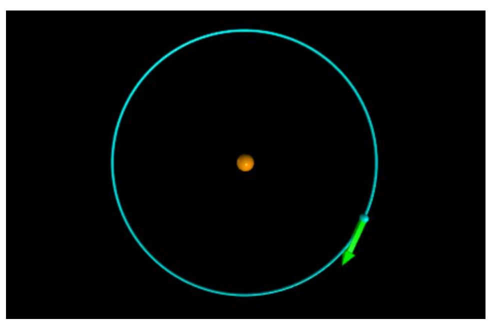
OSLO METROPOLITAN UNIVERSITY
STORBYUNIVERSITETET

Fakultet for lærerutdanning og internasjonale studier

OsloMet – storbyuniversitetet

Technology in Physics Class

What Is the Impact of Technology and the Implementation of Programming to a Physics Class in Secondary School?



Ciprian Sima

Master Thesis in Digital Learning Design

Masteroppgave i Digital læringsdesign

2019

Table of Contents

Abstract

Technology has entered the Physics class and it is being used to support gaining conceptual knowledge and skills necessary in today’s highly digital world. In this action research study, I investigate the effects Programming and Technology may have on a secondary school Physics class and implications. The impact can reflect in both student grades and student perception of class work and programming activities. Programming is found in national and international curricula. I identify methods of introducing programming and leveraging it within Physics classes in Secondary School. Technology and Computational Physics will be the main areas of investigation.

The paper emphasizes the current use of technology and its impact on teaching (from a teacher’s perspective) and learning (from a student perspective). The practical part is a research study with an initial goal of finding out if introducing Programming in Physics lessons will work or not.

During the school year 2018-2019 I changed the way in which I taught “mechanics” topics, to include Programming. The change took place only in one of the two parallel Pre-IB Physics (vg1) classes in my school. The other Pre-IB Physics class attended regular Physics lessons, and although it had the same content, it did not include any Programming. The Programming lessons were both followed by questionnaires, reflecting students’ participation in the lesson. The test at the end of the unit was identical for both groups. Findings show that although the median and the average score for the students that used Programming in their Physics classes is higher, the difference is not statistically significant. If examination scores show no measurable statistically significant difference, surveys reflect positive outcomes, enthusiasm and benefits of using Programming in Physics lessons.

The findings are shared with the management of the school and all teachers in the science department. I want to continue and explore ways to teach other topics in Physics with programming as a supportive tool. The impact would not be limited to the teaching of just one topic, but it is limited to a small class.

Acknowledgements

Many people have contributed to the completion of this paper and I would like to show my appreciation.

I would like to thank my professors at OsloMet – Oslo Metropolitan University, Norway, for their guidance and support and the opportunity they gave me to study and work for my master's degree.

I would like to thank students, parents, colleagues and management at Oslo International School. Students have been a motivation and an inspiration to me.

I would like to thank my family for being supportive in my studies and a blessing all the way.

List of figures

Acronyms and abbreviations

AAPT	– American Association of Physics Teachers
CCSE	– Centre for Computing in Science Education
ICT	– Information and communication technology
IBDP	– International Baccalaureate Diploma Programme
IBO	– International Baccalaureate Organization
IBDP	– International Baccalaureate (IB) Diploma programme (DP)
ICILS	– International Computer and Information Literacy Study
NSF	– National Science Foundation
NTNU	– Norwegian University of Science and Technology
OSLOMET	– Oslo Metropolitan University
STEM	– Science, technology, engineering and Mathematics
UIO	– University of Oslo
TIMSS	– Trends in International Mathematics and Science Study

Chapter 1 Setting the Scene

The purpose of this action research study is to see how technology and Programming can be embedded into Physics lessons and to understand their impact on approaches to teaching and learning experiences of the students.

1.1 National perspective on the use of ICT and Programming in Physics lessons

Based on results from "Trends in International Mathematics and Science Study" (TIMSS) from 2015, we can say that Norwegian students perform poorly in Physics/Chemistry compared to American students for example. There are no significant changes from 2003 to 2015, but there are positive trends. (Nilsen, Bergem, & Kaarstein, 2016)

In 2013 Norway participated in ICILS (International Computer and Information Literacy Study) and findings show that around 40 percent of students in 9th grade, that is Secondary School, responded that they had never used technology in Mathematics lessons and 35 percent of students answered that they had not used technology in Science lessons. Reasons for these poor results could be, as suggested in the report, due to the limited use of professional development opportunities by teachers when it comes to digital skills. (Throndsen et al., 2015)

Back from 2006 it is stated in the Norwegian curriculum for Primary and lower Secondary School the five basic skills students need to have. The students need to: be able to write, be able to calculate, be able to read, have oral skills and be able to use Information and Communication Technology (ICT), called digital skills. ICT is not a subject for Primary and Secondary Schools, but it is integrated in all the rest of the subjects. (Guðmundsdóttir & Hatlevik, 2013).

In November 2017 the framework was revised by the Ministry of Education and Research to facilitate the future development of the curriculum. (Kunnskapsdepartementet, 15.11.2017) The framework describes all the skills separately, and for digital skills the following areas are described: the use of and understanding of digital resources and equipment; finding and processing information from digital sources; developing and creating digital resources, communicating and interacting digitally and lastly exercising digital judgment.

(Kunnskapsdepartementet, 15.11.2017). If creation of digital resources is a skill that students should learn, consideration needs to be given to how it could be implemented.

Programming can be among the processes of creating digital resources, such as applications or simulations. Programming as a subject is also part of the current (2018-2019) measures in the digitization strategy in Norway. It became an elective in middle school (lower secondary school) and Modelling and Programming together became a subject in upper secondary education.

As of now the Norwegian school system is moving forward and starting in fall of 2020 there will be new curricula in schools. Digital skills remains one of the five basic skills that students should develop, but Programming and Algorithmic Thinking will become part of the Mathematics and Natural Science subjects. (The Norwegian Government, 26.06.2018)

Considering that digital skills, including Programming are to be implemented across different subjects, e.g. Physics and Mathematics, this thesis aims to be a contemporary guide for teachers, as well as a reflective process. Studies show that teachers have a decisive role in selecting the technical means and guiding students through the creative process. (Barak, 2005)

In this context, I want to present my action research study and find out if introducing Programming in Physics lessons would have any impact.

1.2 Research question

The topic of the thesis is “Technology in Physics class” and it answers the question “What Is the Impact of Technology and the Implementation of Programming to a Physics Class in Secondary School?” through action research. In the Physics classes I teach, technology is used by both students and I. We become proficient users, but we depend on others to create the software, simulations, animations or the websites that we are using. Through action research I will try to see how I may be able to introduce Programming in Physics lessons. To clarify the focus, I’ll be looking at the required digital skills that students and teachers should have and would need to develop in order to meet the needs in this technological era. After establishing a focus, I will be looking at the required tools that will allow me to implement Programming in my Physics class and emphasize the implementation of an action plan. In the end, I will evaluate the change it has produced in term of the teaching strategies. Learning outcomes will be discussed together with limitations and improvements. Finding will be shared with the teachers in the science department and the leadership of the school. Dissemination can lead to improvements and the beginning of another action research cycle.

The implementation of the new curriculum in Mathematics and Science that includes computational thinking will open the door to more action research projects. Interventions will need to be carried out by teachers in order to understand how learning could be improved in small or large classes (Mills & Butroyd, 2014). Because it has not been investigated before in Secondary Education nationally, there is a need to carry out research to find out the effects, challenges and possibilities of using technology and implementing Programming in Physics lessons. I have been teaching Physics in Secondary School both at Standard Level and at Higher Level for more than ten years and never included Programming in my lessons. I have taken Programming as a subject both at master level at OsloMet and as a degree from University of Oslo. I am using visual effects, animations and simulations done by others in my teaching practices. Therefore, the paper is of interest to me. The focuses is on both the "use of" and the "development of" digital resources that can be achieved through programming in a Physics class.

The overall goal is to see if Programming could be implemented into Physics lessons, understand the problems that could arise in the implementation process, as well as discuss the implications and actions for the future. In the paper the implementation of Programming in three different sessions during the school year 2018-2019 are analysed. Data collection techniques used in my action research would use multiple perspectives. Triangulation comes from using a combination of data generated through my teaching activity that is linked to data from other school settings. Research data comes from mixed methods using the material produced through qualitative and quantitative questionnaires, test marks that and course guides.

I am working in an international school in Norway, so for this reason I want to also look at how technology is used in Physics lessons internationally, including international schools that implement the International Baccalaureate Organization (IBO) programme.

1.3 International perspective on the use of ICT and Programming in Physics lessons

The International Baccalaureate Diploma Programme is implemented in 156 countries and it is designed for high school students, ages 16 to 19. The IB Physics curriculum requires students to "develop experimental and investigative scientific skills, including the use of current technologies" where students "should develop and apply 21st century communication skills in the study of Science". (IB, 2014b). Students are expected to use information communication technology in day to day activities as well as in practical activities when collecting data, analysing data, plotting graphs and writing essays. (IB, 2014a) Although the data can be collected from hands-on experiments, the use of other sources, such as simulations and databases, are also encouraged (IB, 2014a) The "use of" simulations and not the "development" of simulations is mentioned in the Physics course of the IB Diploma Programme, and this gap is what I have identified as a possible area of improvement.

The two different approaches of introducing Programming in schools, either as a standalone subject or integrated into other subjects, such as Mathematics or Science (for example Physics, Chemistry and Biology) have been considered by my school as well.

Schools across the world are updating their curricula and Computational Thinking is becoming an important element in the framework. Computational Thinking has a bigger output than Programming, but it includes it. (Curzon, Bell, Waite, & Dorling, 2019)

Studies that connect Programming and Physics, called computational Science, have been done at the university level, but with curriculum changes on the way, this can extend to other school settings. According to Vondracek (2007), research on Computational Physics is not usual in high schools. In universities on the other hand, "Computational Research and using simulated experiments is exploding in popularity, quality, and applications." (Vondracek, 2007)

1.4 Structure of the thesis

The paper is divided into four sections that include: (1) area of focus and the context followed by (2) related literature, (3) methodology of caring out the research and the implementation process and in the end, (4) findings, evaluation and conclusion.

In the introduction the paper looks at national and international approaches of using technology and creating digital resources in schools. The developing of the digital resources such as simulations and animations can be achieved through Programming and this is the tool that will be used in the Physics class as a creative and hopefully enjoyable problem-solving activity.

In the related literature section current contexts and relevant terminology are identified. The skills needed by teachers are considered from the European framework for a digital competent educator. Studies that combine Science, Technology, Engineering and Mathematics

(STEM) are also considered to be able to put this action research study in a global context and relate existing research with my own.

Action research is the methodology used in this thesis to create change of personal teaching practices. It seeks to understand the effects of introducing Programming and how it impacts the attitudes, experiences and learning in a small classroom setting. The practical process describes some possibilities of implementing technology and Programming in Physics lessons and is part of the scheme of work utilized. To verify the findings multiple perspectives will be considered to achieve triangulation. Programming will be implemented only in one of the two Pre-IB Physics classes the marks constitute one quantitative data that results from a quasi-experimental study. The analysis and evaluation section focus on the empirical outcome, considering limitations of the finding. Referring to the literature a comparison with other studies is done even if the context might be different.

The conclusion summarizes the findings and the importance of the paper and includes unresolved issues and relevant recommendation for Physics teachers, including practice researchers and school curriculum decision-makers.

Chapter 2 Related literature

Given there is a gap as noted in the first chapter regarding research done in national secondary schools on Programming in the Physics curriculum and trends on the curriculum regarding computational thinking, the literature review aims to focus on studies conducted in other educational contexts, like national and international undergraduate university degrees or international secondary school systems as well as trends and frameworks for the future.

In order to focus my action my action research and be informed on what studies exist relevant literature review needs to take place.

2.1 Literature search process

The search process includes:

- Selection of sources, both internationally and nationally
- Terminology used in the search list in both English and Norwegian
- Specific search options using the Boolean operators OR, AND.

The source selection criteria include:

1. Academic books, from 2010 onwards
2. National and International Journals, Articles and Newspapers from 2010 onwards.
3. Government reports, models, frameworks and standards that deal with digital skills, computational thinking and Programming

An overview of the search process is presented in the Figure 1 below.

