

TARTU UNIVERSITY
FACULTY OF SOCIAL SCIENCES

NARVA COLLEGE
INFORMATION TECHNOLOGY SYSTEMS DEVELOPMENT

Kristofer Velbaum
PRACTICAL PACKAGE IMPROVEMENT AND DEVELOPMENT FOR THE
P2NC.01.084 VISUAL PROGRAMMING COURSE

Diploma thesis

Supervisor *Dr. Yar Muhammad*

NARVA 2019

Olen koostanud töö iseseisvalt. Kõik töö koostamisel kasutatud teiste autorite tööd, põhimõttelised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on viidatud.

Kristofer Velbaum

.....

Töö autori allkiri ja kuupäev

TABLE OF CONTENTS

ABSTRACT	1
INTRODUCTION	3
ABOUT THE COURSE.....	3
ABOUT THE PRACTICAL PACKAGE	4
BACKGROUND	5
SIMILAR COURSES.....	5
TEACHING METHODS	10
LEARNING METHODS	15
METHODOLOGY.....	16
IMPLEMENTATION OF PACKAGE.....	18
Exercise 1:	18
Exercise 2:	19
Exercise 3:	20
Exercise 4:	22
Exercise 5:	24
Assignment 1:.....	25
Assignment 2:.....	27
Assignment 3:.....	29
FEEDBACK.....	31
QUESTIONNAIRE.....	31
CONCLUSION	35
FUTURE IMPROVEMENT	36
REFERENCES.....	37
APPENDICES	39
APPENDIX 1. Weekly course plan	39
APPENDIX 2. Questionnaire link.....	40
LICENSE	41

ABSTRACT

English:

The topic of this thesis is to deliver a practical package for “Visual Programming (P2NC.01.084)” course taught in Tartu University Narva College. The aim of the package is to provide students with the opportunity to practice the skills gained in lectures by creating certain small applications. The package is aimed at bachelors’ students who don’t have a huge experience in programming. It contains a set of exercises and assignments. Literature survey has been done to understand how other universities are teaching similar courses, what content they are using. Four different courses from different universities were studied, such as City, University of London, Malmö, University of Sweden, University of Washington, USA and Tallinn University of Technology. Different teaching methodologies were discussed in the thesis. In this course, Problem Based Methodology is already used. After the course completion, the feedback questionnaire has been created and students were asked to answer it:

- Exercises and assignments are corresponding to the objectives of the course – 81.8% agreed, 18.2% somewhat agreed.
- Exercises and assignments are easy to understand – 54.5% agreed, 27.3% somewhat agreed, 18.2% somewhat disagreed.
- Exercises and assignments develop programming skills – 100% agreed.
- Exercises and assignments evaluation policy was transparent – 100% agreed.

According to gathered data, more than 70% of students successfully completed the course.

Estonian:

Selle lõputöö teema on pakkuda praktilise paketi kursusele “Visuaalne Programmeerimine (P2NC.01.084)” mida õpetatakse Tartu Ülikooli Narva kolledžis. Paketi eesmärk on anda õpilastele võimaluse harjutada loengutes omandatud oskusi, luues teatud väikesed rakendused. Pakett on suunatud bakalaureuseõppe üliõpilastele, kellel ei ole suurt kogemust programmeerimisel. See sisaldab mitmeid harjutusi ja ülesandeid. Kirjandusuuring oli tehtud selleks, et saada aru, kuidas teised ülikoolid sarnaseid kursusi õpetavad, millist sisu nad kasutavad. Uuriti nelja erinevat kursust erinevatest ülikoolidest, näiteks City, Londoni Ülikool, Malmö, Rootsi Ülikool, Washingtoni ülikool, USA ja Tallinna Tehnikaülikool. Selles lõputöös arutati erinevaid õpetamismetoodikaid. Selles kursuses kasutatakse juba probleemipõhist metoodikat. Pärast kursuse lõpetamist oli loodud tagasisideküsimustik ja õpilastel paluti sellele vastata:

- harjutused ja ülesanded vastavad kursuseeesmärkidele – 81,8% on nõus, 18,2% mõnevõrra on nõus.
- harjutusi ja ülesandeid on lihtne mõista – 54,5% on nõus, 27,3% mõnevõrra on nõus, 18,2% mõnevõrra ei ole nõus.
- harjutused ja ülesanded arendavad programmeerimisoskusi – 100% on nõus.
- harjutuste ja ülesannete hindamise poliitika oliläbipaistev – 100% on nõus.

