during the entire design process. Compiling-up animations over the different graphical and textual components of the object is set in this stage. This is done by applying temporal transitions driven by the teaching topic under consideration. The assorted presentation-based animations must be selected precisely to reflect and mimic the intended teaching topic in the mind of the instructor and according to the mechanism of the considered topic. In a presentation-based animation software packages, an assorted animation effects can be selected from. Specifically, "Entrance", "Emphasis", Exit, and "Motion paths" animation categories with multiple effects under each category are available and can be selected from. They can be sequentially cascaded over different components of a teaching object. Moreover, animation transitions and proceeding can be controlled by mouse clicks, timings, or can be linked to navigation or action buttons using triggers. This task can be performed according to the need of the instructor and/or be steered by the topic matter.

4.5 Adding Controls

Controls and action buttons are used for navigation and transitions. They are properly incorporated and integrated in this stage of the design. They offer flexibility and easiness when using the developed teaching objects. The controls require programming and linking to the appropriate object's components for accurate operation and proper functioning.

4.6 Testing

The implementation process might end up with a relative complex teaching object that includes drawings, images, textboxes, captions, etc, superimposed with assorted cascaded animation affects and timings. These blended components require testing before use or delivery. The last stage in the design process that precedes distributing and delivering of a developed teaching object is to run a thoroughly test to validate the proper operation and functioning of the entire teaching object as a standalone or integrated whole object. In case of an object have malfunction operation, or being not in accordance with the design specification, the corresponding design stage is revisited and validated to meet the intended requirements.

Testing takes place as an incremental testing module after the completion of the design and implementation of each individual stage and step of the teaching object. Testing is also conducted as an integral module after the completion of the entire design process for validating the functionality of the overall object.

5 Applications

Computer programming languages include various programming structures. These structures are used to implement computer applications and systems and to solve computational problems using computers (see [10] for more detail).

The approach presented in this paper has been applied and used to implement teaching objects for all the programming structures; specifically, the structures: *if* selection, *if-else* selection, *nested-if*, *while*, *do-while*, and *for* repetition. Some other structures related to object-oriented, data structures, and computer algorithms have been developed and implemented as well. The next sub-sections will highlight the implementation of two commonly used structures, namely: if-else and do-while programming structures. The developed structures have been used in the classes of teaching programming and related courses.

5.1 if-else Programming Structure

Instructors use static flowcharts to explain this programming structure. Figure 2 illustrates a teaching object for this programming structure based on the flowchart approach. Animations and graphics are augmented to the exhibition to aid the instructors during the teaching task. Specifically, memory chip is added to represent storing of the data during the execution. A keyboard that symbolizes the entry of data to the structure is used. A monitor that stands for the output of the structure is also added. A control button is added to steer the teaching object by the instructor during the teaching sessions.

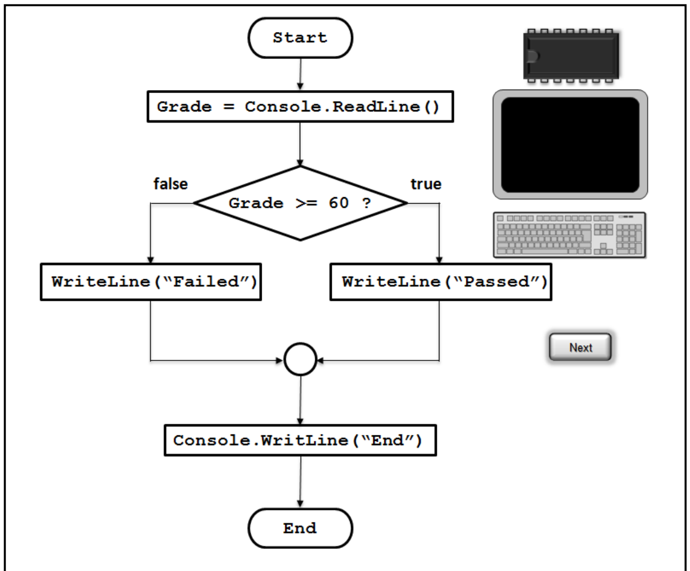

Fig. 2. Teaching object for the if-else programming structure. Flowchart is used to show the flow of execution sequence, memory chip for data store, keyboard for input, monitor for output and control button.

Table 1 lists all the actions that have been superimposed over the teaching object's components. It reflects components that are visited, the flow of control and the actions and changes that will take place during the execution phase of this programming structure.

