

Dictation introduced at lectures as a practical method to increase the effectiveness of teaching software engineering in a digital era

Razumowsky A.I.

Trapeznikov Institute of Control Sciences RAS
Moscow State University of Technology STANKIN
Moscow, Russia
razumowsky@yandex.ru

Abstract — The article discusses prerequisites for the need to change the lecture form of training at a university. This results from the fact that during the oral presentation by the lecturer, the student rarely began to use notes. Instead, the student takes photos of the notes on the information board or presentation slides using a smartphone or tablet. The possibility is discussed to increase the effectiveness of teaching Software Engineering by developing the ability to "think with your hands" as a practice of creativity. In that regard, an initiative is proposed to introduce a dictation at a lecture, which requires that the student take notes. Such an innovation will allow for not only developing and strengthening the skill of writing any program code, but at the same time, for verifying its correctness individually. It is expected that regular lecture dictations will allow students to more effectively improve on skills of algorithmic written thinking when the thought being initially fettered by obscure or unusual characters gradually gets used to them. As we might expect, a creative insight will occur in a student's mindset at some point, which will determine the beginning of understanding. This moment will mean the initialization of an individual educational process. At these moments, mental effort will simultaneously begin to decrease due to an emerging or improving skill. The introduction of a lecture dictation will make it possible to practically provide training for a wide audience of the skill to record basic elements of Software Engineering. Furthermore, a dictation regularly used will create more comfortable conditions for creative assimilation of the procedure for using software resources. A dictation is able to add extra focus on the proper handling of random data. And, most important, the student has a higher responsibility for software implementation, which means that the foundation is laid for creating reliable, fault tolerant software systems.

Keywords — programming, programming training, lecture, think with your hands, dictation.

I. INTRODUCTION

This is the time that is characterized by active use by students of modern information technology means for interaction and information management: smartphones, tablets, e-books. For this reason, an extremely urgent issue is to change traditional teaching forms, primarily lectures.

Pursuant to [1,3], a lecture has the following main advantages:

- It gives the lecturer the opportunity to disclose unpublished or not always available materials to students;
- It allows the lecturer to accurately determine the objectives, content, organization of the material, and to set the pace and rhythm of its presentation. However, student-centered methods, such as discussions or lab work, require the teacher to be able to manage the process when faced by unforeseen situations, questions or unexpected remarks;
- It allows you to arouse interest in the subject;
- It provides an opportunity to supplement and clarify text materials;
- It involves transferring knowledge to a wide audience.

However, the lecture form of education also has some disadvantages [2]:

- the student is inactive;
- a natural obstacle is created between the lecturer and the student due to the lack of full discussion, which leads to an incomplete understanding of the material by the student;
- a significant amount is required for the student outside the lecture time to understand and deeply percept the lecture contents;
- the lecturer's highly-effective oratory and cognitive skills are necessary and important.

Quite a lot of works have been focused on the study of different lecture forms, both in the educational sphere and in the context of practical philosophy. Thus, the work [4] studied a phenomenon of previously recorded lectures instead of full-time lectures and their impact on class attendance and the quality of students' thinking. The work [5] is devoted to the issue of creating a feedback context for lectures. In such work, the authors aimed to find out whether it is possible to create such a context of the lecture environment that would increase the students' ability to control their own learning. The work [6] studied the possibilities of game mechanics in order to move away from routine activity in training. The work [7] captures a significant increase in the conceptual perception of

knowledge at interactive lectures. The work [8] is devoted to the study of how the quality of the material assimilation depends on the volume of lecture space. Unfortunately, there is not a single work found which is related to lectures organized in terms of students' creative activity. Meanwhile, the effectiveness in transferring knowledge is to create imaginative and comfortable learning environment.

