Brain-based Programming continued

Effective Teaching In Programming Courses

Barbara Sabitzer
Department of Informatics Didactics
Alpen-Adria-Universität Klagenfurt
Klagenfurt, Austria
barbara.sabitzer@aau.at

Stefan Pasterk
Department of Informatics Didactics
Alpen-Adria-Universität Klagenfurt
Klagenfurt, Austria
stefan.pasterk@aau.at

Abstract— **After the promising results of the pilot phase of "Brain-based Programming", a new teaching concept for introductory programming courses based on neurodidactical principles, the empirical study continues this year in three groups. The results obtained so far in the current semester at our university are promising, too: In the first exams the students of the experimental groups achieved significantly better than their colleagues in the traditional courses and female students seem to benefit even more of the new concept. This paper describes the basics of the teaching concept "Brain-based Programming" and reports on the empirical results regarding the positive impact this concept on the learning outcomes (Cohen's d = 0.42) as well as the students' and teachers' feedback.**

Keywords—brain-based; neurodidactics; programming; computer science education;

I. INTRODUCTION

For a lot of students learning programming seems to be a hard part of computer science education. Therefore the introductory programming course called "Introduction to structured and object-based programming", shortly ESOP, represents a big hurdle for them. In order to make this hurdle a bit smaller, the project "Brain-based Programming" was initiated in 2012. During this project we developed a new teaching concept and an additional self-learning booklet based on neurodidactical principles [1].

As Neurodidactics (Neuroscience + Didactics) is a relatively young interdisciplinary science it is still not very established in current research. The term is mainly used in German speaking countries whereas the Anglophone world talks about educational neuroscience, neuropedagogy, brain-based learning, or Mind, Brain & Education Science (MBE). Fig. 1 shows how MBE is embedded in and interwoven with other disciplines. But, one essential field is missing: Didactics including also subject specific (e.g. computer science) or competence related didactics, which is relevant in several subjects (e.g. problem-solving). One aim of our approach "brain-based programming" was to integrate neurodidactics in practice, exactly in computer science and to support understanding and learning of complex topics and competences.
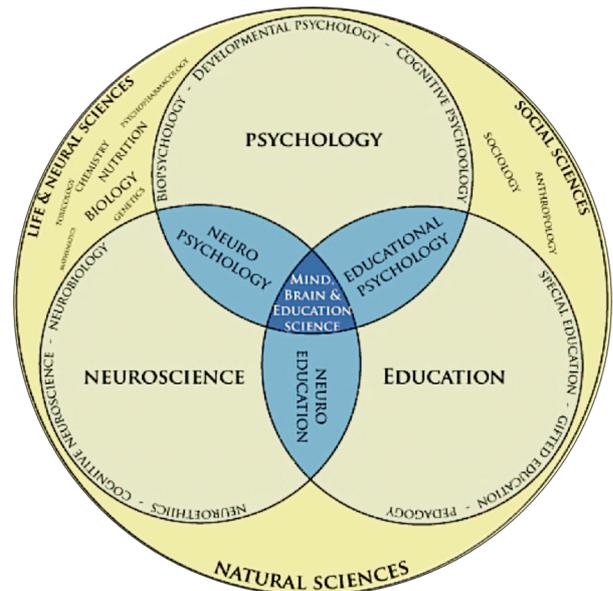
Fig. 1. Mind, Brain & Education Science [2], adapted by the author.

Certainly, neuroscience cannot be transferred to the classroom one to one and this is not the aim of neurodidactics. It rather wants to inform about the learning brain and to give proposals for effective learning and teaching, considering not only neuroscience but also findings of educational science, progressive pedagogy, didactics and psychology. Literature in this field including educational neuroscience and brain-based learning is rather theoretical and gives advices for practice and general didactics [3, 4, 5, 6, 7, 8, 9, 10]. But, only very few projects have studied the impact of neurodidactical approaches in the classroom like [11]. They found a significant impact of a concept based on Caine's main principles of "brain-based learning" in two Science classes of a Turkish high school.