TABLE 1
ANIMATIONS/ACTIONS CASCADING OVER THE IF/ELSE PROGRAMMING STRUCTURE'S TEACHING OBJECT

Execution Step	Animations/Actions Cascading
1. Start	box flash, change fill-color
2. Arrow (in flow of control)	change color and width
2. Grade =Console.ReadLine()	box flash, change fill-color show pressed keyboard buttons insert entered data into memory display entered data to monitor
3. Grade >= 60 ?	box flash, change fill-color
4. WriteLine(“Passed”) or WriteLine(“Failed”)	box flash, change fill-color display output to monitor
5. Console.WritLine(“End”)	box flash, change fill-color display output to monitor
6. End	box flash, change fill-color

Figure 3 shows one frame for the in-action teaching object of the if-else programming structure. The input from the input device and the output from the structure are directed and displayed on the monitor as shown in the figure.

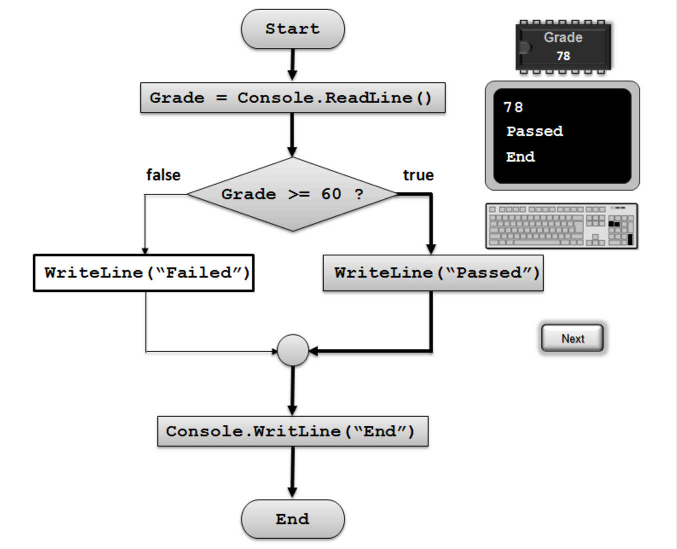

Fig. 3. In-action teaching object for if-else programming structure. This shows the input entered by the user from the keyboard, the action of storing the entered data into the memory, the action of the output displayed to the monitor and the flow of control from the start to the end according to the mechanism of this programming structure.

In order to assist the instructors in teaching and the students in following-up with the execution cycle, the place of execution in the structure is flashing during the teaching process of this structure. This reflects the mechanism of operation of the structure. It also supports to keep track with the order and the flow of operation. Figure 3 shows one frame for the in-action teaching object of the if-else programming structure.

The teaching object for this programming structure is very helpful and informative to the novice students and aid them to better comprehend this programming topic.

5.2 do-while Repetition Programming Structure

The second example for the implementation of a teaching object for programming structure selected to be demonstrated here is the do-while. A table similar to Table 1 is set for this structure to prepare all the actions that will be superimposed over the exhibition. Figure 4 shows the last frame for the execution of this structure.

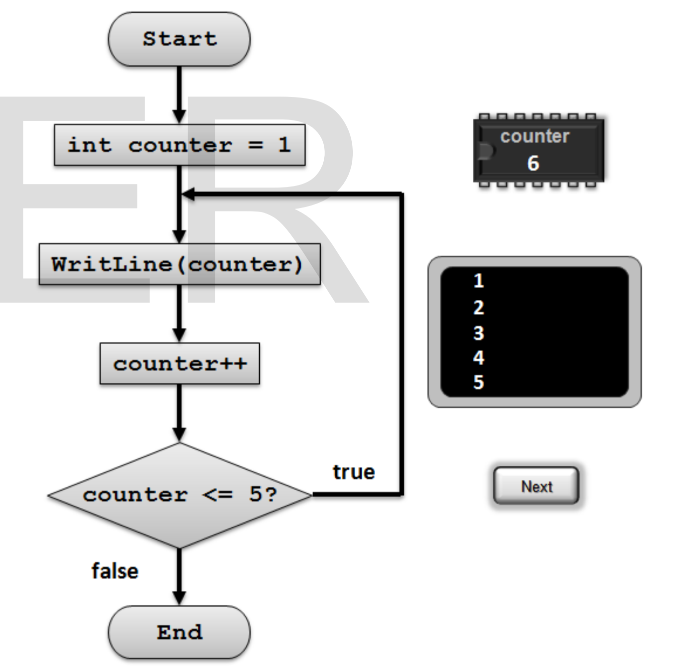

Fig. 4. In-action teaching object for the do-while programming structure. This shows the action of storing the set data into the memory, the action of the output displayed to the monitor and the flow of control from the start to the end according to the mechanism of this programming structure.