Kogutud andmete kohaselt lõpetas kursuse edukalt üle 70% õpilastest.

INTRODUCTION

ABOUT THE COURSE

Software has always been an important part of the technology community, because as long as the needs of humanity grow, the appropriate tools and software for these tools must be developed in order to achieve certain goals. Every day the number of existing software increases, but the demand for new software also increases. To meet the demand, we need programmers who can create this software. And this course is intended for their training.

The objectives of this course are:

- To familiarize students with basics about .Net Framework
- To introduce the Integrated Development Environment called Visual Studio
- To guide students through different Objective-Oriented-Programming concepts with help of C# programming language.

At the end of this course, students should be able to:

- Use basic programming data types in C#
- Use decision statements and repetition (loops)
- Perform Object-Oriented-Programming in C#
- Create and use methods
- Use complex data structures and collections
- Work with Generics
- Handle events and write delegates
- Build the interface using Visual Studio tools

To achieve these goals, we need both theoretical and practical parts. The theory will be covered with lectures, which are conducted by the teacher. The practice will be provided by practical package. And this is where my thesis comes into.

ABOUT THE PRACTICAL PACKAGE

The goal of this thesis is to offer the complete practical package such as:

- Development of the Learning Scenario (i.e. what shall student learn, what will they do in the classes/labs, hands-on, exercises, assignments)
- Development of the learning material for the students to use
- Development of the practical problems and corresponding solutions

Practical package contains 5(five) Exercises and 3(three) Assignments. Detailed description is available in chapter “Practical Package”.

BACKGROUND

SIMILAR COURSES

Visual Studio development environment and the C # programming language are widespread, so the ability to use them is necessary for every professional programmer. For this reason, there are many courses dedicated to them. The following four courses can serve as examples.

.NET Object–Oriented Programming Using C# course from **City, University of London**

This course is a short evening course that teaches students how to create new Windows applications using C# programming language. This is the ideal course for those looking to create an application with the connectivity, look and feel of Microsoft Windows. (City, 2019)

Although it is aimed at students with some knowledge of procedural programming – for example 20 hours tuition in VB or VB.NET or C/C++. If a student has no programming experience, he should first attend **Java: Object-oriented programming with Java | Part 1** course. The syntactical similarities between C# and Java make an introductory Java course more than adequate for a course of study in C#. (City, 2019)

Here is the list of themes that this course covers:

- Introduction to C#. Why use C#?, Language comparison with C/C++, Java, VB.NET
- Overview of the .NET framework; using Visual Studio.NET
- C# fundamentals: Types, Variables, Constants, Expressions, Statements, Operators, Namespaces
- Classes and Objects: Defining Classes, Creating and Destroying Objects, Passing Parameters, Overloading, Encapsulation.
- Inheritance and Polymorphism: Specialization and Generalization, Inheritance, Polymorphism, Abstract Classes

- Interfaces: Implementation, Accessing Interface Methods, Overriding Interface Implementations
- Handling Exceptions: Throwing and Catching Exceptions, Exception Objects, Custom Exceptions, Rethrowing Exceptions
- Windows Programming: Building Windows Applications, Creating Simple Windows Forms and Windows Forms Applications.
- Text handling applications: Reading and Writing files, string manipulation, file management

(City, 2019)

By the end of the course, students are supposed to be able to:

- Write object-oriented programs in C#
- Use Visual Studio .NET to create GUI Applications (Rapid Application Development)
- Use object-oriented methods to write more robust and more reusable applications
- Gain familiarity with the .NET framework and component-based programming methods.