Neurodidactics is related to various research disciplines and therefore integrates several theories, concepts and methods. Most of them are already studied as single methods in different contexts, but not as combination or in relation to neurodidactics respectively to the neurobiological basis. Effective teaching methods that satisfy neurodidactical principles are the following summarized e.g. in [12]: learning by doing or active learning [12, 13], questioning [12, 14], cooperative learning [12,

15, 16], small group learning, peer tutoring [12, 17, 18], assisted discovery learning [19] and the use of worked examples [12, 20, 21]. In the context of computer science methods like peer tutoring [23], cognitive apprenticeship [24], pair programming [25, 26] and discovery learning [27] have been evaluated as effective.

All these studies concentrate on single teaching methods. But, "the whole is more than the sum of its parts". This statement of Aristotle (384–322 B.C.E.) has become a principle of neurodidactics, too. Hence, we developed and implemented a complete teaching concept that considers and supports the brain functioning as well as the whole learning and memory process on different levels: in task design, lesson structure, and teaching methods. In the context of computer science this neurodidactical approach is new and was firstly tested in the pilot project "Brain-based Programming". The promising evaluation results where already presented at the Frontiers in Education conference in 2013.

This paper reports on the follow-up study in the winter term 2013/14, which confirms the success of the pilot project. The main research questions were:

1. Can "Brain-based Programming" increase the learning outcomes in the bachelor programming courses?

2. Do students and teachers appreciate the whole concept and teaching materials)?

After describing the concept of "Brain-based Programming" as well as the extended approach "Brain-based Teaching" this paper presents the results of the second experiment in 2013/14.

II. "BRAIN-BASED PROGRAMMING"

A. The concept "Brain-based Programming"

"Brain-based Programming" [1] is a new teaching concept for introductory programming courses at university level. It has been developed during a pilot project of the same name and implemented as a didactical experiment in one of seven parallel practical programming courses (bachelor) at our university, which accompany a lecture of weekly 90 minutes. The aim of the project was to improve the learning outcomes and to teach programming considering neurodidactical principles. These principles were considered in the lesson design as well as in the teaching methods, worksheets and tasks. [1]

The most important principles are the following:

1. Knowledge cannot be transferred. It has to be generated in each student's brain [29].

2. Learning through imitating [28].

3. The brain recognizes and produces patterns, categories and rules itself [6].

4. New content is always built on existing knowledge and learning occurs through associating [6, 28].

5. Learning is more effective when it makes sense and has meaning [28].

6. The brain needs time for consolidation [28, 30].

7. The instruction method has impact on the retention of new information [28].

8. Double coded is double saved (multimedia effect) [31].

During the units students should be active and have more possibilities according to their individual preconditions and competencies. To reach this, the students were divided in three groups: professionals with already high programming competencies who worked as peer-tutors, amateurs with some competencies and real beginners [1].

The students had the possibility to choose between different types of tasks, topics and collaboration according to their knowledge, needs and interests. The weekly lessons of 90 minutes were organized as open learning units, where students follow their own learning rhythm. Three phases describe the sequence of the lessons:

1. Questions (10 to 15 minutes): During this time the students worked in small groups with one peer-tutor in each group. They were free to ask any question they had in mind.

2. Discovering (10 to 20 minutes): In the second phase the students remained in the same small groups and worked on topics they had learned in the parallel lecture. Reading exercises, step-by-step instructions, exercises including solutions and short video clips should support the students to (re-)discover these topics.

3. Pair Programming (rest of time): Students should solve different tasks of given worksheets by collaborating in pairs. [1]

This year the new concept of brain-based programming was tested by three different teachers, who got an introduction and could use the material developed last year. The four control groups followed the traditional course process with only two settings in turns: one week the students did a part of the exercises that they continued at home (laboratory), and the following week one of the students presented his/her solutions of the exercises. During the presentation the other students were passive and, if necessary tried to correct their own program code. When they had questions, they asked them in the plenary session.