This teaching object reflects the idea of the repetition loop by showing the steps inside the loop will be animated five times during the use of this teaching object. All the corresponding actions, such as memory and monitor use, are used accordingly.

The in-class dynamic animations have been demonstrated to the students. The students have better perception and visualization for the concepts of the flow of control, looping and repetition, the change of memory contents, the displayed output to the output device during the execution of the structure.

The other programming structures have been implemented and demonstrated to the students in a similar fashion and according to their respected way of functioning and sequencing.

6 Results and Discussions

We have used the developed teaching objects for the programming structures and shared them among our colleagues and computer science students during several academic semesters.

Students have reported advantages of teaching objects use in teaching of the programming structures. Students have mentioned that "animations helped them to visualize the concepts and processes of programming structures by representing the subject matter in a more visualized manner." The teaching objects have demonstrated the process, and have facilitated the connection between the subject components. Some students have commented that it is much easy to visualize the concept, to understand it, and to remember it in this way of teaching. Another advantage of using the teaching objects is that students have the chance to use the objects to work with and manipulate in their own time, and to run them over and over as much as they need.

From our observations and comments of other instructors involved in the use of the implemented teaching objects, students have showed better and faster understanding and increasingly participation in the topics covered using this approach.

The instructors who used the compiled teaching objects in their classes were very enthusiastic and expressed their satisfaction and appreciation with this way of teaching. Some have commented that it gave them an opportunity to gain insight into students teaching process and saved times during teaching the related topic matters compared with the traditional teaching approaches. That saved time has been exploited in education by giving the instructors more time that they have utilized that time for extra course activities and discussions. Some instructors have expressed their wish to apply the model of presentation-based animations on other topics as well.

The approach has been shared with other instructors which attracted their attentions and they declared its effectiveness Furthermore, the experience of teaching objects has been applied in other computer science and technological courses as well.

7 Conclusion

This paper demonstrated use of presentation-based tools to implement teaching objects. The implemented objects have been adopted during the teaching and delivering of the basic computer programming structures in undergraduate computer programming courses. Specifically, electronic contents for topic matters have been developed. The teaching objects are characterized by the simplicity and rapidness of implementation. Other features include attractiveness, easiness of use and manipulation. Teaching objects play a role of in aiding the instructors during the class sessions. Moreover, when using the developed objects using the presented approach, there is no need for any external software, plug-ins, and/or add-in modules. The learning objects that have been developed here are also characterized by the reusability and possibility of integration into e-learning environments. The approaches and techniques have been applied in our courses and trainings could be, as well, implemented in other fields and disciplines of science and technology.

As a conclusion, it is recommended to use the methodology explained in this paper to develop teaching objects. It is also recommended to encourage the instructors in the education institutions to get benefit of the approach discussed here to implement teaching objects due to their effectiveness, rapidness of the development, and the reductions in the effort, time, and cost. Another recommendation is to raise awareness among instructors for the importance of the use of electronic educational materials and teaching objects developed using the mentioned techniques and encouraging such trends. It is recommended to hold short-courses, practical sessions and workshops for educating staff members about the production of educational materials and teaching objects using simple and attainable tools and building upon their cumulative experiences. This approach is more efficient and attractive to their desires and it will be more productive, time-wise. Finally, it is recommended that instructors share their developed teaching objects with colleagues in other institutions. This kind of sharing opens the floor for exchanging ideas and enhancing and enriching teaching process of sciences technology.

References

[1] Wikipedia, the free encyclopedia, from http://en.wikipedia.org/wiki/Educational_animation.

[2] Educational Benefits of a Computer, from http://www.ehow.com/about_5427864_educational-benefits-computer.html#ixzz2GsgIbCqL.

[3] Y. Rotbain, G. Marbach, R. Stavy, "Using a Computer Animation to Teach High School Molecular," Journal of Science Education and Technology (Vol. 17, Is-sue 1, pp. 49-58). Springer Netherlands, 2008.