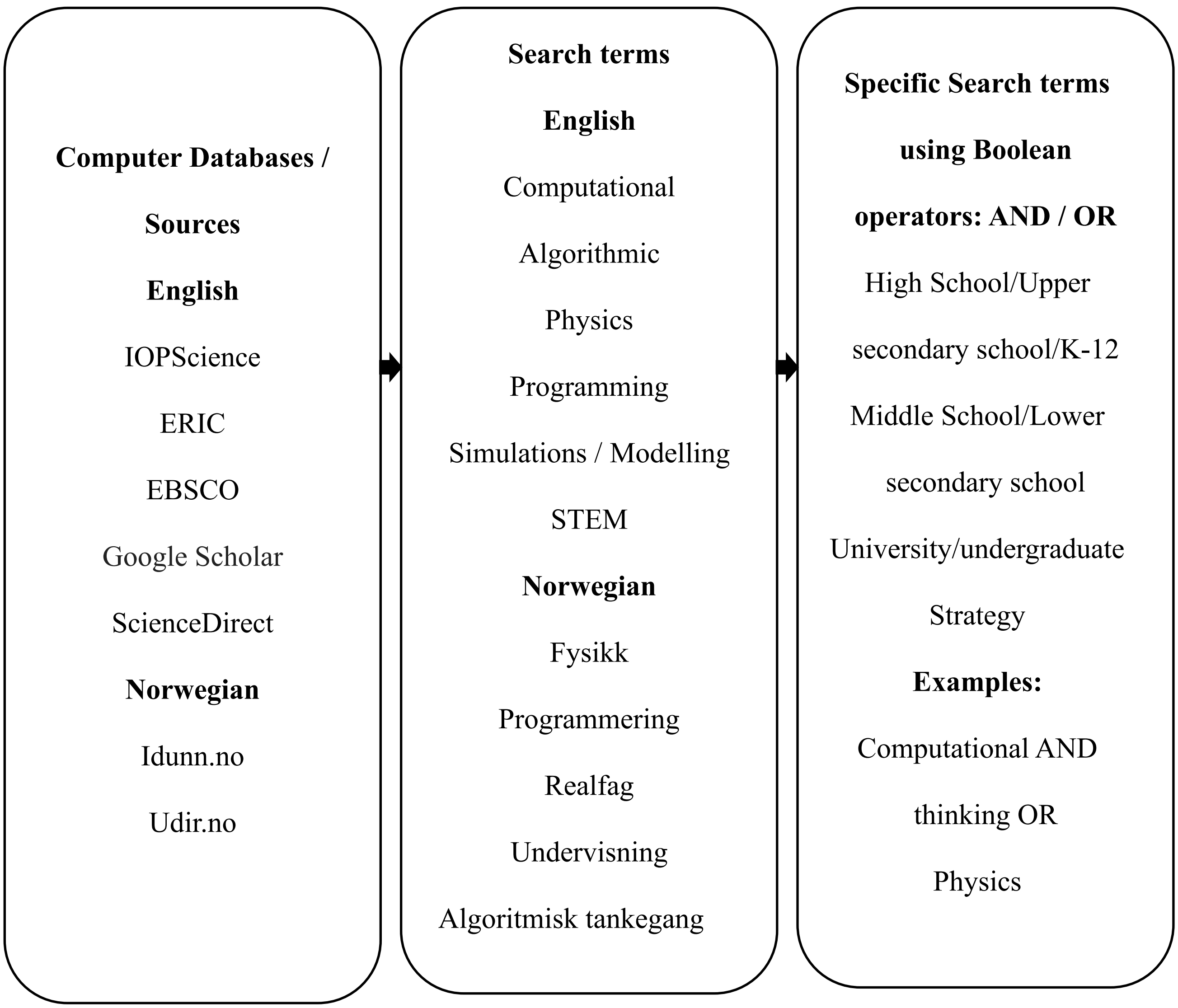


Figure 1 The search method

The initial search was on Programming in secondary school systems, standards and practices nationally and internationally. The report “Developing Computational Thinking in Compulsory Education” from the Joint Research Centre (European Commission) provided a clear picture of the approaches on integrating Technology into teaching and developing Computational Thinking in Europe and beyond. As I got familiar with the terminology used by different countries, I was able to narrow my search to studies that focused on Computational Thinking, Computational Physics and STEM (Science, Technology, Engineering and Mathematics) education.

The main three themes that can be identified are: (1) context that includes national and international education for secondary school and university, (2) skills needed and (3) models of implementing technology and Programming.

When looking at the standards and frameworks of introducing Technology and Programming in education the arena for implementation will be a Physics class with a special focus on the Physics curriculum.

2.2 Context

There is a recent trend everywhere in the world to include Computational Thinking in the curriculum. According to Norwegian Directorate for Education and Training “Algorithmic Thinking” is the Norwegian translation of “Computational Thinking “.(Training, 2019) An algorithm is a series of unambiguous instructions designed in order to solve a problem and achieve a certain goal in a finite number of steps.” (Dimitriou & Hatzitaskos, 2005) and according the Cambridge dictionary Programming is “the instructions that tell a computer what to do” and “the process or skill of writing programs for computers“. (Cambridge, 2019). Since Programming is a skill it can be learned. It includes designing, implementation and analysing computer programs. (Nuutila, Törmä, & Malmi, 2005)

An overview of the European status for different curricula in different European countries can be found in the report “Developing Computational Thinking in Compulsory Education” from the Joint Research Centre (European Commission). There are countries that have implemented Computational Thinking and Norway and another seven (CZ, GR, IE, NL, SE, UK-WLS) are currently planning to introduce Programming into compulsory education. The main reason for introducing Computational Thinking in most countries is to prepare students with the necessary 21st century skills to successfully face a digital world.

According to Erstad the situation in Norwegian schools is that the status of ICT has “been improving” and that ICT now creates a foundation that can be built upon in other classes, including Physics. (Erstad, 2007)

Other countries, as seen in Figure 2. have a long-standing tradition in Computer Science, mainly in the upper secondary school and those countries will also include Computational Thinking in lower secondary and primary levels. (Engelhardt et al., 2018)

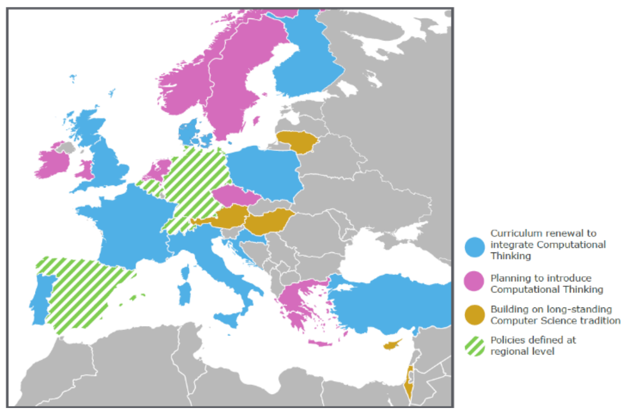


Figure 2 Integration of ICT in the school curricula in Europe (Engelhardt et al., 2018)

“A high-quality Computing Education equips students to use computational thinking and creativity to understand and change the world.” The slogan belongs to the UK Department of Education that since September 2014 has introduced Programming in primary and secondary schools. This started a chain reaction throughout Europe. (*National curriculum in England: computing programmes of study*, 2013)

A 2015 study shows that in Norwegian schools there is very little use of ICT in teaching and learning, and 35% of students have answered that they do not use ICT in Science classes. If lack of skills is the cause, measures that can be taken need to be addressed. (Hatlevik, Throndsen, Gudmundsdottir, & Olsen, 2015)

In Norway, the Centre of Computing in Science Education (CCSE) that is governed by the Department of Physics at the University of Oslo (UIO) has as goal “to include computing as a natural tool for all Science and engineering students from the first semester of their undergraduate studies”. ("CCSE - Center for Computing in Science Education," 2019)

In the US, formal programs in computational Physics are absent from most undergraduate Physics programs. (Caballero & Merner, 2018) If Programming is a skill than students should be taught. If it needs to be taught it must be implemented in the curriculum. But when should that skill be acquired? At international level research was conducted by the American Association of Physics Teachers (AAPT), including National Science Foundation (NSF) in "Survey the State and Implications of Computational Physics in Courses for Physics Majors" and "Integrating Computation into Undergraduate Physics – Building a Sustainable Community through Faculty Development" (NSF DUE 1505278), but those studies were done in an undergraduate university context, so that suggests that at university level Programming should be present in Physics Majors.

Recent efforts have been or are being made locally and internationally to incorporate Programming in Mathematics and Physics classes also in primary and secondary education, not only undergraduate studies.

2.3 Skills needed

To understand the needs of the school education system in terms of staffing and what are the attributes of a digital competent teacher we look at the European framework for the Digital Competence of Educators (DigCompEdu).

DigCompEdu is the European framework for the Digital Competence of Educators and is developed by the European commission. The concept of "Digital Competence" is defined and discussed both for the teacher and for the student. (Redecker & Punie, 2017) The model includes three areas of competences: Educators' professional competences, Educators' pedagogic competences and Learners competences and six areas of focus:

Area 1. Professional Engagement

Area 2. Digital Resources

Area 3. Teaching and Learning

Area 4. Assessment

Area 5. Empowering Learners

Area 6. Facilitating Learner's Digital Competence.

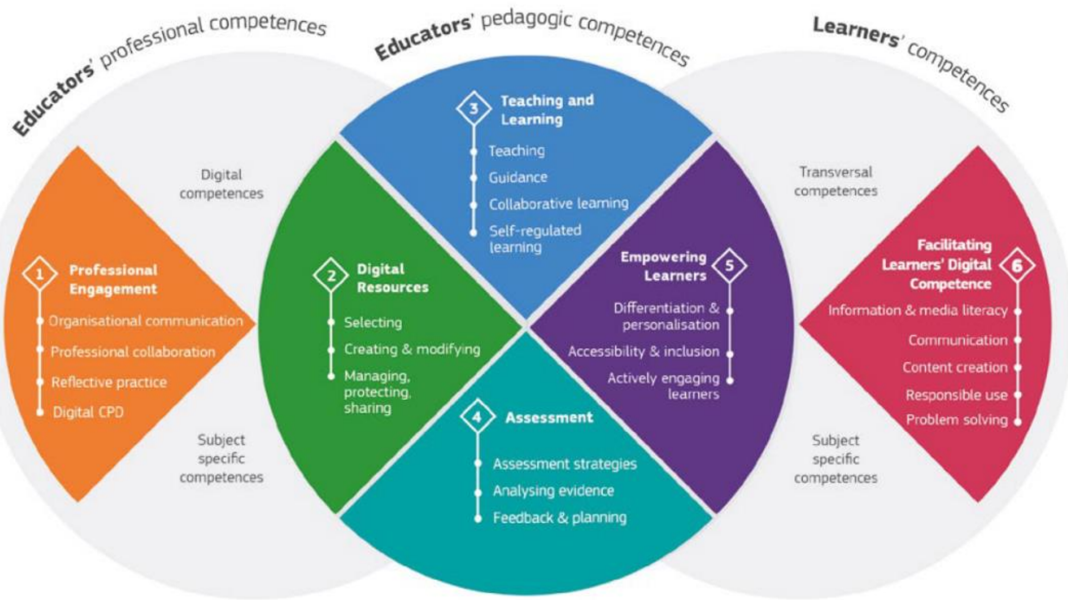


Figure 3 DigCompEdu areas and scope (Redecker & Punie, 2017)

As seen from Figure 3, educators that excel in Area 2 will select, create and modify digital resources to suit the learning objectives of the lesson. Teaching (Area 3) should develop new strategies that facilitate learner's digital competence (Area 6).

2.4 Approaches to integrating technology and Programming in Science

I work in an international school myself and I am familiar with the IB DP curriculum documents that are specific for Physics. In order to demonstrate excellent performance students should have a thorough knowledge and understanding of concepts in Science, form explanations to a variety of phenomena, demonstrate problem solving skills, demonstrate the ability to collect and analyse data, draw meaningful conclusions and use technology proficiently. (IB, 2014a)

2.4.1 Acquisition of data in Physics using technology

Hands-on learning is what gives students the opportunity to experience Science education. (Urban & Falvo, 2016) Experiments can take advantage of digital devices for efficient collection of measured data. For large amounts of data, as well as complex measurements, data loggers are recommended, since they are small handheld devices that get the data from various sensors. There are many different suppliers (Pasco, Vernier, DataLogger) on the market today that provide data logging equipment for Science education, as well as supportive manuals and materials. In Physics, dataloggers can be used in many lessons and the data collected is precise and accurate. In addition mobile phones with different sensors and applications can also be used as dataloggers. (González et al., 2014)

The data in Experimental Physics can be collected from hands-on experiments, but other sources such as simulations and databases are also encouraged. (IB, 2014b) Although Programming is not mentioned in the Physics curriculum, it is through Programming that those simulations have been created in the first place. (Chami, 2006) Simulations used as animations have a visual impact on understanding and involve a creative part. Roger D. Smith, in his article "Simulation: the engine behind the virtual world", defines simulation as "the process of designing a model of a real or imagined system and conducting experiments with that model". (Smith, 1999) Simulations can be used by both teachers and students to explore Physics ideas and demonstrate complex concepts encountered in real-life situations. Simulations can be used to generate data that can be organized in databases that are available for students and teachers. Programming can make the connection between real world and the concepts studied in school through means of visualization

In many simulations a wide range of variables can be manipulated and are suitable for data collection and that can be individually processed by students. There are studies done in secondary school that emphasize the characteristics and the positive effects of using simulations in online Physics lessons. (Rosenberg & Lawson, 2019) If properly used, simulations used for modelling or animations have the potential of increasing the understanding of Science. In cases where hands-on experiments are not possible due to a lack of equipment or other restrictions, such as health risks (in using radioactive samples in atomic and nuclear Physics for example), the use of simulations, animations and modelling is welcomed. (Rosenberg & Lawson, 2019)

Some popular digital platforms that can be used for Physics simulations that I use myself in my Physics class can be found at:

- PhET (phet.colorado.edu)
- Physics: The Physics Classroom (www.Physicsclassroom.com)
- Interactive Physics
- Yenka Education (https://www.yenka.com/Science/)
- Gizoms (explorelearning.com)
- SageModeler (https://sagemodeler.concord.org/index.html)
- https://www.myPhysicslab.com/

If simulations already exist, the reasons for learning Programming could be to modify and create content that meets specific needs and is an enjoyable problem solving activity. (Morelli & Walde, 2005)

2.4.2 Towards computational Physics

In Science classes (Physics, Biology and Chemistry) the purpose of Programming languages such as Python, Java, JavaScript, C++ is to solve a specific task. The simulations created can collect, generate, manipulate and display data. University studies conclude that Programming in Physics provides support for the conceptual understanding of Physics (Chabay & Sherwood, 2008).