B. From Brain-based Programming to Brain-based Teaching

Based on the principles summarized above and the results of the pilot project in 2012 (observations and empirical results) in programming courses, a more flexible concept for brain-based teaching was developed. Firstly presented in [32] as "COOL Informatics" this model represents a basis for courses using neurodidactical knowledge to improve learning in different subjects.

The four main principles are *Discovery*, *Cooperation*, *Individualization* and *Activity* (Table 1, 1st column). Each of them is related to effective teaching and learning methods (Table 1, 2nd column), in part already empirically proven, and a neurodidactical basis, which builds the basic elements for the principles. The following table gives an overview of the concept:

TABLE 1: BRAIN-BASED TEACHING CONCEPT [32]

	Neurodidactical principles	
	Teaching and learning methods	*Neurodidactical basis*
1. Discovery	Solution-based learning Observational learning Step-by-step instructions and tasks Video tutorials Hands-on, Minds-on Learning with all senses	Pattern recognition Mirror neurons Individual learning rhythm modality / multimedia effect
2. Cooperation	Team and group work Peer tutoring and peer teaching, Pair programming Cross-curricular learning Project-based learning	"A joy (=knowledge) shared is a joy (=knowledge) doubled." Recall = re-storage in long-term memory Integrating individual needs, talents and competences as well as practical relevance
3. Individuality	Competence-based learning Questioning, Self-organized learning with compulsory and optional tasks	Connecting new information to previous knowledge, Considering individual interests, needs, tasks, methods and learning rhythm
4. Activity	Hands-on, Minds-on Learning by doing Learning by animation and simulation Learning by playing and designing games (creative learning)	Knowledge must be newly created (constructed) by each learner (= constructivism) Learning is an active process (=progressive education)

The teaching methods used in this concept are not new, in part well known. They consider different neurodidactical principles that are indicated in the right column. This table is not complete, but reflects the most important methods and principles. The neurodidactical basis cannot be matched one-to-one to a certain method. Some methods satisfy different neurodidactical principles and each principle is relevant for different methods.

III. METHODS AND RESULTS

"Brain-based Programming" was firstly implemented in 2012/13 as an experiment in one introductory programming course (held by one of the authors): "Introduction to structured and object-based programming". The pilot project started in winter 2012 with one "brain-based" experimental (21 students) and six parallel control groups (105 students) with traditional setting (only laboratory and presentation in turns). In this pilot project the average grades of the students were compared to the six parallel traditional groups as well as to the average of the last years in order to evaluate the impact of the concept on the learning outcomes. Furthermore, three questionnaires and the official course feedback revealed the acceptance among the students as well as their satisfaction with the concept, the used teaching methods and materials.

Results of the pilot project

The results of the pilot project, published in [1] were promising: the average grade of the students in the brain-based programming course (2.19; in Austria 1 = excellent, 5 = failed) was considerably better than in the parallel groups (2.94) and in the last three years (3.1) [1]. Due to this small group and a

certain teacher bias the positive results of this pilot project concerning the learning outcomes are not statistically relevant, but support the hypothesis that learning can be more effective when brain and memory functions are considered in lesson and task design.

Regarding the students' evaluation of the concept and the tasks in the verbal feedback the concepts of peer-tutoring and peer-teaching, reading exercises, discovery learning, mini exercises including solutions, step-by-step instructions, and the revision worksheet at the end of the course are most notably and helpful [1]. Taking a closer look at the official feedback with grades on a scale from 1 (best) to 5 (worst), it can be said that the experiment was successful. The "brain-based" course was graded with 1.1, whereas the average grading of all equivalent courses in the last three years lies at 2.1.

Results of the second phase

In the winter term 2013/14 the experiment was repeated in three experimental groups (71 students) and four control groups (81 students). The following table shows the distribution of the different groups:

TABLE 2: NUMBER OF STUDENTS IN CONTROL (CG) AND EXPIREMENTAL GROUPS (EG)

	Number of students				Number of students		
Groups	*female*	*male*	*total*	*Groups*	*female*	*male*	*total*
CG1	11	6	17	*EG1*	8	15	23
CG2	7	18	25	*EG2*	8	13	21
CG3	10	16	26	*EG3*	8	19	27
CG4	9	11	20				
total CG	37	51	88	*total EG*	24	47	71

Like in the pilot project, the learning outcomes where tested in a mid-term and a final exam. The average points were compared in both exams (a maximum of 25 points could be achieved in each exam). We checked the significance in a t-test for independent groups and calculated the effect size Cohen's d in order to measure the impact of the teaching concept on the learning success.