II. METHODOLOGY

Each lecture is a dual process of introducing the essence of the matter and its perception. The perception is due to personal interest and ideas born in the mind. Usually, the process that determines the search and acquisition of ideas is correlated with the intelligence. However, there is yet another opinion among psychologists that the author of the idea is the creative side of human thinking [9,10]. Here, the mental activity should be considered in terms of its productivity and practicability. In other words, it is important to isolate the factors of emergence and support of the creative process, as well as to follow the results of creativity in terms of their educational value.

When it comes to the creativity, 4 stages it is divided into are usually referred to, which are described by the English teacher Graham Wallace in 1926 [11], although similar explanations had already been made by other authors (P. Engelmeier and T.-A. Ribot [21,22]) by that time. So, the stages of the creativity are analysis of an issue, incubation, illumination (insight) accompanied by emotional experience, and verification. In the educational process, it is very important to rely on the psychological factor of the sudden birth of a solution, understanding an idea, and illumination that is insight.

The division of creativity and intelligence can be partly traced in the works by D. Guilford [12] and Ya. A. Ponomarev [9], although such division is rather conditional than essential because there are still enough blind spots in understanding the psychology of the processes of learning and perceiving ideas. However, we should pay more attention on some important facts of creativity. First of all, such a fact is the very possibility of a solution, insight, only if the key to it is already contained in an unconscious experience [9].

Thus, the accumulated experience is an essential determining element in the development of the final solution. Therefore, the prerequisite for strengthening the dominance of creativity in the lecture will be the condition for the accumulation of individual experience in the overall process of responding to the information offered by the lecturer. The second fact of the creative process that will be taken into account is comfortable conditions formed to understand the idea that is the time of solution maturing, which corresponds to the incubation stage in the Wallace model.

Regarding the effectiveness of lectures as a method to transmit information, it was argued in [13] that they are relatively ineffective for attracting interest in a subject, as well as for teaching behavioral skills. Learning Software Engineering should just be attributed to relevant practical skills, habits, techniques that are closely correlated primarily

with cognitive interest and interest in solving specific engineering and practical issues.

Currently, during lecturers' oral presentations of the material, students rarely use notes. Instead, they use smartphones or tablets to photograph graphically expressed slides of lecture presentations or notes on the information board.

It is noted that the success to perceive knowledge in Software Engineering - and this is primarily due to the acquisition of the ability to write program codes - is largely determined by manual work: taking notes, writing specific tasks, writing or entering program codes into the editor. Therefore, the search to improve effectiveness of lectures should be carried out in the direction of introducing mandatory manual activities for students during lectures. As noted in [13]: "by working with didactic material manually, students create favorable conditions for the comprehensive perception of different qualities of objects, and one or another of their combinations allows children to find certain relationships and make necessary conclusions about it. Scientists and inventors also know the expression - "think with your hands", which any truly creative work requires [14-16]. Paying special attention to creativity, we cannot but quote G. Leibniz's words about the connection between the ease of finger action and the freedom of imagination: "just like playing harpsichord, you need your fingers get the habit of moving by themselves, and in order to create a beautiful aria, to compose a beautiful poem, to imagine the details of architectural decoration or to sketch mentally a picturesque image, it is required that our imagination is accustomed to independence, after which we can allow it freely to give up to its flight, without consulting with mind and as if in a state of delight" [17]. As for this effect of imagination release, we should also expect the liberation of thoughts. The consciousness being free of any obscure and unusual characters, and gradually getting used to them, will make at some point, as can be expected, a sort of a leap towards a certain idea of independent implementation of a specific algorithmic step. This moment will mean the initialization of the individual educational process. At this time, due to an emerging or improving skill, a mental effort begins to lessen at the same time [18].

The most functionally intensive way, from a practical point of view, to implement manual work for students who listen to lectures in Software Engineering is a dictation. Its implementation can serve not only as the formation and strengthening of basic skills, but also as a tool to manage the individual educational process, its pace, content and purposefulness.