In the past it was believed that entrance into the technological era would require that everyone engages with Programming. While this proved not to be true there are still reasons to study Programming. Programming is a creative and enjoyable problem-solving activity. (Morelli & Walde, 2005) In university on the other hand majority studies will argue that computational Science should be present in Science education (e.g. Physics, Biology). (Taub, Armoni, Bagno, & Ben-Ari, 2015) Should it not begin before university?

There are many Programming languages out there such as Java, Python, C, C++ and the choice depends mainly on the desired outcome. Python is considered the number one

Programming language in 2018 based on IEEE spectrum ranking (Spectrum, 2018) it's well suited for Science at graduate students (Stevens et al., 2018).

Considering the context, gaps and possible outcomes I want to carry a small action research project with my own Physics class. The research question discussed in the paper is "What is the Impact of Technology and the Implementation of Programming to a Physics Class in Secondary School?". The action research study is carried out in a secondary school, where there is a need for such research. In addition, little has been done to prepare secondary Physics teachers or primary general Science teachers for teaching Programming (AAPT). Considering action research tries to find out if something will work or not (Mills & Butroyd, 2014) this thesis based on the readings and my own praxis will look into how Programming can be introduced. Implementing Python Programming in high school Physics class is a pilot praxis that has the possibility to work towards bridging the gap between new and emerging high school curricula and university computational Physics courses.

Chapter 3 Methodology of caring out the research and the implementation process

My original question in my own action research project was "How can Programming be implemented in Physics lesson to upper secondary school students?". I was thinking to see if the intervention will work or not. My first Physics lesson that embedded Programming showed that students are able to follow instructions and are more interested how Programming will affect their understanding of Physics. My question changed to "What is the Impact of Technology and the Implementation of Programming to a Physics Class in Secondary School?" and positive outcomes of introducing Programming were considered.

3.1 Aims

I focused on implementing Programming to help students understand physical concepts through creation of digital content.

More specifically my aims were:

- To change the students from uses towards creators of digital resources

- To explore possibilities of improving my own practice by using Programming in a Physics class

- To improve experiences of the students in a Physics class and increase student interest

- To disseminate my research findings with colleagues and look for new opportunities or beginnings of new action research cycles

3.2 Collaborators

For the school year 2018-2019 there were 27 students taking Physics in Pre-IB (equivalent of the Norwegian Vg.1). The students were divided into two groups at the beginning of the school year. The students who were in school a year before (10^{th} grade) were evenly distributed among the two classes and new students were also placed in both classes, so that their previous performance and background knowledge were similar.

The implementation of the first two Programming sessions was done in one of the two classes, where 13 students were enrolled. Students of both classes were interested in Physics since it was an elective subject out of the three Sciences offered: Chemistry, Biology and Physics. Programming was not taught as a subject and was used as a support to help students understand Physical concepts that could be visually represented. In the last Programming lesson all 27 students participated.

3.3 Ethical considerations

Verbal consent was received from the Management Team of the school. The Science and the Mathematics teams of teachers were informed about the research before it began. Consent forms that included information about the research was given out to students who took part in the first two Programming lessons. Students were 15 or 16-year-old and written consent was received from all students and parents. Students participation will also include:

1. Completion of questionnaire
2. Group discussions
3. Giving and receiving feedback

Data collected during the research will be kept securely and then destroyed when the research project is complete. Anonymity is achieved at the reporting stage as students will not be named, personal data such as name, age and gender are not collected, and no person can identify itself. The identity of the students is known by the researcher but not to others and so both anonymity and confidentiality will be provided in this research.

3.4 Data collection techniques

Mills and Butroyd (2014) points out that collaboration with students is essential for action research and it is the students that are evaluating the teaching practice and their learning.

Data that is collected will try to answer the following questions:

- what students do in their free time in terms of technology use and development
- what students think
- how students learn Physics
- how do students manage Programming in Physics
- how students perform in Physics
- what students perceived and desire for the future

Surveys method based on questionnaire, a quasi-experiment, written assessment test and the research will create different perspectives that will be used to verify the findings and achieve triangulation.

The implementation of the first Programming exercise in the Physics lesson took place in October of 2018 followed by a paper questionnaire. The second Programming lesson took place in November followed by the second paper-based questionnaire. In addition, a quasi-experimental research design was achieved. The experimental group consisted of 13 students who participated in the first two Programming lessons. The control group consisted of 14 students. Both groups followed the same unit plans in Physics, and they had the same end of unit test. The independent variable was the implementation of Programming while the dependent variable was the marks obtained. The end of topic test took place in November 2018 and it tested concepts learned by the two classes. Soon after there was an end of topic test that was given to both groups. Descriptive statistics such as mean, median and standard deviation are discussed.

The quantitative and qualitative data is collected and analysed. Paper surveys were completed by the students and the results analysed in the IBM software SPSS. The information collected through the questionnaire contains opinions but no biographical data (age, gender, class). The opinions of the investigated subjects are requested in the forms of appreciation scales, choosing one or more options from a given list and in the form of open questions.

One last Programming workshop was done at the end of the school year 2018-2019 for all 27 Pre-IB Physics students (equivalent of the Norwegian Vg1 students) using the Norwegian learning platform found at diggit.no. The desire of students to have Programming implemented in school was questioned online before and after this last Programming workshop and results were made available to both Physics teachers and the management team.

3.5 Other data collection considerations

3.5.1 Reliability

The desire of students to do Programming was surveyed in October and once again in June of the following year to test for reliability. Probably there are difficulties with replication of action research because goals were personal and involved my own approach to teaching Physics through introduction of Programming. As for me, Programming worked, and I will continue to use Programming in my Physics lessons as a tool that engages students.

3.5.2 Validity and generalisability

According to Mills action research happens when the practices are improved and this leads to internal validity. (Mills & Butroyd, 2014) Since the intervention in this action research was to change the way in which I teach the Mechanics topic in my small Pre-IB Physics generalization to a larger population was not achieved.

3.6 My own approach to introducing Programming into a Physics class

In this section I describe how technology and Programming were incorporated into a secondary school Physics course. Providing good learning material that can be used by teachers in their computational Physics lessons is “the biggest challenge today”. (Malthe-Sørenssen, Hjorth-Jensen, Langtangen, & Mørken, 2015). This is the challenge that I am attempting to address through small steps in this subchapter.

3.6.1 Rationale

The goal of introducing Programming in Physics lessons is to increase understanding of Physics by means of visualization and simulations and increase the student's digital skills. The tasks have been created in such a way that the focus is on Physics, not on Programming.

3.6.2 Tools for computational Physics

There are advantages and disadvantages of using different Programming languages and while this discussion is not part of the paper, the possibilities of using Programming in Science exist, no matter what Programming language is selected. Python is well-known among the scientific community (Community et al., 2015) and that was the reason why it was selected in this action research.

Although the tools presented in this chapter can be used in any field, the examples provided are using Physics concepts such as mass, velocity, momentum and energy.

VPhython can be used in tinket.io for Programming in Python and it can be used to easily create and display animations. VPython is particularly suitable to teach computational Physics. (Borcherds, 2007) By using trinket.io students don't have to download or install any software in the Programming lessons. They just need to use any browser on any device to access the web-based resource from tinket.io, and it works equally well on both Windows and Mac, without the need to log in.

3.6.3 The first Programming lesson in Physics class

The Key Question in the inquiry base Programming lesson is how to compute the sum of two scalars and the sum of two vectors?

First, we define scalars and vectors. Quantities in Physics are either scalars (for example temperature, distance, speed, time, mass or other quantities that have only magnitude) or vectors that have size and direction (such as force, velocity and acceleration). (Tsokos, 2014)

Adding scalars is easy because you can just add the numbers, for example 1500 kg + 70 kg = 1570 kg.

To calculate the sum in Python using trinket.io we need to write the following command line:

```
print (1500+70)
```

After writing one line of code, pressing the play button to compile and run the program will give us the output as seen in Figure 4.

Figure 4 Sum of numbers in Python powered by trinket.io

An alternative would be to assign two variables carMass and driverMass the desired values before computing totalMass as the sum of carMass and driverMass and displaying the results.

The program in Python will in this case contain four lines (comments are included in the command lines by using "//"):

1. carMass = 1500 // we set the value 1500 kg to the mass of the car; the variable called carMass
2. driverMass = 70 // we set the value 70 kg to the driver; the variable called driverMass
3. totalMass = carMass + driverMass // we set the totalMass to be the sum of the two scalars
4. print (totalMass) // we display the total mass totalMass and the totalMass output can be seen in the Figure 5.

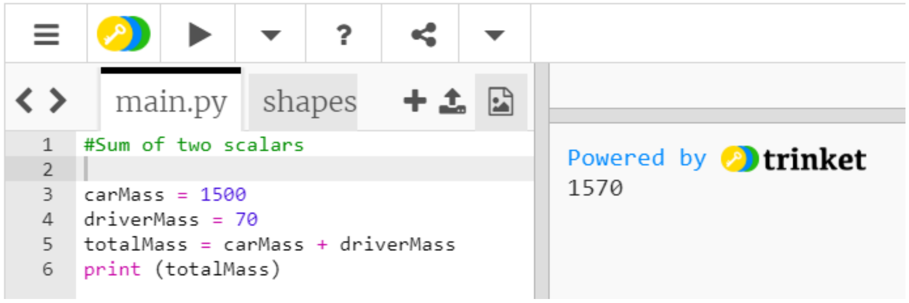
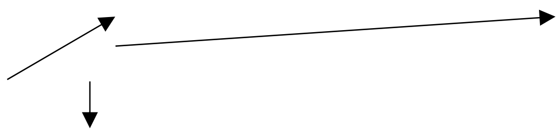


Figure 5 Sum of two scalars in Python powered by trinket.io

A vector has magnitude and direction and is represented by an arrow with a specific size and direction. The vectors in Figure 6.b are all equal, because they have the same length and in addition to being parallel to each other all point in the same direction. In other words vectors do not have to start from the same point to be equal. (Tsokos, 2014)



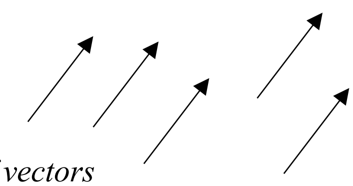


Figure 6 Representation of vectors

a. 2D vectors by arrows ***b. These vectors are equal***

If addition of scalars is a simple arithmetic operation for example 3 kg + 4 kg = 7 kg, the addition of vectors, for example 3N + 4 N will give the answer 7N if and only if the two vectors point in the same direction. Otherwise the answer can vary because of the angle between the two vectors, as seen in the Figure 7.

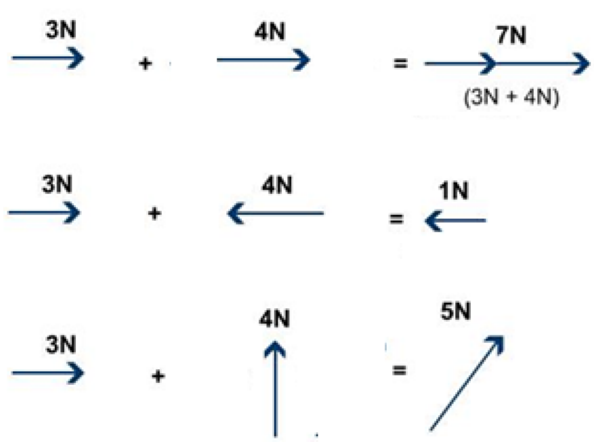


Figure 7 Sum of the two vectors in different situations

The first Programming lesson students are given program sections that they must modify, together with instructions. This reduces the time it takes the students to learn Programming.(Yevick, 2005)

The addition of two vectors, where a vector with size 3N East is added with a vector of size 4N North is written in GlowScript 1.1 VPython using the following 6 lines

1. a=vec(3,0,0)
2. b=vec(0,4,0)
3. aarrow = arrow(pos=vec(0,0,0), axis=a, color=color.green)
4. barrow = arrow(pos=vec(0,0,0), axis=b, color=color.green)
5. c=a+b
6. carrow = arrow(pos=vec(0,0,0), axis=c, color=color.blue)

In https://trinket.io/ teachers can write instructions and share them together with the code as in Figure 8. The first implementation of Programming in the Physics lesson is a creates as output the visual representation of the resultant vector, that is the sum of two vectors.