In a questionnaire at the beginning of the semester we collected personal data and previous competences in computer science and particularly in programming (about 50% of the students in each course had already some programming competences) in order to find out, which students could work as peer tutors. After the second exam we evaluated the acceptance and satisfaction with the concepts, certain teaching methods and materials in a further questionnaire.

The results in 2013/14 confirmed those of the pilot project and show a positive overall impact of the brain-based teaching concept on the learning outcomes. The following advantages were found:

• The success rate in the three "brain-based" groups was higher (52%) than in the parallel four control groups (40%) and the traditional groups in the preceding years (usually 30% - 40%).

• In the first exam the students in the brain-based groups achieved in average significantly more points (comparison, t-test for independent groups, p = 0.008, Cohen's d = 0.42).
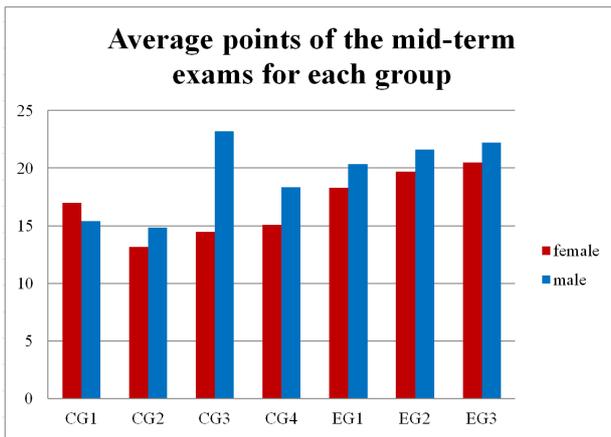
Fig. 2 The average points of all groups from the mid-term

This significant difference affects the overall results of the course, too.
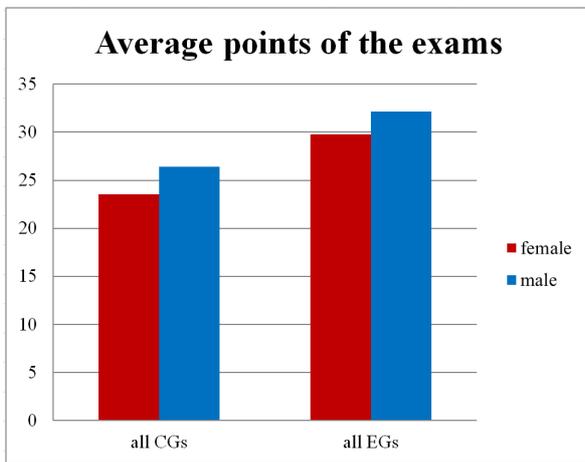
Fig. 3 Total average points (both exams, max. 50, under 25 = failed)

Female students could benefit even more from the brain-based teaching concept: Whereas females achieved significantly less points in the control groups, this gender gap could not be observed in the experimental groups.

TABLE 3: GENDER DIFFERENCES IN CONTROL GROUPS (ACHIEVED POINTS)

Sex	N	Mean	SD	T	p	Cohen's d[1]
male	47	19,85	6,70	2,52	0,014	0,58
female	36	15,54	8,45			

[1] Cohen's d was calculated on
http://www.soerenwallrodt.de/index.php

TABLE 4: GENDER DIFFERENCES IN EXPERIMENTAL GROUPS

Sex	N	Mean	SD	T	p	Cohen's d
male	46	21,85	5,47	1,25	0,214	-
female	25	19,72	5,14			

Although the average of the achieved points was higher in the second exam, too, the difference was not significant.