(City, 2019)

Programming in C# course from Malmö, University of Sweden

Course Code: DA204B revision 1.2

Level of specialization: G1N

The aim of the course is for the students to develop their programming skills to be able to develop structured Windows applications using the C# language and principles of object-oriented programming. Students should gain theoretical and practical knowledge of general programming techniques. (Malmö, 2019)

This course is also suitable for students with prior programming experience who wish to strengthen their knowledge in object-oriented programming with Windows. (Malmö, 2019)

The course covers the following main topics:

- Object-orientation: class and encapsulation

- Variables, operators, struct and enum
- Sequence, selection and iteration
- Arrays and an introduction to collections
- Graphical user interface
- Documentation and structuring of source code

On completion of the course the student should be able to:

- show knowledge and understanding of the fundamental principles of the object-oriented programming
- show knowledge and understanding of the fundamentals of writing applications to solve problems
- demonstrate fundamental knowledge and understanding of the general structure of the C# programming language
- show skills in writing simple but well-documented and structured Windows applications with graphical user interface
- show skills in implementing the basic concepts of object-orientation with an in-depth focus on Encapsulation

(Malmö, 2019)

The course consists of a theoretical and a practical part, including several programming exercises and assignments. It contains a number of mandatory assignments, quizzes and a project work (for higher grades). (Malmö, 2019)

***Distance Education:** The education is offered exclusively online; attendance or other physical presence is neither planned nor required. Students are expected to have access to a PC and Internet.* (Malmö, 2019)

Foundations of Programming (C# & .NET) course from University of Washington

In this course, students will learn how to write software the Microsoft way. They will explore and practice basic programming concepts using C#. They will explore Microsoft Visual Studio and the .NET Framework, and become comfortable with Visual Studio, solutions and projects. Students will also learn how to use Visual Studio's built-in debugger and how to think like a programmer. The course is designed for anyone interested in learning how to code in C#. (University of Washington, 2019)

What students will learn:

- C# structure: syntax, grammar, rules and keywords
- Techniques for breaking down a problem and debugging your solution
- Built-in tools that make C# programming easier

(University of Washington, 2019)

Hands-on experience:

- Code during class and get constructive feedback on solutions
- Complete a final real-life coding project using all the learned skills

(University of Washington, 2019)

Advanced Course in Programming I course from **Tallinn University of Technology**

Course code: IAG0010

The primary goals of the course are:

- Explain the basic operating principles of Windows as one of the most important modern operating systems.
- Explain the fundamental principles and problems of programming in Windows environment.
- Explain the fundamental paradigm of object-oriented programming.
- Show how to implement the object-oriented programming paradigm in C++.
- Give a comprehensive overview of the programming technologies widely used in modern software industry:
 - o Microsoft .NET together with C# and C++/CLI
 - o Sun's Java.
- Enhance the students' practical programming skills.

(TTU, 2019)

After successfully completing this course, the student:

- has comprehensive overview about how Windows is actually operating
- is acquainted with the fundamental paradigm of object-oriented programming;
- has comprehensive knowledge about C++;
- is able within reasonable time to find and use necessary supplementary material (like help files) to design, code and debug:
 - Windows console applications in C++;
 - Windows .NET windows-driven applications in C# or C++/CLI;
 - Java windows-driven applications.

(TTU, 2019)

As we can see, all the courses are somewhat similar with our course, they have almost the same themes and goals. The differences are:

- The course from Malmö University, as mentioned above, is exclusively online.
- The course from Tallinn University of Technology contains not only C# and .NET, but also C++ and Java.

TEACHING METHODS

- **Case-Based:** Using a case-based strategy involves students discussing particular situations that resemble or are typically real-world instances. This technique is student-centered with strong communication among the respondents as they construct their understanding and operate together as a team to examine the situation. The function of the instructor is that of a facilitator while the students collaboratively evaluate and solve issues and fix issues that do not have a single correct response. This technique gives students a appropriate chance to see theory in reality. Real world or genuine environments reveal students to viewpoints from various sources and see why individuals might want distinct results. Students can also see how distinct respondents will be affected by a choice, both favorably and negatively. Case-based technique needs students to evaluate information to achieve a result. Students can also exercise selecting suitable analytical methods, because many tasks are open-ended. Instructors who use case-based teaching claim that their students are more engaged, interested, and involved in the education process. This technique is used to improve analytical, communicative and cooperative abilities along with understanding of material. Students sort out factual information, apply analytical instruments, articulate problems, reflect on their appropriate experiences, and draw findings that they can connect to the new circumstances in effort to discover alternatives and achieve choices through debate. They obtain meaningful expertise in the system and create analytical, cooperative and interaction abilities. Many faculty also use case studies in their curriculum to teach content, connect students with real life data, or provide opportunities for students to put themselves in the shoes of the decision-maker. (Queens, 2019). Case Based method is depicted in Figure 1.