The first Programming lesson can be accessed at:

https://trinket.io/glowscript/022b629586?showInstructions=true

The links were distributed through ItsLearning, an LMS platform used in the school.

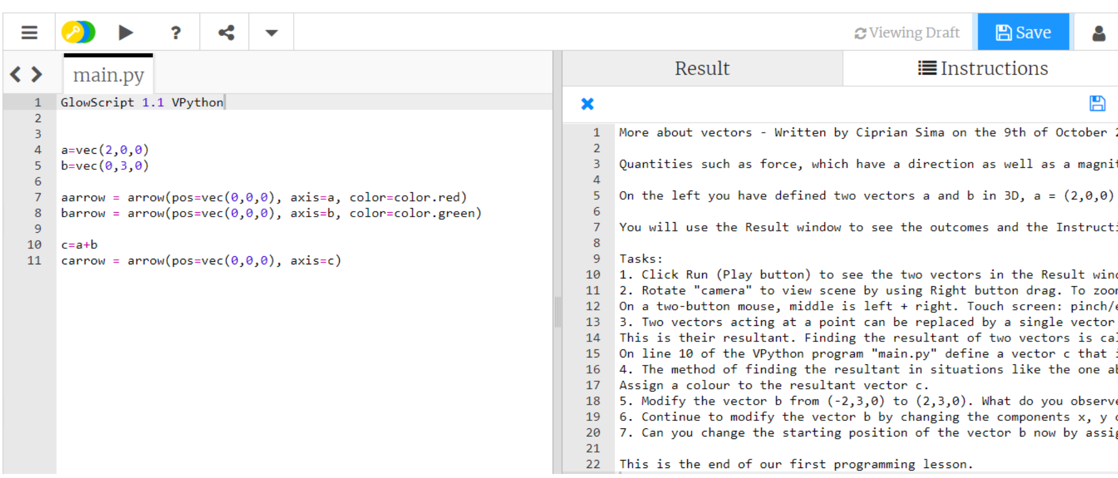


Figure 8 Sum of two vectors – VPython algorithm and instructions

The Programming lessons on vectors for the experimental group took place in October 2018 and Programming was used as a tool to explain the sum of the two vectors.

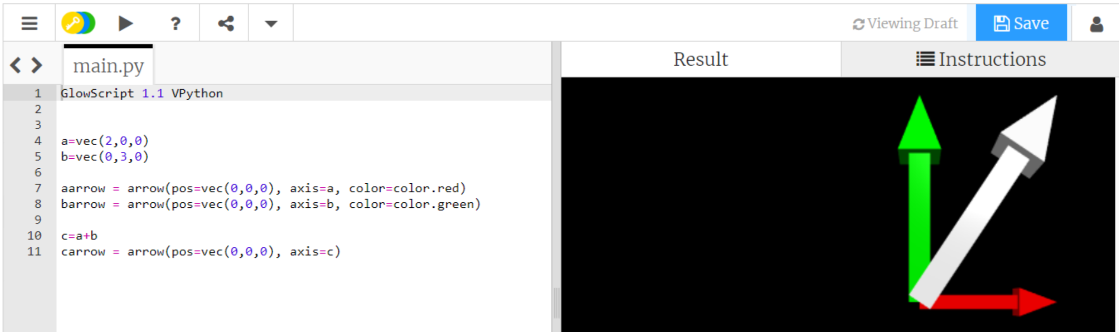


Figure 9 Sum of two vectors – VPython algorithm and output

From the output in Figure 9. we see that to add two vectors **a** and **b** we can plot them, so they start in the same point. We can see that the diagonal in the parallelogram whose sides are **a** and **b** is the vector **c**, such that **c** = **a** + **b**.

Scalars and vectors was the Physics lesson addressed because the implementation of Programming is straight forward using easy to follow instructions and visualization is made possible through the Programming of vectors in VPython.

The first Programming lesson was concluded by a questionnaire (Appendix 1) and followed by problems on the concepts just learned using pen and paper.

3.6.4 The second Programming lesson in Physics class

Another Programming lesson was implemented in November and once again Programming was just a tool to visually determine what will happen in elastic collisions when I have to identical masses.

Considering I have two objects, a blue that is stationary and an orange ball that has identical mass and approaches the blue ball. The second object stops and remains at rest while the first object moves off at the speed that the first object had before the collision.

In this exercise the Programming section contained a lot of comments written in Python using // followed by instructions. The program contained an iteration expressed using a “while loop” statement that is extremely useful in Programming when we want to repeat a statement or a block of statements within an algorithm.

The second Programming lesson and the instructions that students had to follow can be accessed at: https://trinket.io/glowscript/25875fb001?showInstructions=true

Once again, the link was distributed through an LMS platform, so students didn't have to type the link into the browser, but only click on it.

And a picture of the motion of the orange ball can be seen in the Figure 10 below.

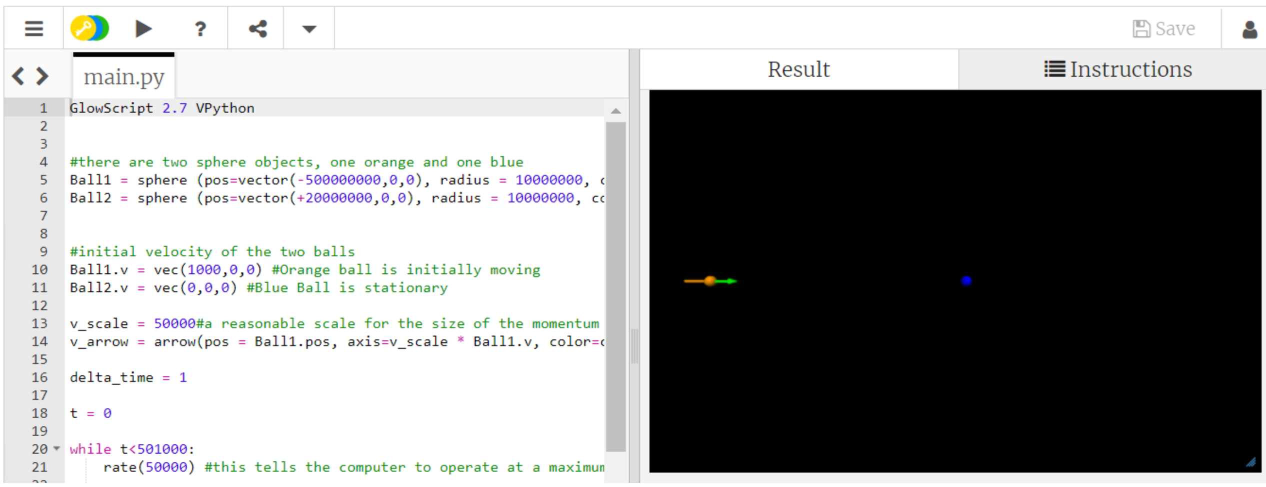


Figure 10 The initial output from the second Programming lesson

The students had to follow the instructions in such a way that the momentum and kinetic energy is transferred from the moving ball to the stationary ball. The desired outcome of the tasks can be seen as a video at: https://cipriansima.trinket.io/sites/conservation-of-momentum-and-kinetic-energy and a screen shot of this motion is presented in Figure 11.

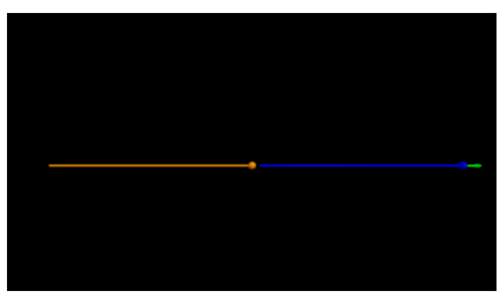


Figure 11 Screen frame from the final video output

The second Programming lesson was also finalized with a questionnaire.

A few weeks later we had an end of topic test. The test included questions that were discussed in the two Programming sessions, but did not include any Programming sections, so students who did not participate in the experiment could complete the questions successfully.

3.6.5 Next steps and improvements

Next steps in the action research involved all Pre-IB Physics students and Physics teachers in the school. At the end of the school year 2018-2019 a visit from a Norwegian learning digital platform took place in our school.

The goal was to deliver a workshop on Programming to the two Pre-IB Physics classes in the school. To do this, a Norwegian Technology Learning Platform that incorporated JavaScript, HTLM and Introduction to Python as courses offered online was used as a tool. The Programming lesson started with a survey and then students learned Programming by doing programs and exercises. The lesson ended with the same survey that was given in the beginning and it is analysed in the next chapter. The course can be accessed using the link https://diggit.no/learn. ("Norwegian Technology Learning Platform," 2019)

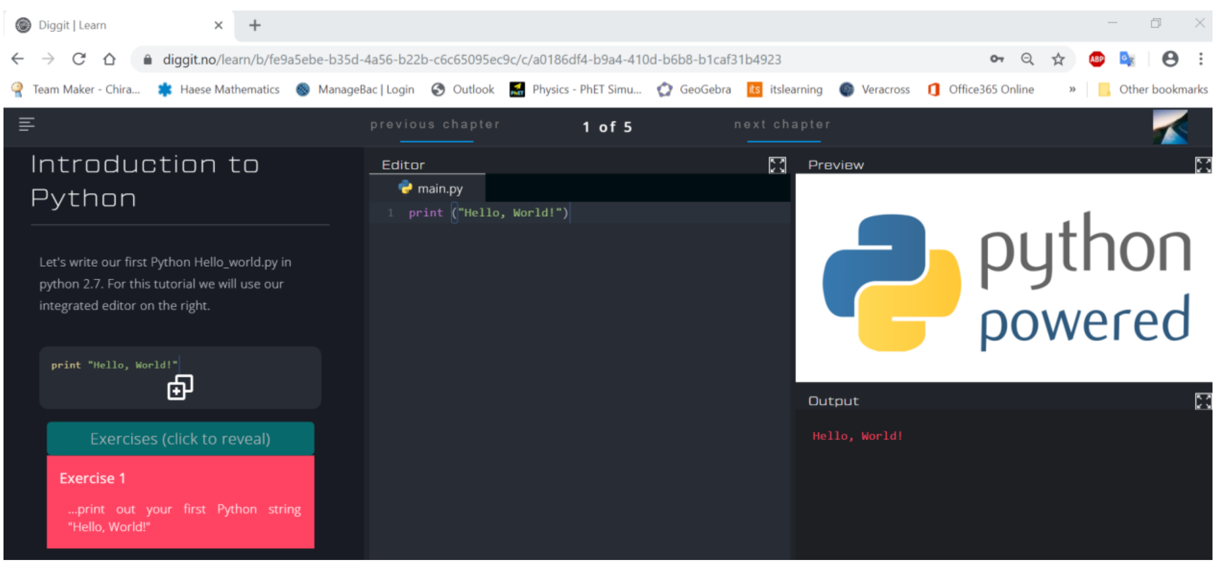


Figure 12 A Norwegian Learning Platform for coding from https://diggit.no/learn

Although it looks similar to other websites that offer online Programming courses in Python (such as https://www.codecademy.com/learn, https://trinket.io/ or https://www.w3schools.com/) the advantage is that it is a local software company from Norway.