TABLE 5: AVERAGE OF ACHIEVED POINTS IN EXPERIMENTAL AND CONTROL GROUPS

Brain based	N	Mean	SD	T	p	Cohens d
yes	71	10,73	6,98	-1,545	0,124	-
no	88	8,82	8,64			

The evaluation of the questionnaires concerning acceptance and satisfaction with the teaching concept, methods and materials reveals that in both years the most useful elements are the cooperative learning forms (peer tutoring, pair programming) and the exercises supporting discovery learning (step-by-step instructions or video tutorials, sample solutions). Figure 4 shows the most useful methods and tasks. The methods and tasks were evaluated corresponding to the Austrian grading system from 1 (excellent) to 5 (failed).
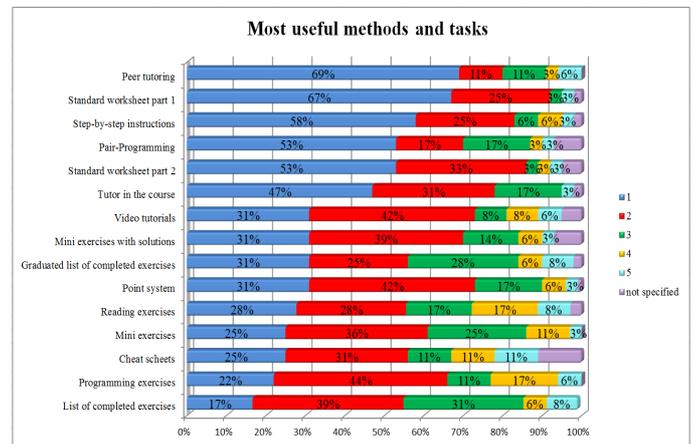
Fig. 4 Feedback from the mid-term evaluation, n = 37
Legend: 1 = very useful, 5 = not useful.

A SWOT-analysis for students, peer-tutors and teachers about the strengths and weaknesses of the concept should give a possibility of evaluation from another point of view. The results from 24 answers (18 students, 2 peer-tutors, 4 teachers) are similar to those from verbal feedbacks and interviews. From 18 students 14 mentioned the following course aspects as very positive: working in groups, free and self-organized learning and the good and relaxed atmosphere in the courses. All four teachers included these elements in the strengths of the concept.

Another strength, pointed out by eight students, is learning by teaching, including peer-tutors and the presentations held by students. Students seem to prefer asking other students for help. Again all four teachers see the positive effect of peer tutoring because students with prior knowledge are active and do not get bored of content they already know. Peer-tutors themselves also recommend this method because of a higher motivation

for learners with prior knowledge and an improved learning effect by trying to answer a lot of different questions. A positive aspect for teachers is the time they can dedicate to give individual help or to ask the students different questions in order to check the understanding and knowledge.

The additional exercises were very helpful for three teachers and most of the students. Some students determined the exercises as too many and too extensive for too little time. Here should be mentioned that some of them often did not recognize that these exercises were not meant to be solved additionally, but to have some freedom of choice in exercise type and content.

IV. DISCUSSION

Regarding the results of the pilot phase in 2012/13 and the follow-up project in 2013/14 both main research questions indicated in section *1 Introduction* can be answered by yes.

1. Applying "Brain-based Programming" in the bachelor programming courses "Introduction to structured and object-based programming" could significantly increase the learning outcomes and the success rate compared to the parallel control groups as well as to those of the three preceding years. This result should not be interpreted too optimistically, because teaching and learning is influenced by so many other factors that could not be controlled or considered in this study. This could e.g. the students' intelligence and their performance in other subjects, the personal preconditions and the level of previous knowledge and competences, the time spent for learning and exercises outside the course as well as the teacher bias or the attitude of the students towards the course. Further empirical studies are necessary in order to take more factors in consideration. Furthermore the gender effect has to be studied more in detail. In the brain-based groups the usual gender gap in programming did not exist. In the pilot project the females were even better than their male colleagues. This fact may be due to the way of problem solving: the cooperative learning forms and the possibility to solve problems by talking about them with different persons might be more appropriate for females.