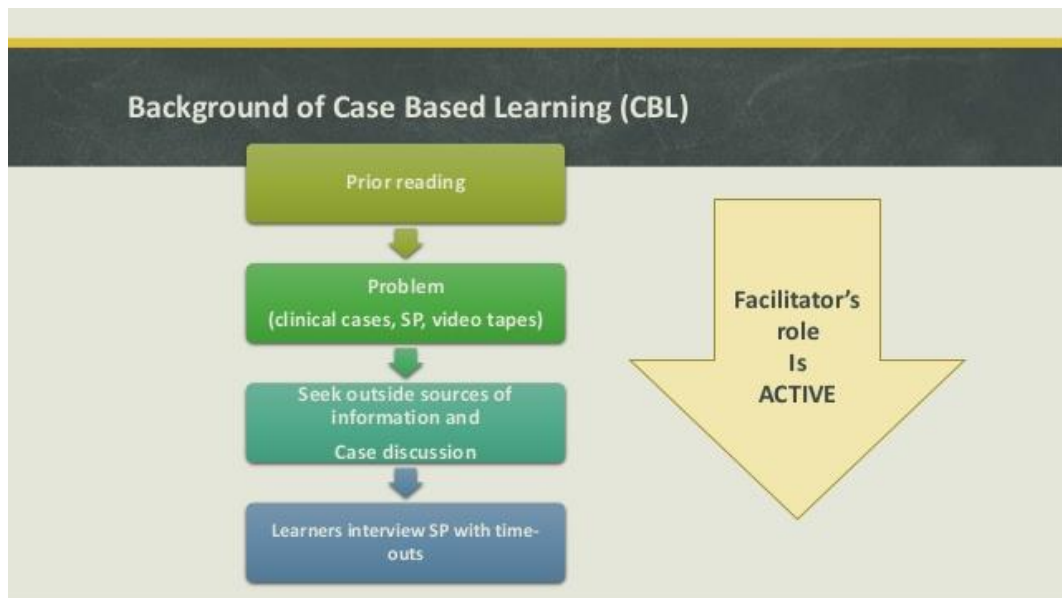


Figure 1. Case Based Learning. (Umbrin, 2014)

- **Field-Based:** In field-based learning, instruction is expanded to a location outside the class or workshop, introducing students to a real-world environment. Students learn through immediate communication with an environment that represents educational ideas rather than teaching through indirect setting presentations such as textbooks or conferences. Field-based learning can serve a variety of teaching purposes and objectives as students are supplied with a view of products, items or events that are not available or cannot be assessed in other places. Field-based learning is generally chosen because experience: provides an opportunity to present materials, objects or phenomena that are not otherwise accessible to students in a way that allows direct contact and interaction; provides students with an opportunity to practice skills or techniques that can not be carried out elsewhere; stimulates a greater knowledge and strengthening of earlier studied content ; stimulates recognition, appreciation or interest of the visited setting. (Queens, 2019) Field Based method is depicted in Figure 2.

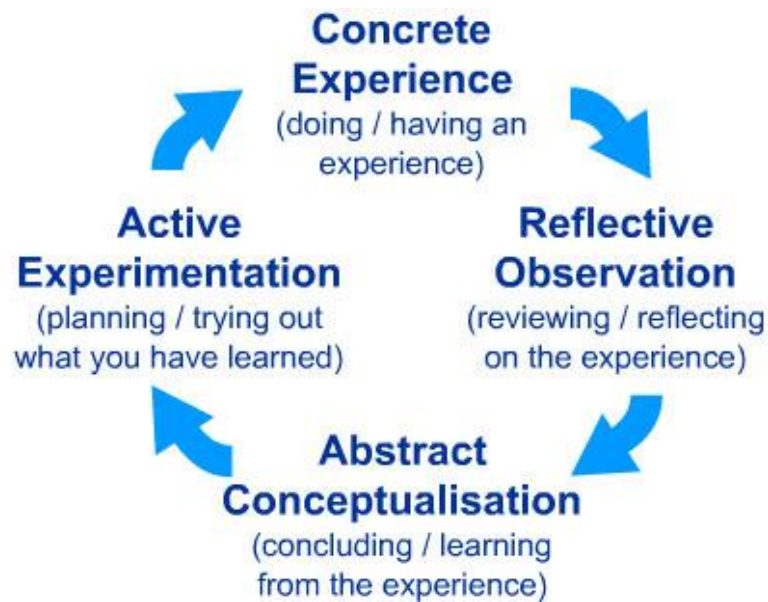


Figure 2. Field Based Learning. (Kolb)