Students suggested a course in programming and that was offered at the end of the school year. Another possibility would have been to introduce flowcharts and this was considered but not achieved with the current group of students. One advantage of using the Flowgorithm software is that once the algorithm is written as a flowchart the software can easily generate the code in the desire language, such as C#, C++, Java, Python or many more as seen in the Figure 20. Flowgorithm at the moment is only available for Windows platforms. Students and teachers can download this tool freely from the following link: http://flowgorithm.org

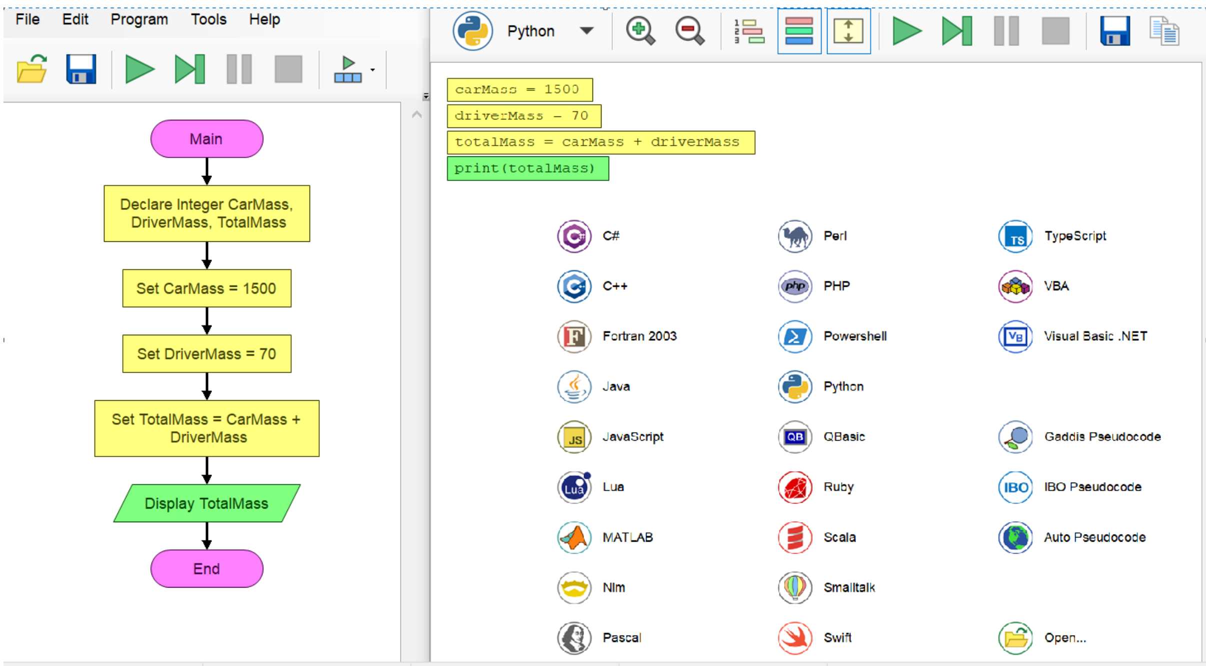


Figure 13 Flowchart and Python code generated by the Flowgorithm software

Chapter 4. Analysis and results

Surveys and examination results are presented and analysed in this chapter.

4.1 Processed data for the experimental group

The research question answered in the thesis is “What is the Impact of Technology and the Implementation of Programming to a Physics Class in Secondary School?”. In this chapter we analyse the results from the surveys by looking at how technology is being used by students and consider their viewpoints in terms of Programming and its role in a Physics lesson.

Figure 14 shows an overview of the profile of students in terms of the time allocated to the use of software each week.

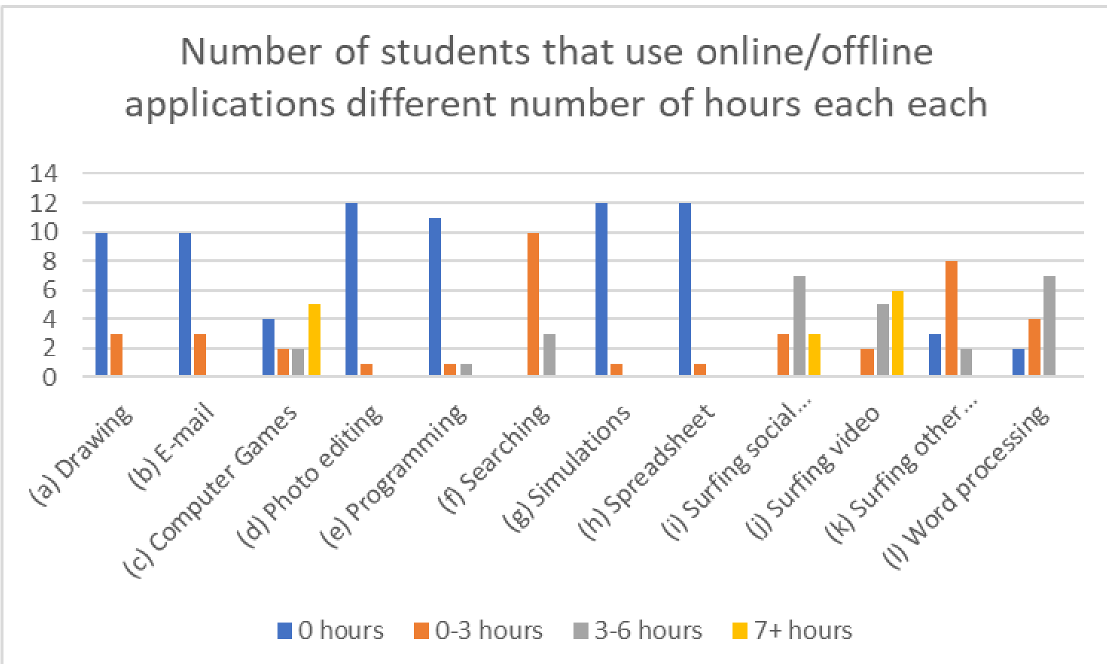


Figure 14 Collaborators use of software (hours per week)

To reach students computer games, digital social media or video and other content can be created or used by teachers. For the next Physics lesson that will embed Programming I considered myself the introduction of a visual output, similar to a video in order to better communicate the desire of the outcome of the Programming session.

I have anticipated some challenges in implementing Programming in the Physics lesson, but the first Programming lesson demonstrated that my expectations on students’ abilities and

Programming skills were too high. Even if they are very capable in using a computer, they need more guidance in developing applications or simulations by Programming.

Two things that majority of students in this Physics class wanted to be able to do with technology even if the skills were not in place was the creation of websites (62%) and the creation of apps/simulations using Programming. The pie chart below (Figure 15) shows that 77% of students would like to use more of their time to be able to create apps or simulations with Programming even if they don't know how now.

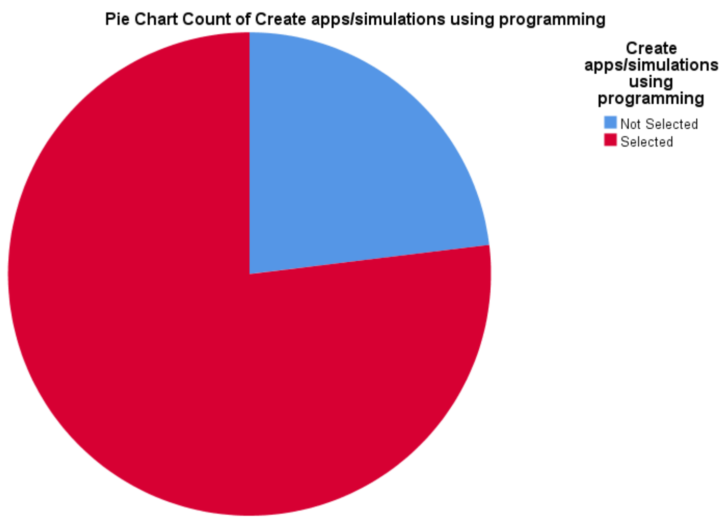


Figure 15 Pie chart - The desire to create apps/simulations using Programming

Doing Programming in Physics lessons is the intervention that happened in this action research paper because it was considered as having the potential of improving their performances.

The perceived outcome from the student is analysed from statement 3 in the first survey (Appendix A1. 3) and sees if Programming helped the students understand the Physics lesson or not. The scale consists of 7 Likert scale items, but I only specify one of them, where by 7 you totally agree with the statement. The mean is 5.23 on a 1 to 7 scale where 7 means "totally agree" and SD = 1.48 so this result indicates that students have benefited from the Programming lesson.

I treated the scale in Figure 16 as continuous and therefore I have a histogram that includes the 0 and 8 and not a bar graph from that includes the extremes 1 to 7.

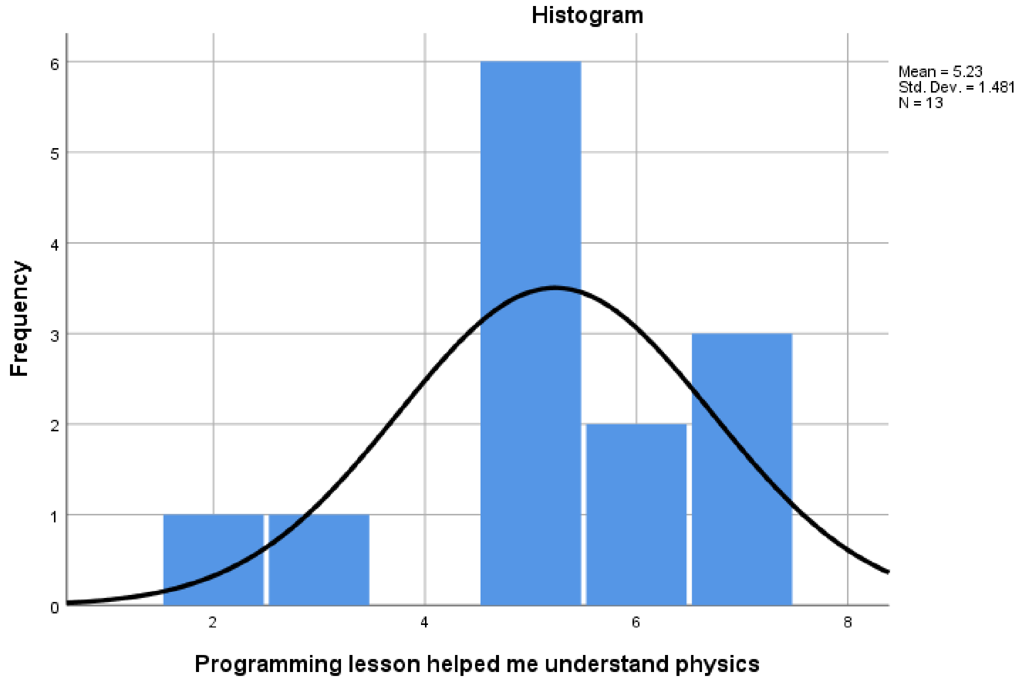


Figure 16 Histogram Programming supporting understanding of Physics

The topic of vectors and scalars can be confusing because we are under the impression that 3 N + 4 N is always 7 N when in fact for most cases it is not 7 N because vectors depend on direction. The result is backed up by research that states that Programming can help so that students are not confused, and they overcome misunderstandings in Physics concepts. (Taub, Armoni, Bagno, & Ben-Ari, 2015).

I know there are simulations online already that show sum of two vectors and I was using them and many more myself and my students are familiar with simulations, but Programming can give us the opportunity to create visual content, such as simulations as well. The parallel Pre-IB Physics class is also using simulations and in recent studies in the United States that compare learning achievement for undergraduate students using traditional hands-on equipment and non-traditional lab such as virtual simulations there is an equal or higher outcome achievement in favour of simulations. (Brinson, 2015) Whether the creation of simulations through Programming can have the same outcome will be further considered comparing the results in grades achieved by the two classes in the next section.

Another result that also emphasized student experience in the Physics class is seen in the responses to the statement "My first Programming lesson in Physics was ...." One of the three options had to be circled: organized, chaotic but rewarding or chaotic and frustrating. (Appendix A 1.5).

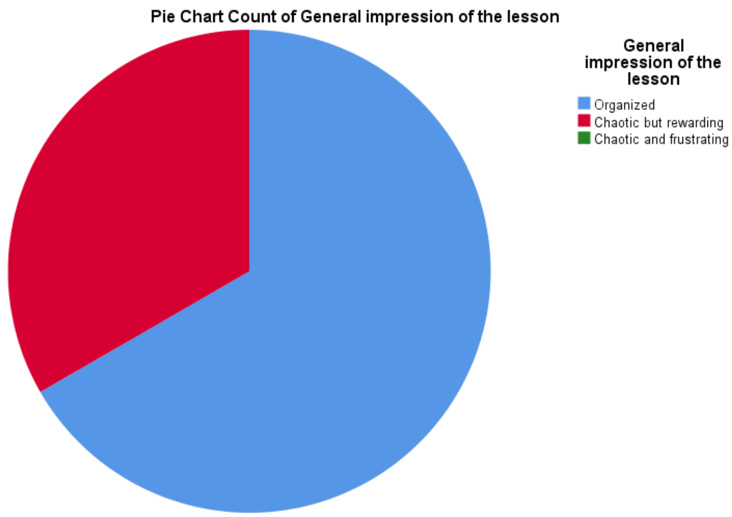


Figure 17 Students' impression about the organization of the class

According to the pie chart above students view the Physics lesson as organized (62%) or chaotic but rewarding (38%) but not frustrating. Programming in the Physics lesson can be challenging but useful and interesting. The majority of students (92%) completed all tasks that involved Programming.

The second student survey looked more at the outcomes and challenges and steps for improving student experiences with Programming. A video output was created with the desired outcome and 75% strongly agreed that the visual outcome made the goal clear. This can be seen in relation to the findings from the first survey where students identified surfing video as one of the activities they are involved in regulatory.

When writing a computer program there are three types of errors could occur:

- Syntax errors

- Runtime errors occur when the program is executed

- Logic errors are errors in the design

Only few students indicated that they had encountered problems when writing the program into VPython, so majority were able to debug their own programs. As a help there were instructions to follow and comments in the program were found very useful.

As things that can help in writing and understanding the computer programs 4/12 wrote comments such as "crash course in Programming", 3/12 wanted printed instructions while 5/12 wanted instructions and programs organized in a course online.