In general, a dictation is defined as a type of teaching the correct writing to consolidate skills [19,20]. As far as the important point in discipline of Software Engineering is the skill, so we will define a dictation as the student's recording of a program code offered by the lecturer in form of a practical task to be implemented with writing.

When object-oriented programming in C ++ is taught, the dictation task can be as follows:

- Create two classes - basic and derivative - by connecting them polymorphically in a hierarchy.
- The basic class contains a dynamic data member of the pointer type on ‘float’.
- Within classes, implement three main types of constructors and an assignment operator, while the constructors must initialize accordingly the dynamic data member of the basic class with a numerical value.
- In addition, in classes, implement a function that demonstrates the behavioral difference of objects of created types, for example, ‘print’, which can type-out the name of an object type to the console.
- In the program main function – ‘main’, - define the STL (Standard Template Library) - a container of type std::vector<T> or std::list<T> with a type of template T that allows for storing together objects of any type of the created polymorphic hierarchy.
- Further, in the program main function, create two objects of user types and place them in the general storage of a selected container type.
- Below, show the behavior difference of the objects contained in the repository by calling the ‘print’ function for each item in the repository.
- In conclusion of the main function, as well as, if necessary, in the content of the classes, add the program code with missing elements for the program to function properly.
- Such formulation of a dictation task contains several important details for teaching Software Engineering:
- The skill of writing the basic elements of Object-Oriented Programming in C ++ is formed.
- The emphasis is on the use of a dynamic memory as the main resource of the program.
- The attention is focused on handling the data located in the memory, including understanding the need and the content of the class destructor.
- An increase is expected in the level of responsibility for software implementation in terms of reliability of the program execution, primarily the correct memory allocation and deallocation.

III. RESEARCH RESULTS

Such formulation of a dictation task contains several important details for teaching Software Engineering:

- The skill of writing the basic elements of Object-Oriented Programming in C ++ is formed.
- The emphasis is on the use of a dynamic memory as the main resource of the program.

- The attention is focused on handling the data located in the memory, including understanding the need and the content of the class destructor.
- An increase is expected in the level of responsibility for software implementation in terms of reliability of the program execution, primarily the correct memory allocation and deallocation.

IV. DISCUSSION

Along with the straightforward matter of developing programming skills through a lecture dictation, the issue of reducing (or completely eliminating) the students' manual activity is also solved - now manual work becomes mandatory, which steadily leads to an increase in the student's self-discipline and organization level and, consequently, the formation and strengthening of the most important quality of an active person - responsibility for his own actions.

The general discipline effect accompanying the introduction of a dictation in lectures should also be noted, which is expressed in better class attendance and in addition, better results during control events. Finally, a lecture dictation is aimed to provoke an interest in student's mind by intensive manual work.

Furthermore, the coordinating role of a lecture dictation is needed to be emphasized separately, when passive and therefore largely irresponsible students' sitting at lectures is replaced by active, creatively interested behavior. In other words, a dictation introduced at a lecture gives a clear and distinct meaning to the lecture itself, and it specifies in many aspects the abstract discipline of Software Engineering by providing it with the status of practical necessity and significance.

A serious point is the fact that a dictation is able to most accurately direct or focus the student's activity on the right path of development of their ability to think algorithmically using a variety of programming paradigms from structural, functional or procedural to object-oriented one. At the same time, choosing a specific task for a dictation (including the variety of its conditions), a dictation pace and intonation, and focusing on specific details of the program implementation, it is possible to develop specific skills and techniques in the development of specialized software systems - from databases, network applications or computer graphics to systems of parallel architectures and real time.

V. CONCLUSIONS

Having the intensive manual training during weekly classes by the end of the second month of study, the student will, gradually becoming absorbed in the circle of ideas and ways to solve programming issues, be able to work systematically and independently in terms of both researching programming techniques and creating his own software solutions. In other words, by the end of the semester, the student will be able to cope with a variety of practically significant programming tasks, and the knowledge acquired by him will be needed in science, industry and economic activity

where Software Engineering plays an increasingly significant and important role in a digital era.