- **Inquiry-Based** Based on the idea that education starts with the student's curiosity, investigation in the classroom places the accountability for teaching and promotes them to come to an awareness of ideas on their own. This technique follows the four fundamental phases of self-directed teaching where students take more responsibility for: determining what they need to know; defining resources and how to learn from them more effectively; using resources and reporting on their teaching; evaluating their advancement in teaching. Most courses will build on one or more of these phases, while there will be all four of these components in a extensive senior inquiry course. Students will take the action and be mainly accountable for seeing that they effectively finish their teaching in a specified region. Generally, students draft a "learning contract" that indicates what they will learn, the assets, and how their learning will be proved and evaluated. Throughout the course students then perform their learning contract with the teacher submitting a grade upon completion of the contract. Encouraging students to be self-directed is a critical ability that students need to gain in order to be effective in post-secondary education and professions. This technique promotes students to develop appropriate study abilities, such as defining teaching objectives and resources that are then applied to fulfill deadlines. (Queens, 2019) Inquiry Based method is depicted in Figure 3.

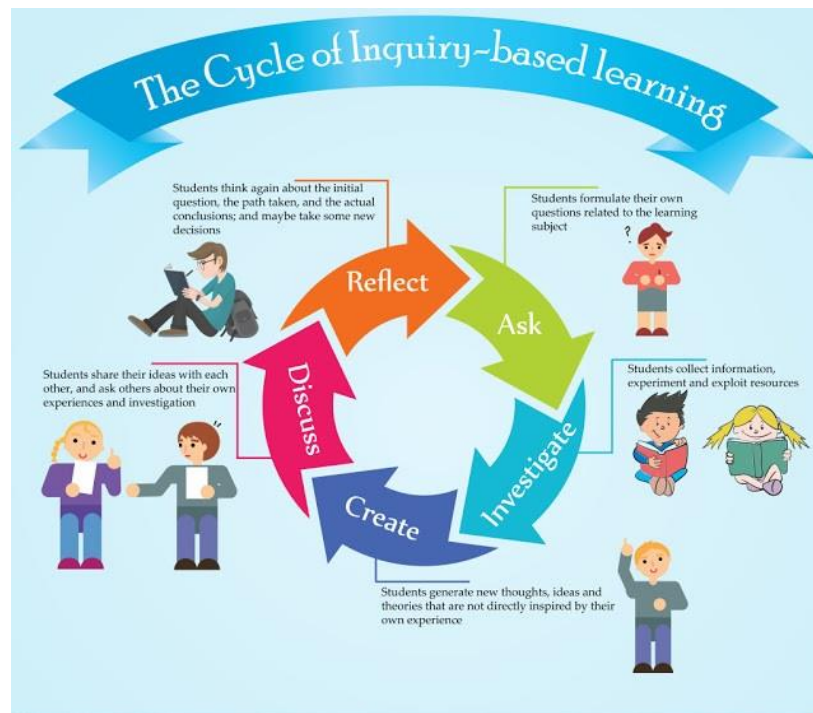


Figure 3. *Inquiry Based Learning.* (Pavlova, 2015)

- Problem-Based:** This technique is focused on the chaotic, complicated issues experienced in the real world as a basis for studying and incorporating and arranging acquired data in ways that will guarantee its recall and implementation to potential issues. Problems are presented at the beginning of the subject before some of the appropriate information has been learned. By actively engaging with the issue, students acquire abilities in discovering data, recognizing what data they still need, and sources, where they can find that data. Students can connect what they learn during the lectures with their own life and significant problems in their world. In PBL, students meet an issue and try to fix it with data they already possess enabling them to understand what they already understand. They also recognize what they need to know to better comprehend the issue and how to solve it. Once they have worked with the issue as much as feasible and recognized what they need to know, the students participate in self-directed study to explore the data required to find and use a range of data materials (books, publications, records, internet data, and a range of individuals with suitable fields of knowledge). In this manner, training is personalized to the individual's requirements and learning styles. The students then move to the issue and add what they have learned to their job with the issue in order to better comprehend and solve the issue. After finishing their problem work, the students evaluate themselves and each other to improve abilities in self-assessment and peer

positive evaluation. Self-assessment is an important ability for efficient autonomous learning. The teacher's duty in PBL is to provide the teaching tools and guidelines that promote learning. The teacher's main function in PBL is that of a facilitator or instructional manager (often referred to as a "tutor" in PBL jargon) who guides the students in the PBL process. As students become more skilled in the PBL method, the teacher becomes less involved. (Queens, 2019) Problem Based method is depicted in Figure 4.