[4] V. Williamson, M. Abraham, "The effects of computer animation on the particulate mental models of college chemistry students. Journal of Research in Science Teaching," (Vol. 32, pp. 521–534), 1995.

[5] D. Gordin, R. Pea, "Prospects for scientific visualization as an educational technology," Journal of Learning Sciences, Vol. 4: pp. 249-279, 1995.

[6] M. Windschitl, "A practical guide for incorporating computer-based simulations into science instruction," American Biology Teacher, (Vol. 60, pp 92-97), University of California Press, USA, 1998

[7] P. Matray, S. Prolux, "Integrating computer/multimedia technology in high school biology curriculum," American Biology Teacher, (Vol. 57, pp 511-520), University of California Press, USA, 1995.

[8] B. Berenfeld, A, Pallant, B. Tinker, R. Tinker, Q. Xie, From genetic code to protein shape using dynamic modeling. Proceedings of the 77th Annual Meeting of the National Association for Research in Science Teaching [NARST], Vancouver, Canada, 2004.

[9] M. Kayaoglu, R. Akbas, Z. Ozturk, "A small scale experimental study: using animations to learn vocabulary," The Turkish Online Journal of Educational Technology, Vol. 10, Issue 2, pp.24-30, 2011.

[10] H. Deitel, P. Deitel, Visual C# How to Program. 5th Edition, Prentice Hall Publishers, NY, 2013.

[11] Images and graphics databases, from http://www.iconarchive.com/.

[12] Images and graphics databases, from http://all-free-download.com/free-vector/.

[13] K Chansilp, R. Oliver, "Reusable and Sharable Learning Objects Supporting Students' Learning of Data Structures in University Courses," Proceedings of the EDU-COM 2006 International Conference. Engagement and Empowerment: New Opportunities for Growth in Higher Education, Edith Cowan University, Perth Western Australia, pp 105-113, 2006.

[14] R. Osz, K. Robert, "The use of animations in teaching technical drawing," The 7th IEEE International Symposium on Applied Computational Intelligence and Informatics, pp 311-314, 2012.

[15] Y. Huang, "The Application of Multimedia Animation in Physics Experiment Teaching," International Conference on Multimedia Technology (ICMT), pp 3258-3261, 2011.

[16] U. Reddy, B. Sripada, R. Kulkarni, "How to Teach Medical Concepts Using Animations by Designing Multimedia Instructional Application.," 2006 IEEE International Conference on Systems, Man, and Cybernetics, pp 1778-1785, 2006.

[17] F. Campos, S. Bolívar, A.Rodríguez, J. Villanueva, J. Tejada, P. Bullejos, J. Carceller "3D Animations Used for Teaching," EAEEIE Annual Conference, pp 1-5, 2009.

[18] A. Salim, S. Hassan, S. Hamdi, S. Youssef, H. Khattab, M. El-Ramly, A. Zweil, "On Using 3D Animation for Teaching Computer Programming in Cairo University," The 7th International Conference on Informatics and Systems (INFOS), pp 1-5, 2010.

[19] B. Jin, M. Jin, X. Xue, "Algorithm Animation and its Applications in Instruction," The 3rd IEEE International Conference on Ubi-media Computing (U-Media), pp 272-276, 2010.

[20] L. Baldwin, J. Kuljis "Visualization Techniques for Learning and Teaching Programming," 22nd International Conference on Information Technology Interfaces IT1, pp 83-90, 2000.

[21] D. Dekson, E. Suresh, "Intelligent System to Teach Programming Languages," International Conference on Intelligent Agent & Multi-Agent Systems, IAMA 2009, pp 1-5, 2009.

[22] I. MAZILU, "Learning Objects for Java Programming Language," Conference Proceedings of Learning Technologies and Mathematics Middle East , Sultan Qaboos University, 2007.

[23] L. Yan, "Teaching Object-Oriented Programming with Games," Sixth International Conference on Information Technology: New Generations, ITNG '09, pp 969 - 974, 2009.

[24] H. Chen, "Video Case Teaching Method in Programming Courses," The 8th International Conference on Computer Science & Education (ICCSE 2013), pp 1098-1101, 2013.

[25] J. Mtebe, H. Twaakyondo, "Developing and using animations and simulations to teach computer science courses," International Conference on e-Learning and e-Technologies in Education (ICEEE), pp 240-246, 2012.