The information from the second Programming lesson can be used as a starting point for finding another focus for a new action research cycle.

Integrating Science, Technology, Engineering and Mathematics (STEM) activities, can make both Mathematics and Science fun and interesting and can help students do much more than just learn. In the U.S., the National Science Board made two recommendations to the US government that address coherence in the Nation’s STEM education system and that teachers are well-prepared. The goal is that U.S. students acquire the knowledge and skills in Science, technology, engineering and Mathematics that will enable them to be successful in a technological 21st century. (Foundation, 2007)

4.2 Comparative analysis of marks for the experimental and control groups

The end of topic test in mechanics was administrated in November 2018 and had a total of 100 marks and the same test was given to both groups. The raw data table (appendix 2) contains the marks awarded out a total of 100 marks.

The test included concepts that were discussed in the two Programming sessions but did not include any Programming section, so students that have not participated in the experiment could complete the questions successfully.

The effect of introducing Programming in Physics lessons has been attempted measured through descriptive statistics.

By looking at the box-and-whisker plots for the two groups in the Fig below ,we can see the distribution of grades and the median mark.

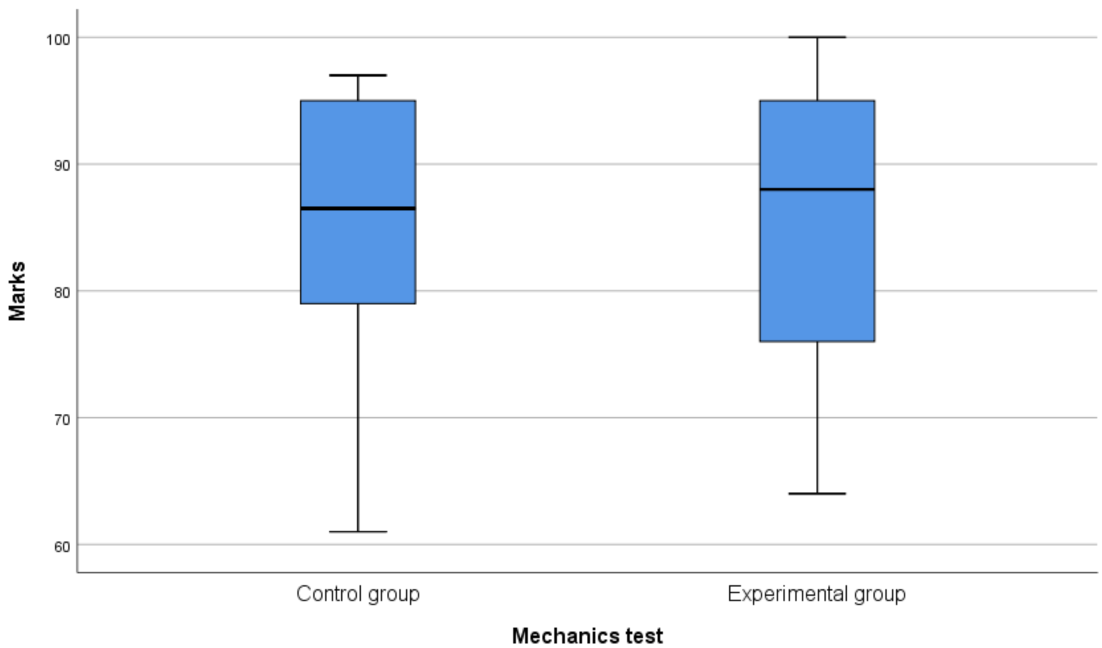
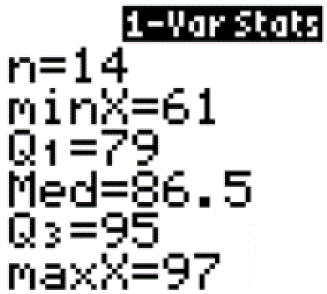
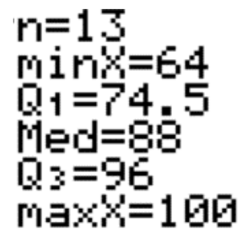


Figure 18 Box and whisker plots for the marks of the two classes

Both groups have very academic students that selected Physics out of the three Sciences (Physics, Chemistry and Biology) offered in the school as elective subjects. While Programming developed their understanding of Physics through Programming the control group has used the time to study trough other means such as hands on experiments or solving Science problems.



For the control group that did not do Programming before the test, we have a number of 14 students, maximum mark = 97, minimum mark = 61, range = maximum mark – minimum mark of 36, lower quartile of 79, upper quartile of 95 mark, an interquartile range = upper quartile – lower quartile of 16.5 marks and the median of 86.5,.



For the experimental group we have 13 students, maximum mark = 100, minimum mark = 64, range = maximum mark – minimum mark of 36, lower quartile 74.5, upper quartile of 96, interquartile range = upper quartile – lower quartile of 10 of 21.5 marks and median of 88,

Comparing the two data sets we can see that the median mark is higher for the experimental group that had Programming, but we also see that the box in the middle representing 50 percent of the students have a bigger variation of grades in the experimental group. The fact that the median is higher (the seventh score for the 13 students in the experimental group after the grades where arranged in order) cannot be linked as cause and effect with Programming. The grades achieved reflect student's performance, but the performance can be influenced by different factors. Students can correctly solve problems in Physics if they use previous knowledge or have done similar examples and have plausible reasons.

How else can we compare the data? We will compare the average and the standard deviation using a t-Test.

Descriptive Statistics

Mechanics test		N	Minimum	Maximum	Mean	Std. Deviation
Control group	Marks	14	61	97	85.14	10.502
	Valid N (listwise)	14				
Experimental group	Marks	13	64	100	85.46	11.702
	Valid N (listwise)	13				

Table 1 The mean and standard deviation for the control group and experimental group

Although the average mark for the experimental group is 85.46 and it is higher than the average mark for the control group (85.14) the values are not that different.

The distributions of marks are seen in Figure 19.

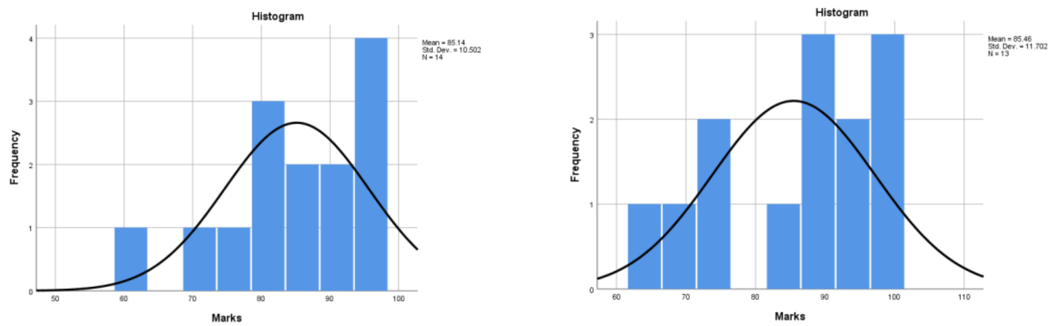


Figure 19 Histogram representing the marks obtained in the two classes

a) Histogram for the control class b) Histogram for the experimental class

The difference in the averages is very small, and the question now is if the difference is significant.

To compare the two means and to understand the variation we see in the figure above an Independent (unpaired) sampled T-test is carried out in SPSS, with the null hypothesis that there is not a significant difference between the two groups; any observed differences may be due to chance and sampling error.

		Levene's Test for Equality of Variances				
		F	Sig.	t	df	Sig. (2-tailed)
Marks	Equal variances assumed	.282	.600	-.075	25	.941
	Equal variances not assumed			-.074	24.178	.941

Table 2 Independent Sample T-Test for the marks achieved by the two classes

The F value in the Levene's Test for Equality of Variances is 0.282 and being bigger than 0.05 means that we should assume equal variances (distributions). The p value is 0.941 and it's much bigger than 0.05 that means there is no statistically significant difference between the two classes. The probability of this result, assuming the null hypothesis, is 0.941.

An unpaired-samples t-test was conducted to compare the results in the mechanics test for students participating in the Programming lessons and those who did not participate. There was no significant difference in the scores in the mechanics end of topic test for the Programming (M=85.5, SD=11.7) and non-Programming (M=85.1, SD=10,5) conditions; $t(25)=0.075$, $p=0.941$. These results suggest that Programming does not have any effect on the scores in the mechanics test. There was no significant difference in the scores in the Mechanics end of topic test for the Programming (M=85.5, SD=11.7) and non-Programming (M=85.1, SD=10,5) conditions; $t(25)=0.075$, $p=0.941$. These results suggest that Programming does not have a significant effect on the scores in the mechanics test.

Next step in the action research was to combine the two classes and look for the learning experiences of the students and their feedback.

4.3 Pre and post the last Programming session

In the last Programming workshop that was carried out at the end of the school year 2018-2019 in June the two Physics classes were joined. Results can be used internally in the school, or as a reference point. The number of participants taking Physics and participating in the last computational Physics lesson was now 27, representing all students from the combination of the two Pre-IB Physics classes in the school.

Out of 27 students only 19 students answered the survey question "Would you like coding experiences in your school?" before and 18 students answered the same question at the end of the Programming lesson.

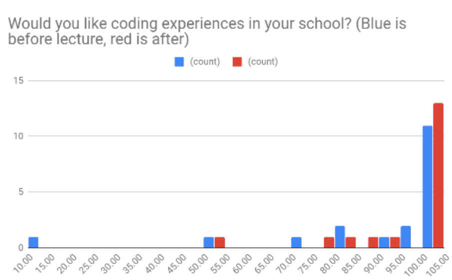


Figure 20 Responses to question: Would you like coding experiences in your school?

Although not all students have answered the questions “Would you like coding experiences in your school?” (with 0 = “totally disagree” and 100 = “totally agree”), the graph indicates somehow more students wanted to have Programming implemented in school after the Programming lesson took place.

Group Statistics

	Groups	N	Mean	Std. Deviation	Std. Error Mean
Would you like coding experiences in your school? (0 = no, 100 = yes)	Before	19	88.26	22.703	5.208
	After	18	93.72	12.555	2.959

Table 3 Would you like coding experiences in your school?

The question that I focus on in the last Programming session is “Would you like coding experiences in your school?” and looked at their answers before and after the last Programming lesson. With 0 = no and 100 = yes the average was very big from the start. But did others change their minds after the Programming session, considering that some of the students didn’t have Programming before, not as a subject and not in the Physics class?

The table above indicates that the average is higher after the lesson (M=93.72, SD=12.56). I see that the mean result before the lesson was (M=88.26, SD=22.70) lower so I can state that the average number of students that would like to see Programming in school is equal or more after the lesson compared to before the lesson. But is there a significant difference and can I state that more students on average want to have Programming experiences in schools compared to before the Programming lesson? A paired t-test would cannot be performed since there is not the same number of students answering the question before and after the Programming lesson.

I will use the independent (unpaired) T-test as seen in table below.

Independent Samples Test

		Levene's Test for Equality of Variances				
		F	Sig.	t	df	Sig. (2-tailed)
Would you like coding experiences in your school? (0 = no, 100 = yes)	Equal variances assumed	2.370	.133	-.898	35	.375
	Equal variances not assumed			-.911	28.367	.370

Table 4 Independent Sample T-Test for desire to have programming

An unpaired-samples t-test was conducted to compare the results from one question that could be important for the management in the school: Would you like coding experiences in your school? (0=no, 100=yes). There is no significant difference in the answers before (M=93.72, SD=12.56) and after the Programming lesson (M=88.26, SD=22.70), t(35)=0.898 and p=0.375, these results . These results suggest that the Programming lesson did not significantly changed the opinion of other students regarding Programming.

The size is of the class is small, but the test was done to determine reliability of the results.

4.4 Impacts and relation to existing research

Students were engaged with programming and showed a genuine interest in possibilities that programming can bring into a Physics class. Internationally there is a decline in the number of students taking Physics in secondary school (Gill & Bell, 2011). In UK one way to increase student interest in Physics is the introduction of the Action Research for Physics (ARP) programme that encourages teachers to use action research to increase student interest in Physics.

What was achieved through the process of implementing Programming in a Physics class was developing of new formats for instructions where Programming is a learning activity that helped students create digital context. The results need to be considered using also the European

Framework for the Digital Competence of Educators when it comes to both teaching and facilitating learners' digital competence. (Redecker & Punie, 2017)

4.4.1 National efforts to equip teachers with Programming skills

In order to implement Programming in lessons teachers need Programming skills. Reluctance to use Programming in Science class could be overpowered by professional development opportunities. (Stevens et al., 2018)
National in-service workshops have been developed in Norway with the goal of preparing teachers for the new curricula that would be implemented from 2020 in Mathematics and Science that includes Programming in those subjects. UiO, OsloMet and Simula have seminars for all teachers and school leaders to answer questions such as why students need Programming skills and in what way should be taught. (UiO)

Under University of Oslo (UiO) the Centre for Computing in Science Education (CCSE) is organizing ProFag – a continuing education that will use Programming to strengthen and develop the Science subjects: natural Science, Biology, Chemistry, Physics and Mathematics.