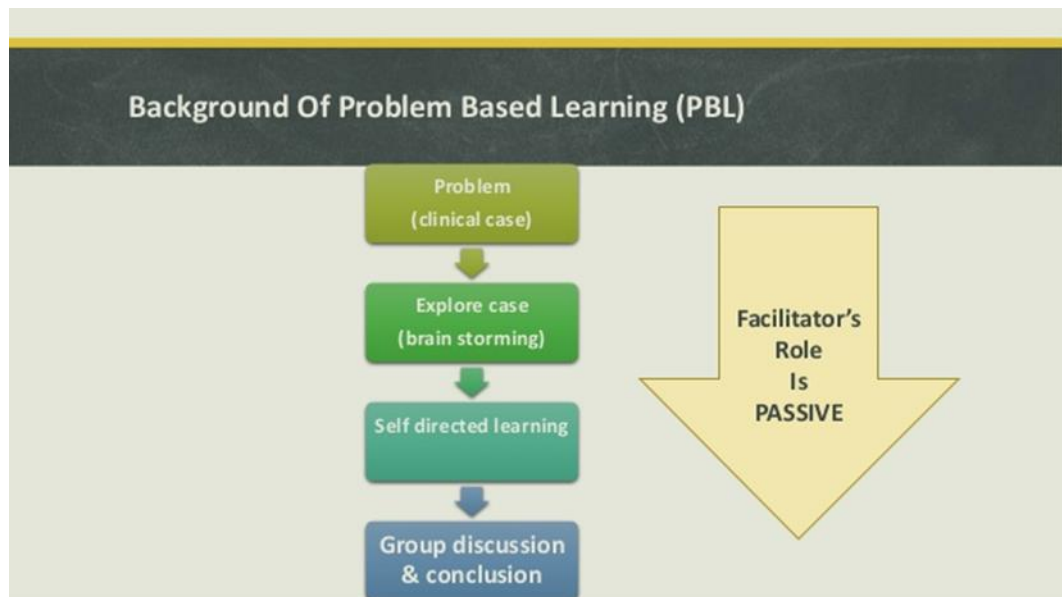


Figure 4. Problem Based Learning. (Umbrin, 2014)

LEARNING METHODS

Studies indicate that variable research techniques and equipment will improve data retention and recall and enhance teaching experience. The "learning pyramid" depicted in Figure 5, sometimes referred to as the "cone of learning," created by the National Training Laboratory, indicates that most students only remember about 10 percent of what they read from textbooks, but maintain almost 90 percent of what they learn by educating others. The Learning Pyramid system indicates that some learning styles are more efficient than others and that variable study techniques will contribute to greater learning and longer-term retention. (Education Corner, 2019)