2. Students and teachers appreciate the new teaching concept and teaching materials[2] as well, but with one limitation concerning the exercises: due to time restricts the teachers are not able to provide so much material in every course and to develop enough step-by-step exercises themselves. This may be solved by collecting and exchanging teaching material as well as by motivating the students to develop their own material as the author recommends in her own courses. The students appreciate it and the advanced learners are often glad to bring in their own ideas.

V. CONCLUSION

The evaluation of the second phase in "Brain-based Programming" confirms the positive results of the pilot project. This supports the hypothesis that teaching and learning can be improved by considering the brain functioning and supporting

the memory process. But, brain-based teaching and learning is not a panacea as some commercial publications may claim, but it can help students in learning difficult and complex subject matters like computer science. Many proposals of neuro-didactics are not really new but refer to well-known teaching concepts like constructivism and progressive education and teaching methods like cooperative and discovery learning or learning by teaching. "Brain-based Programming" is the first teaching approach that considers neurodidactical principles on different levels. Up to now the concept as a whole was tested and evaluated. In a follow-up project that is being planned now we want to test the adapted concept in schools and have a closer look at specific aspects like the use of pattern recognition in classroom or the impact of cooperative methods from the point of view of neurodidactics. Furthermore, there will be a closer look to gender aspects.

REFERENCES

[1] B. Sabitzer, S. Strutzmann, "Brain-based Programming" in Proceedings of *IEEE Frontiers in Education*, Oklahoma City, Oklahoma, US, October 2013.

[2] http://www.alanjoneseducation.com/wp-content/uploads/2011/03/Tokuhama2.1-300x300.png

[3] G. Roth, *Bildung braucht Persönlichkeit. Wie Lernen gelingt.* Klett-Cotta Verlag Stuttgart, 2011.

[4] U. Goswami, "Neuroscience and education: from research to practice?"In *Nature Reviews Neuroscience*, 7(5), 406–413, 2006.

[5] J. Hall, "*Neuroscience and education: A review of the contribution of brain science to teaching and learning.*" Scottish Council for Research in Education, 2005. Retrieved from: http://arrts.gtcni.org.uk/gtcni/bitstream/2428/12735/1/Neuroscience.pdf.

[6] M. Spitzer, Lernen. *Gehirn-Forschung und die Schule des Lebens.* Heidelberg, Spektrum-Verlag, Berlin,Germany, 2003.

[7] E. Jensen, *Brain-based learning: The new paradigm of teaching.* Thousand Oaks: Corwin Press, 2008.

[8] R.N. Caine, C. McClintic, *12 Brain/Mind Learning Principles in Action: Developing Executive Functions of the Human Brain.* SAGE Publications, 2009. Retrieved from http://books.google.at/books?id=4B9rPlvhiLoC.

[9] M. Gurian, K. Stevens, *Boys and Girls Learn Differently! A Guide for Teachers and Parents.* Wiley, San Francisco, USA, 2010.

[10] J. Geake, P. Cooper, "Cognitive Neuroscience: implications for education?" in *Westminster Studies in Education*, 26(1), 7–20, 2013.

[11] M. Ozden, M. Gultekin,"The Effects of Brain-Based Learning on Academic Achievement and Retention of Knowledge in Science Courses." in *Electronic Journal of Science Education*, 12(1),2008. Retrieved on 19/07/2013 from: http://ejse.southwestern.edu/article/download/7763/5530.

[12] J. Hattie, *Visible Learning: A Synthesis of Over 800 Meta-Analyses Relating to Achievement.* Taylor & Francis, London, New York, 2009.

[13] H.W. Reese, "The Learning-by-Doing Principle." In *BEHAVIORAL DEVELOPMENT BULLETIN*, 11, 2011.

[14] S.D. Craig, J. Sullins, A. Witherspoon, B. Gholson, "The deep-level-reasoning-question effect: The role of dialogue and deep-level-reasoning questions during vicarious learning." In *Cognition and Instruction*, 24(4), 565–591, 2006.

[15] C.J. Roseth, D.W. Johnson, R.T. Johnson, "Promoting early adolescents' achievement and peer relationships: The effects of cooperative, competitive, and individualistic goal structures." in *Psychological bulletin*, 134(2), 223, 2008.