Norway led the way in Programming when in 1961 Simula I was developed by O.J.Dahl and K. Nygaard at the Norwegian Computer Centre. Simula 67 was the first object-oriented Programming language and it was the inspiration for the development of the C++ language. During the autumn of 2018 the Simula School arranged a pilot course in Python Programming with practical guidance on how Programming can be used as a tool in the classroom. (Simula, 2018) Looking at the current knowledge in national and international research on implementation of Programming in secondary school Physics lessons, the current study becomes significant for colleagues in Physics departments worldwide and educational institutions alike.

4.4.2 Reasons to study Programming in different disciplines

If Programming is a basic skill, then, according to University of Oslo, a modern Physics education must include it. Universities across the world, including University of Oslo have introduced Programming in the first year of study for Physics degree. Among the arguments they provide for integrating Programming in Physics lessons are a necessity of modernization in the Physics field. (Malthe-Sørenssen, Hjorth-Jensen, Langtangen, & Mørken, 2015)

In one university article ("Matematikk og programmering for teknologistudenter ved NTNU") Marius Thaule aims to connect Programming and Mathematics and supports the claim that learning outcomes for the students taking the undergraduate course have increased. (Thaule, 2015). Students in my own class expressed desire to have programming and measurement of the learning outcomes was very limited, but the skills learned can be considered as positive outcomes.

Some studies claim that student's exposure to technology will boost performance in Mathematics and Science (Skryabin, Zhang, Liu, & Zhang, 2015). Will experiencing with technology can have positive effects on Mathematics some studies have seen that using technology for entertainment purposes has a negative effect on Science. (Skryabin, Zhang, Liu, & Zhang, 2015) It is vital for students to understand the difference between the scientific theory and opinion and be critical of so-called "Science knowledge" found on the Internet.

Successful attempts have been done in using of Mathematics and Physics problems and situations in traditional Programming lessons that helped students identify misconceptions and clarify concepts. (Nuutila et al., 2005)

A recent interdisciplinary study involving the combination of two academic subjects on "Teaching Programming by Computational Physics" carried out in a Chinese secondary school connected Programming and Physics in a Computer Science course. As an outcome, students performed better in both Programming and Physics exams. (Lin, Wang, & Wu, 2019). My study is in a Physics course not a Computer Science and marks of students that have done programming in Physics have been equal to other students that have not done programming.

Integrative STEM (Science, technology, engineering, and Mathematics) education will allow students to make connections and provide the skills needed in a workforce surrounded by technology. (John, Bettye, Ezra, & Robert, 2016) There are secondary schools that have implemented the interdisciplinary STEM instructions and many universities in the United States offer summer courses for middle and high school students that promote STEM instructions, including those at Johns Hopkins University (https://cty.jhu.edu/summer/grades7-12/academic/catalog/Mathematics .html), Stanford (https://summerinstitutes.stanford.edu/) and Northwestern University (https://ocep.northwestern.edu/programs.html). The results of those courses have been decisive in career decisions of the students participating. Whether my study has influenced the results is not known but students were already interested in Physics from before.

Programming is one of the essential skills that scientists should develop, especially if they want to communicate directly with the computer and not rely on some other software that was created for a different purpose. (Theisen, 2019) The basics of Programming are easily learned and are among the obligatory skills a scientist should have. (Ekmekci, McAnany, & Mura, 2016)

In medical field Programming is also recognized as a life skill that doctors should acquire to be able to adapt to the modern medical world. (Morton, Smith, Lwin, George, & Williams, 2019). My study is a first step to expose the students to programming.

Becoming a novelist is not a goal for all those learning to write and the same principle should apply to Programming. Learning Programming will not make students programmers but will give them skills that could be needed by as scientists, lawyers or artists. (Curzon et al., 2019)

Years ago, it was considered that knowing how to use a computer require Programming skills. Today we see that computers come with easy to use software and knowing Programming is not a requirement for using a computer. Even careers in computing field are available to nonprogrammers as well as programmers. (Morelli & Walde, 2005)

Chapter 5 Conclusion

Bringing Programming into a Physics lesson can add to the complexity teachers must learn to deal with in this technological area. To be successful all teachers and students must acquire specific skills. As computer applications are expanding because of the development in both hardware and software technologies there is a time of change in curriculum. Students can be more than users. They can become developers by creating digital resources on personal laptops and enter a world that before was limited to highly qualified programmers from different institutions that have expensive technological facilities. To bring about change in approaches to teaching and learning using technology factors such as time and training must be considered together with the introduction of the new curriculums. Changes affect teachers and students alike.

Targeted Programming training needs to be provided to students and teachers to enable them to utilize Programming and technology as a learning tool in different areas of the curriculum. There is a need to increase the number of Physics, Biology, Chemistry and Mathematics teachers that have relevant expertise in computational thinking. Programming skills need to be acquired by experienced teachers through tailor made, practical training courses or graduate university degrees that exist or can be initiated worldwide. New teachers would benefit by incorporation of Programming into undergraduate degrees courses in Science and Mathematics.

The paper shows a teacher's approach to using technology and Programming in Physics lessons. The teacher tries to move students towards development of digital resources by introducing Programming in a Physics class, but the focus is not technology. The emphasis is on student learning and experiences from the activities designed to increase student interest in Physics. A common characteristic for all students in school seems to be the desire to experience with Programming in their studies. This is seen through the very positive student related feedbacks from the questionnaires reflecting on the experience of the students within the Programming lessons in Physics. It was also important for me to know student's opinions. Students valued the lessons and believed that they developed an understanding of concepts and principles in Physics through basic Programming.

This study indicates that introducing Programming in a high school Physics class was successful and engages students in abstract learning (Physics concepts) and concrete application development through Programming. Comparing grades, findings show that the median and the average mark for the students that used Programming in their Physics classes is equal or higher to the average mark for the non-Programming Physics class. The Programming sessions had educational value and students were exposed to computational thinking by developing and modifying programs designed to solve problems in Physics.

Many times, we as teachers want to cover the entire content that will be tested in the exams and we sacrifice other learning experiences because there is only a limited, allocated amount of time. Acquiring Programming skills also requires time, so both time and developing the proper skills can be a challenge for teachers. A modification to the order and the time allocated for Programming sessions could increase the impact.

The class is small and findings although limited showed engagement and the desire to see Programming in school. Because the research was done in a private international school the implementation process was relatively uncomplicated, but implications were complex.

I was able to disseminate my findings with the entire Science department and we decided to share practices that can be discussed, improved and implemented in our own practice.

At the time of the research I was already using technology in my own Physics class and because I have good Programming skills the implementation into Physics lesson was accomplished with ease. As for the personal implications for the future, I am looking forward into teaching other topics in Physics with the help of Programming. Findings were also shared with the management team and it was decided that I'll be teaching Programming in a newly created Computer Science course in my school.

Future research into the planning process of integrating Programming in Science is necessary to enable colleagues to find lasting and more significant contributions. Findings can be steps towards understanding how students can use Programming in a Physics lessons, but other practice researchers can find other projects to increase student interest in Science.

Both the National and International curricula will change in time and will introduce Programming either as a subject or implemented in Natural Sciences, including Mathematics and Physics. Computational thinking will be included in university and before that. Are we ready for the change?

References

Barak, M. (2005). From order to disorder: the role of computer-based electronics projects on fostering of higher-order cognitive skills. *Computers & Education, 45*(2), 231-243. doi:https://doi.org/10.1016/j.compedu.2004.12.001

Cambridge, D. O. (2019). Retrieved from https://dictionary.cambridge.org/dictionary/english/programming. Retrieved October 22, 2019 https://dictionary.cambridge.org/dictionary/english/programming

CCSE - Center for Computing in Science Education. (2019). Retrieved from https://www.mn.uio.no/ccse/english/about/

Chabay, R., & Sherwood, B. (2008). Computational physics in the introductory calculus-based course. *American Journal of Physics, 76*(4), 307-313. doi:10.1119/1.2835054

Chami, F. (2006). *Molecular dynamics simulations of the bombardment of iron by chromium ions.* (Thesis (Ph.D.)). Sheffield Hallam University, Retrieved from http://shura.shu.ac.uk/19441/

Curzon, P., Bell, T., Waite, J., & Dorling, M. (2019). Computational Thinking. In A. V. Robins & S. A. Fincher (Eds.), *The Cambridge Handbook of Computing Education Research* (pp. 513-546). Cambridge: Cambridge University Press.

Dimitriou, K., & Hatzitaskos, M. (2005). *Core Computer Science for the IB Diploma Program*. UK: Express Publishing.

Ekmekci, B., McAnany, C. E., & Mura, C. (2016). An Introduction to Programming for Bioscientists: A Python-Based Primer. *PLoS Computational Biology, 12*(6), e1004867. doi:10.1371/journal.pcbi.1004867

Engelhardt, K., Punie, Y., Chioccariello, A., Ferrari, A., Dettori, G., Kampylis, P., & Bocconi, S. (2018). *Developing computational thinking in compulsory education*.

González, M., González, M., Llamas, C., Martin, E., Vegas, J., Martínez, O., . . . Herguedas, M. (2014). *Mobile Phones for Teaching Physics: Using Applications and Sensors*.

Guðmundsdóttir, G. B., & Hatlevik, O. E. (2013). Digital Competence and Students' Productive Use of Computers in School. In *New Voices in Norwegian Educational Research* (pp. 69-81). Rotterdam: SensePublishers.

Hatlevik, O. E., Throndsen, I., Gudmundsdottir, G. B., & Olsen, R. V. (2015). Kapittel 9. In *Læring av IKT* (pp. 171-186).

IB, D. (2014a). Physics guide (first assessment 2016). Retrieved from https://resources.ibo.org/dp/subject-group/Physics/resource/11162-occ-file-d_4_physi_gui_1402_1_e/data/d_4_physi_gui_1402_5_e.pdf

IB, D. (2014b). Physics teacher support material (first assessment 2016). Retrieved from https://ibpublishing.ibo.org/server2/rest/app/tsm.xql?doc=d_4_physi_tsm_1408_1_e&part=5&chapter=1

John, M., Bettye, S., Ezra, T., & Robert, W. (2016). A formative evaluation of a Southeast High School Integrative science, technology, engineering, and mathematics (STEM) academy. *Technology in Society, 45*, 34-39. doi:https://doi.org/10.1016/j.techsoc.2016.02.001

Kunnskapsdepartementet. (15.11.2017, 15.11.2017). Rammeverk for grunnleggende ferdigheter. Retrieved from https://www.udir.no/laring-og-trivsel/lareplanverket/grunnleggende-ferdigheter/rammeverk-for-grunnleggende-ferdigheter/

Malthe-Sørenssen, A., Hjorth-Jensen, M., Langtangen, H. P., & Mørken, K. (2015). Integrasjon av beregninger i fysikkundervisningen. *Uniped, 38*(4), 303-310

E. Retrieved from http://www.idunn.no/uniped/2015/04/integrasjon_av_beregninger_ifysikkundervisningen

Mills, G. E., & Butroyd, R. (2014). *Action Reserach; A guide for the Teacher Reseracher*. United Kingdom: Pearson Education Limited.

Morelli, R., & Walde, R. (2005). *Java, Java, Java, Object-Oriented Problem Solving (3rd Edition)*: Prentice-Hall, Inc.

Morton, C. E., Smith, S. F., Lwin, T., George, M., & Williams, M. (2019). Computer Programming: Should Medical Students Be Learning It? *JMIR Med Educ, 5*(1), e11940. doi:10.2196/11940

National curriculum in England: computing programmes of study. (2013). Retrieved from https://www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study/national-curriculum-in-england-computing-programmes-of-study

Nilsen, T., Bergem, O. K., & Kaarstein, H. (2016). *Vi kan lykkes i realfag (Resultater og analyser fra TIMSS 2015)*: Universitetsforlaget.