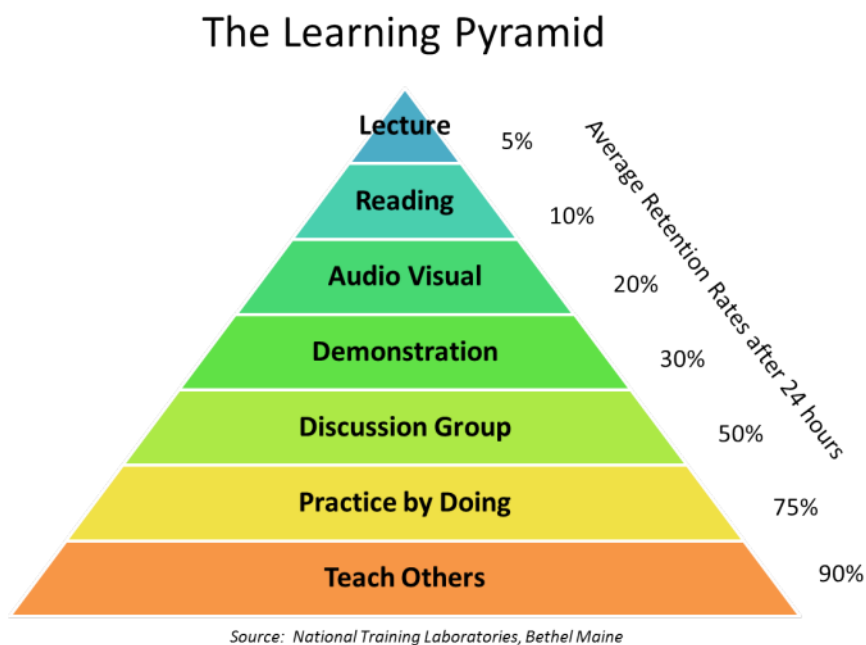


Figure 5. Learning Pyramid. (BK101)

Although in our course we do not use the "Teaching Others" methodology, we still can use "Practice by doing". Of all available methods, this is the most effective way to consolidate the acquired skills. This implies that more practice = more skills.

METHODOLOGY

The universe of today carries with it a fast burst of readily available knowledge. Graduates today need to be self-directed and have lifelong learning abilities. They need to be critical thinkers, problem solvers, and analytics in their strategy. The interdisciplinary nature of the job implies that they need to be prepared to acquire knowledge and abilities from a variety of fields as well as have the interpersonal abilities to be an efficient team member. (Queens, 2019)

Problem-based teaching operations are intended to create transferable abilities and qualities along with the suitable discipline-specific understanding. Transferable skills / attributes are components of the requirements of the degree level that reflect the expected results for college education and are being written into program curriculum. Problem-based learning difficulties make students to acquire the capacity to think critically, analyze issues, discover and use suitable study tools. (Queens, 2019)

A student-centered technique of education; PBL students are gradually assigning more and more accountability for their own education and becoming increasingly autonomous of the teacher for their education. (Queens, 2019)

Problem-based learning courses mainly focus on teaching students through genuine problem scenarios. By generating these circumstances, the course simulates the skill practice and the complicated problems surrounding it. Material is naturally integrated in issues. Through thoroughly conceived issue situations, suitable material is chosen and placed at genuine places throughout the process and problem where students can find it. (Stepien, Gallagher, 1993)

So, for this course, the Problem-Based teaching method was used by teacher. Due to the fact that, according to the Learning Pyramid, the most effective of available learning methods is to practice by doing, the methodology for this course was developed. It consists of the following steps:

- The teacher introduces the topics during lectures
- At the end of every lecture, a Hands-On is given to students in Labs to understand the topics.
- By completing the exercises, students use the knowledge which s/he learned during the lectures.

- Assignments are given to students. To complete it, they have to not only use the acquired knowledge, but also search for necessary information and learn by themselves.

The used methodology is depicted in Figure 6.