References

- [1] C. M. Chen, & C. H. Wu, "Effects of different video lecture types on sustained attention, emotion, cognitive load, and learning performance". Computers & Education, 80, pp. 108-121, 2015.
- [2] N. T. T. Thai, B. De Wever and M. Valcke. "The impact of a flipped classroom design on learning performance in higher education: Looking for the best "blend" of lectures and guiding questions with feedback." Computers & Education 107, pp. 113-126, 2017.
- [3] F. V. O'Callaghan, D. L. Neumann, L. Jones & P. A. Creed, "The use of lecture recordings in higher education: A review of institutional, student, and lecturer issues". Education and Information Technologies, 22(1), pp. 399-415, 2017.
- [4] N. Bos, C. Groeneveld, J. Van Bruggen & S. Brand-Gruwel, "The use of recorded lectures in education and the impact on lecture attendance and exam performance" British Journal of Educational Technology, 47(5), pp. 906-917, 2016.
- [5] K. Ludvigsen, R. Krumsvik & B. Furnes. "Creating formative feedback spaces in large lectures" Computers & Education, 88, pp. 48-63, 2015.
- [6] R. K. Pettit, L. McCoy, M. Kinney & F. N. Schwartz, "Student perceptions of gamified audience response system interactions in large group lectures and via lecture capture technology" BMC medical education, 15(1), a.n. 92, 2015.
- [7] H. Georgiou & M. D. Sharma. "Does using active learning in thermodynamics lectures improve students' conceptual understanding and learning experiences?" European Journal of Physics, 36(1), a.n. 015020, 2014.
- [8] E. C. Bolden, T. M. Oestreich, M. J., Kenney & Jr, B. T. Yuhnke. "Location, location, location: A comparison of student experience in a lecture hall to a small classroom using similar techniques". Active Learning in Higher Education, 20 (2) pp. 139-152, 2017.
- [9] Y. A. Ponomarev, Psychology of creativity. Moscow: Science, 1976.
- [10] D. V. Ushakov, "Personality traits and intellectual structures", INTERNATIONAL JOURNAL OF PSYCHOLOGY, Vol. 31, No. 3-4, pp. 54116-54116, 1996.
- [11] G. Wallas, The Art of Thought, New York: Harcourt, Brace, and Company, 1926.
- [12] J. P. Guilford, Creativity. American psychology. N.Y., 1950.
- [13] V. V. Davydov, "The Mental Development of Younger Schoolchildren in the Process of Learning Activity" Soviet Education 30.9 pp. 48-83, 1988.
- [14] F. H. Cushing, "Manual concepts: A study of the influence of hand-usage on culture-growth" American Anthropologist 5.4: pp. 289-318, 1892.
- [15] M. W. Thring, E. R. Laithwaite. How to invent. London: Macmillan, 1977.
- [16] M., I. Alba-Dorado "Learning to think with hands. Creative strategies in Architecture learning" IV WORKSHOP ON EDUCATIONAL INNOVATION IN ARCHITECTURE (JIDA'16), pp. 1-12, 2016
- [17] G. W. Leibniz, "The monadology" In Philosophical papers and letters, pp. 643-653. Springer, Dordrecht, 1989.
- [18] D. A. Norman, Learning and memory. WH Freeman & Co., 1982.
- [19] P. Davis, M. Rinvolucri, Dictation.: New methods, new possibilities. Ernst Klett Sprachen. 1988.
- [20] S. Kazazoğlu, "Dictation as a language learning tool". Procedia-Social and Behavioral Sciences, 70, pp. 1338-1346, 2013.
- [21] T. H. Ribot, Essai sur l'imagination créatrice. Paris, 1900
- [22] P.K. Engelmeier The theory of creativity, S-Pb, 1910