Norwegian Technology Learning Platform. (2019). Retrieved from https://diggit.no/learn/

Nuutila, E., Törmä, S., & Malmi, L. (2005). PBL and Computer Programming — The Seven Steps Method with Adaptations. *Computer Science Education, 15*(2), 123-142. doi:10.1080/08993400500150788

Redecker, C., & Punie, Y. (2017). *European Framework for the Digital Competence of Educators: DigCompEdu*. Retrieved from https://ec.europa.eu/jrc/en/publication/eur-scientific-and-technical-research-reports/european-framework-digital-competence-educators-digcompedu

Simula. (2018). Simula educates teachers in programming. Retrieved from https://www.simula.no/news/simula-educates-teachers-programming

Skryabin, M., Zhang, J., Liu, L., & Zhang, D. (2015). How the ICT development level and usage influence student achievement in reading, mathematics, and science. *Computers & Education, 85*, 49-58. doi:https://doi.org/10.1016/j.compedu.2015.02.004

Spectrum, I. (2018). Interactive: The Topc Programming Languages 2018. Retrieved from https://spectrum.ieee.org/static/interactive-the-top-programming-languages-2018

Stevens, S. L. R., Kuzak, M., Martinez, C., Moser, A., Bleeker, P., & Galland, M. (2018). Building a local community of practice in scientific programming for life scientists. *PLOS Biology, 16*(11), e2005561. doi:10.1371/journal.pbio.2005561

Taub, R., Armoni, M., Bagno, E., & Ben-Ari, M. (2015). The effect of computer science on physics learning in a computational science environment. *Computers & Education, 87*, 10-23. doi:https://doi.org/10.1016/j.compedu.2015.03.013

The Norwegian Government. (26.06.2018). Fornyer innholdet i skolen. Retrieved from https://www.regjeringen.no/no/aktuelt/fornyer-innholdet-i-skolen/id2606028/

Theisen, K. J. (2019). Programming languages in Chemistry: a review of HTML5/JavaScript. *Journal of Cheminformatics, 11*(1), 11. doi:10.1186/s13321-019-0331-1

Throndsen, I., Gudmundsdottir, G. B., Hatlevik, O. E., Loi, M., Rohatgi, A., & Olsen, R. V. (2015). *Læring av IKT. Elevenes digitale ferdigheter og bruk av IKT i ICILS 2013*. Universitetet i Oslo Universitetsforlaget.

Training, T. N. D. f. E. a. (2019). Algoritmisk tenking. Retrieved from https://www.udir.no/kvalitet-og-kompetanse/profesjonsfaglig-digital-kompetanse/algoritmisk-tenkning/

Tsokos, K. A. (2014). *Physics for the IB Diploma* (Sixth edition ed.). United Kingdom: Cambridge University Press.

UiO. (21/02/2019). Programmering i skolen – hvor står vi, og hvor skal vi? Retrieved from https://www.mn.uio.no/om/aktuelt/arrangementer/programmering-i-skolen-hvor-star-vi-og-hvor-skal.html

Urban, M. J., & Falvo, D. A. (2016). *Improving K-12 STEM Education Outcomes through Technological Integration*: IGI Global.

Vondracek, M. (2007). Diminishing the Gap Between University and High School Reserach Programs: Compuational Physics. *The Physics teacher, 45*(7).

Yevick, D. (2005). A First Course in Computational Physics and Object-Oriented Programming with C++. *A First Course in Computational Physics and Object-Oriented Programming with C++, by David Yevick, pp. 418. ISBN 0521827787. Cambridge, UK: Cambridge University Press, April 2005., 74*. doi:10.1119/1.2203647

Appendix 1 Student surveys

Form A.1 Individual student survey done at the end of the first Programming lesson

1. Each week I spend online/offline on average … hours (circle one for each item):

(a)	Drawing	0	1-3	3-6	7+
(b)	E-mail	0	1-3	3-6	7+
(c)	Games	0	1-3	3-6	7+
(d)	Photo editing	0	1-3	3-6	7+
(e)	Programming	0	1-3	3-6	7+
(f)	Searching	0	1-3	3-6	7+
(g)	Simulations	0	1-3	3-6	7+
(h)	Spreadsheet	0	1-3	3-6	7+
(i)	Surfing social media	0	1-3	3-6	7+
(j)	Surfing video	0	1-3	3-6	7+
(k)	Surfing other content	0	1-3	3-6	7+
(l)	Word processing	0	1-3	3-6	7+

2. I will like to increase my use of technology in the following ways (check all that apply even if you don't master the skills necessary):
   _____ Conduct research via the Internet
   _____ Create apps/simulations using Programming
   _____ Create documents with word processing
   _____ Create offline presentations
   _____ Create websites
3. The first Programming lesson done in class helped me understand the Physics lesson. (On a scale of 1-7, where by 7 you totally agree with the statement, circle the one applicable to you):

1 2 3 4 5 6 7

4. I managed to finish ___ the tasks from the Programming lesson (check one) _____ all _____ almost all _______ not many of _____ none of

5. My first Programming lesson in Physics was ….. (circle one)

   organized chaotic but rewarding chaotic and frustrating

Thank you for your participation in this project. October 2018 – Ciprian Sima

Form A.2 Individual student survey done at the end of the second Programming lesson

Note: the second Programming lesson can be found at:
https://trinket.io/glowscript/25875fb001?showInstructions=true

Things introduced in this program were comments and the "while loops". The statements found inside the loop were evaluated if the time variable was not bigger then a specific value.

1. There was a visual presentation of the desired outcome present in the instructions and that helped me understand better what I need to achieve. (On a scale of 1-5, where by 5 you totally agree with the statement, circle the one applicable to you):

1 2 3 4 5

2. I have encountered problems/challenges when writing the loop that I had to fix before the outcome was as desired (check all that apply):
   _____ I didn't initialize the time (I didn't write t=0 before the loop)
   _____ I didn't use tabulation to write the statements in the loop
   _____ I had some misspellings when writing the while loop
   _____ I deleted things that I shouldn't have done
   _____ I didn't read the instructions first
   _____ Other. Clarify below or write evaluating comments on the next page:

3. The symbol used to write comments in GlowScript 2.7 VPhyton is … (circle one)

% # () // $ @ &

4. I found the comments useful. (On a scale of 1-7, where by 7 you totally agree with the statement, circle the one applicable to you):

1 2 3 4 5 6 7

5. Other things that can help me in writing/understanding the program/lesson:

   _____ Instructions & programs organized and printed on paper
   _____ Instructions & programs organized in a course online
   _____ Other. Clarify below or write improvement comments on the next page:

Additional comments from points 2 and 5 (evaluation and improvements):

Thank you for your participation in this project. November 2018 – Ciprian Sima

Appendix 2. Results from the individual student survey

B1. Results from the first Programming lesson

B.1.1 Results from the hours spend in front of a screen in a week.

(a) Drawing

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	0	10	76.9	76.9	76.9
	1-3	3	23.1	23.1	100.0
	Total	13	100.0	100.0	

(b) E-mail

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	0	10	76.9	76.9	76.9
	1-3	3	23.1	23.1	100.0
	Total	13	100.0	100.0	

(c) Games

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	0	4	30.8	30.8	30.8
	1-3	2	15.4	15.4	46.2
	3-6	2	15.4	15.4	61.5
	7+	5	38.5	38.5	100.0
	Total	13	100.0	100.0	

(d) Photo editing

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	0	12	92.3	92.3	92.3
	1-3	1	7.7	7.7	100.0
	Total	13	100.0	100.0	

(d) Programming

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	0	11	84.6	84.6	84.6
	1-3	1	7.7	7.7	92.3
	3-6	1	7.7	7.7	100.0
	Total	13	100.0	100.0	

(f) Searching

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	1-3	10	76.9	76.9	76.9
	3-6	3	23.1	23.1	100.0
	Total	13	100.0	100.0	

(g) Simulations

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	0	12	92.3	92.3	92.3
	1-3	1	7.7	7.7	100.0
	Total	13	100.0	100.0	

(h) Spreadsheet

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	0	12	92.3	92.3	92.3
	1-3	1	7.7	7.7	100.0
	Total	13	100.0	100.0	

(i) Surfing social media

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	1-3	3	23.1	23.1	23.1
	3-6	7	53.8	53.8	76.9
	7+	3	23.1	23.1	100.0
	Total	13	100.0	100.0	

(j) Surfing video

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	1-3	2	15.4	15.4	15.4
	3-6	5	38.5	38.5	53.8
	7+	6	46.2	46.2	100.0
	Total	13	100.0	100.0	

(k) Surfing other content

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	0	3	23.1	23.1	23.1
	1-3	8	61.5	61.5	84.6
	3-6	2	15.4	15.4	100.0
	Total	13	100.0	100.0	

(l) Word processing

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	0	2	15.4	15.4	15.4
	1-3	4	30.8	30.8	46.2
	3-6	7	53.8	53.8	100.0
	Total	13	100.0	100.0	

B.1.2 Students desire to increase the use of technology by developing digital resources.

Descriptive Statistics

	N	Sum	Mean	Std. Deviation
Conduct research via the Internet	13	5	.38	.506
Create apps/simulations using Programming	13	10	.77	.439
Create documents	13	4	.31	.480
Create offline presentations	13	3	.23	.439
Create websites	13	8	.62	.506
Valid N (listwise)	13			

B.1.3 First Programming lesson helped understanding the Physics lesson

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
Programming lesson helped understand Physics (Scale 1-7 and 7 = “strongly agree”)	13	2	7	5.23	1.481
Valid N (listwise)	13				

B.1.4 Completed tasks

Managed to finish task

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	all	12	92.3	92.3	92.3
	almost all	1	7.7	7.7	100.0
	Total	13	100.0	100.0	

B.1.5 Impressions about the lesson

General impression of the lesson

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Organized	8	61.5	66.7	66.7
	Chaotic but rewarding	4	30.8	33.3	100.0
	Total	12	92.3	100.0	
Missing	No answer selected	1	7.7		
Total		13	100.0		

B.2 Results from the second Programming lesson

B.2.1 Visual outcome made the goal clear (on a scale from 1 to 5, 5 = "totally agree")

Visual output displayed

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	So and so	1	8.3	8.3	8.3
	Agree	2	16.7	16.7	25.0
	Strongly agree	9	75.0	75.0	100.0
	Total	12	100.0	100.0	

B.2.1 Fractional number of students that have encountered different problems.

	N	Mean	Std. Deviation
C1. Didn't initialize the time	12	.0833	.28868
C2. Didn't use tabulation to write the statements in the loop	12	.0833	.28868
C3. Misspellings	12	.1667	.38925
C4. Deleted needed code	12	.1667	.38925
C5. Didn't read the instructions	12	.3333	.49237
Valid N (listwise)	12		

B.2.3 Percentage of students that identified the correct symbol used for writing comments in GlowScript 2.7 VPython

	N	Mean (%)	Std. Deviation
Selected correct symbol for writing comments	12	91.67	.28868
Valid N (listwise)	12		

B.2.4 Comments useful (on a scale from 1 to 7, 7 = "strongly agree")

Descriptive Statistics

	N	Range	Minimum	Maximum	Mean	Std. Deviation	Variance
Benefits of writing comments	12	3.00	4.00	7.00	5.8333	1.11464	1.242
Valid N (listwise)	12						

B.2.5 Material sources that are needed by students

	Mean
Instructions and programs on paper	.2500
Instructions and programs in an online course	.4167
Other	.3333

Appendix 3 Participant Consent Form and information sheet

Title of the action research project: How can Programming be implemented in Physics lesson to upper secondary school students?

Main investigator and contact details:

Ciprian Sima
Oslo International School
Gamle Ringeriksvei 53,
1357 Bekkestua
Ciprian.sima@oslois.no
Tel: +47 – 456 18 825

I agree to take part in the above research. I have been informed about the project and I have read the Participant Information sheet which is attached to this form.

I have been informed that the research will be kept anonymous and confidential as my name will not be collected on any form and will not appear in any processing of the collected data.

I can ask questions during the case study and I can request by e-mail to be removed as participant at any time during the project.

Data Protection: I agree to the processing of data for purposes connected with this research project as it was presented to me. I give permission for the researcher to use and publish data gathered, but no personal data that can identify me personally will appear in any collection or processing of data.

Name of participant
(print)………………………………………………………………………………
Signed………………..………………………………..Date……………………

Name of parent/guardian
(print)……………………………………………………………………………
Signed………………..………………………………..Date……………...

YOU WILL BE GIVEN A COPY OF THIS FORM TO KEEP

Section A: The Research Project

The title of this research project is: How can Programming be implemented in Physics lesson for upper secondary school students?

Observation:

- My secondary school Physics students are users of technology.
- Technology is widely used in Science. In my class we do hands on experiments and/or simulations. Students collect and process data using technology. If there is a lack of equipment or the need to visualize concepts in Physics, simulations can and are used during the lessons. There are many simulations Physics teachers and students use, both online (e.g. Phet – Physics simulations) or offline (e.g. Yenka Physics).

The purpose of the study is to explore ways in which Programming can be implemented in Physics lessons so that students create simulations and are not simply users of simulations. The researcher intends to research his practice and provide a guide in implementing Programming in the Physics lessons.

For further information please contact:

Ciprian Sima

Oslo International School

Gamle Ringeriksvei 53,

1357 Bekkestua

ciprian.sima@oslois.no

Tel: +47 – 456 13 825

Section B: Your Participation in the Research Project

If you agree to participate in the project you will be involved in the following methods of collection of data:

1. Completion of questionnaire
2. Participate in group discussions
3. Give and receive feedback

Data collected during the research will be kept securely and then destroyed when the research project is complete. Your participation in the research will be kept anonymous and confidential as you will not be named at any stage and your name will not be asked in the questionnaire.

YOU WILL BE GIVEN A COPY OF THIS INFORMATION SHEET TO KEEP WHEN THE CONSENT FORM IS HANDED OUT FOR YOU AND YOUR GUARDIANS TO SIGN.