[16] C.J. Roseth, F. Fang, D.W. Johnson, R.T. Johnson, "Effects of cooperative learning on middle school students: A meta-analysis." in San Francisco: *American Educational Research Association Convention*, 2006.

---

[2] Some sample material is available on http://brain-based-programming.jimdo.com/

[17] B. Gokkurt, S. Dundar, Y. Soylu, L. Akgun, "The Effects of Learning Together Technique Which is based on Cooperative Learning on Students' Achievement in Mathematics Class." in *Procedia-Social and Behavioral Sciences*, 46, 3431–3434, 2012.

[18] V. Miller, E. Oldfield, M. Bulmer, "Peer Assisted Study Sessions (PASS) in first year chemistry and statistics courses: insights and evaluations." in *Proceedings of The Australian Conference on Science and Mathematics Education* (formerly UniServe Science Conference) (Vol. 10), 2012.

[19] L. Alfieri, P.J. Brooks, N.J. Aldrich, H.R. Tenenbaum, "Does discovery-based instruction enhance learning?" in *Journal of Educational Psychology*, 103(1), 1, 2011.

[20] A. Renkl, R.K. Atkinson, "Learning from examples: Fostering self-explanations in computer-based learning environments." in *Interactive learning environments*, 10(2), 105–119, 2002.

[21] Atkinson, R. K., Derry, S. J., Renkl, A., & Wortham, D.. Learning from examples: Instructional principles from the worked examples research. *Review of Educational Research*, 70(2), 181–214, 2000.

[22] I. Kalas, M. Winczer, "Informatics as a contribution to the modern constructivist education." in *Informatics Education-Supporting Computational Thinking* (pp. 229–240). Springer, 2008.

[23] L. Porter, C. Bailey Lee, B. Simon, "Halving fail rates using peer instruction: a study of four computer science courses." in *Proceedings of the 44th ACM technical symposium on Computer science education* (pp. 177–182). ACM, 2013.

[24] A. Vihavainen, T. Vikberg, M. Luukkainen, J. Kurhila, "Massive increase in eager TAs: Experiences from extreme apprenticeship-based CS1." in *Proceedings of the 18th ACM conference on Innovation and technology in computer science education* (pp. 123–128). ACM, 2013.

[25] N. Salleh, E. Mendes, J. Grundy, "Empirical studies of pair programming for CS/SE teaching in higher education: A systematic literature review." in *Software Engineering, IEEE Transactions on*, 37(4), 509–525, 2011.

[26] C.M. Lewis, "Is pair programming more effective than other forms of collaboration for young students?" in *Computer Science Education*, 21(2), 105–134, 2011.

[27] D. Baldwin, "Discovery learning in computer science." in *ACM SIGCSE Bulletin* (Vol. 28, pp. 222–226). ACM, 1996.

[28] Sousa, D. A. 2006. *How The Brain Learns*. Third edition. Thousand Oaks, California: Corwin Press.

[29] G.Roth, „Warum sind Lehren und Lernen so schwierig?" in U. Herrmann (ed). *Neurodidaktik: Grundlagen und Vorschläge für gehirngerechtes Lehren und Lernen*. Weinheim, Basel: Beltz (German), 2009.

[30] Brand, M., & Markowitsch, H. J. „Lernen und Gedächtnis aus neurowissenschaftlicher Perspektive - Konsequenzen für die Gestaltung des Schulunterrichts." in U. Herrmann (ed). *Neurodidaktik: Grundlagen und Vorschläge für gehirngerechtes Lehren und Lernen* pp. 69-85. Weinheim, Basel: Beltz (German), 2009.

[31] Mayer, R. E. (2002). Multimedia learning. *Psychology of Learning and Motivation*, *41*, 85–139.

[32] Sabitzer, B. (2014, in press). *A Neurodidactical Approach to Cooperative and Cross-curricular Open Learning: COOL Informatics*. Habilitation thesis. Alpen-Adria-Universität Klagenfurt.