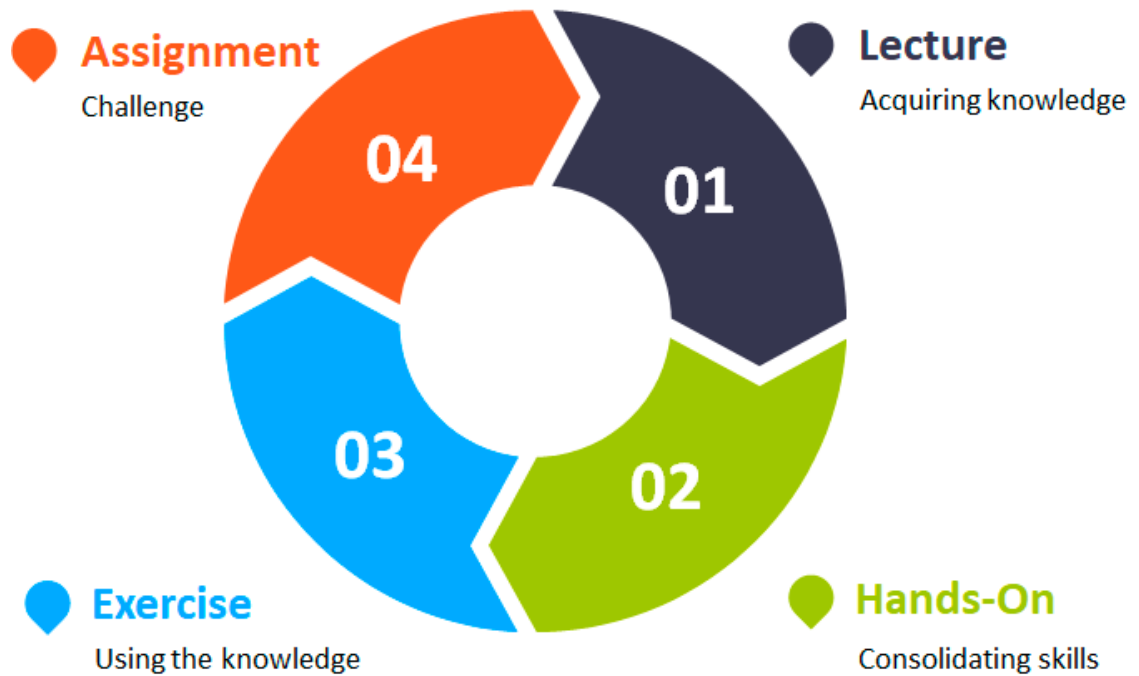


Figure 6. The used methodology.

IMPLEMENTATION OF PACKAGE

As mentioned earlier, practical package contains 5(five) Exercises and 3(three) Assignments. Each Exercise contains 2 Problems, each Problem gives 2 marks. Each Assignment gives 8 marks. $2*2*5 + 8*3 = 44$

Exercise 1:

Problem # 1: Write a program which helps to plan wallpaper budget. The program takes 3 arguments as an input: wall width (accuracy meter), wall height (accuracy meter) and price per square meter (0.00). If client buys more than 10 square meters, he will get 2 euros' discount, and if he buys 20 or more square meters, he will get 4 euros' discount. Implement two additional classes, one for calculating price and second to apply the discount. All code should not contain IF statement. Use only ? : Operator (ternary operator)

Example I/O:

Wall width > 4

Wall height > 2

m2 price (in format 0.00) > 2.3

Total price is 16 euros.

Problem # 2: Write a Console Application which asks for room's parameters like height, width and depth from user and calculates the floor or ceiling area, floor and ceiling total area, longest wall area, shortest wall area, room total area and room volume. For each operation write own method (it's good for code reusing) and try to implement methods with reusing already implemented methods.

Example I/O:

Enter room width: 5

Enter room height: 2.5

Enter room depth:3

Floor or ceiling area is 15 square meters.

Floor and ceiling area is 30 square meters.

Longest wall area is 12.5 square meters.

Shortest wall area is 7.5 square meters.

Room total area is 70 square meters.

Room total volume is 37.5 cubic meters.

Exercise 2:

Problem # 1. Create a console application that draws rectangles. Program takes width and height as an arguments and prints rectangle in console window. For horizontal lines use "--", for vertical use "|" and for empty space use " ". For printing without newline use `Console.Write(string)`.

HINT: Program uses nested for-cycles. Try this:

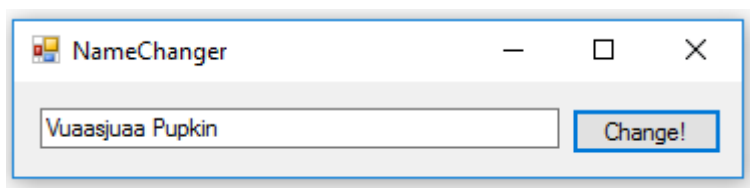
```
for (int i = 0; i <= 20; i++)  
{  
    for (int j = 0; j <= 10; j++)  
    {  
        Console.Write("++");  
    }  
    Console.Write("\n"); // new line  
}  
Console.ReadLine();
```



Problem # 2. Now try to do name changer. It has one text box and one button. User input should input his own name with last name and press the button. As a result the name in text box should change (some chars should replace others). Program logic should be based on for-cycle and conditional statements. What characters and how to change them is limited only by your fantasy.

HINT: Use "Length" property, for getting string length and "Equals" method for comparing chars: `currentName[i].Equals('a')`.

I typed in Textbox Vasja Pupkin, clicked on Change button and got the name changer result in Textbox Vuaasjuua Pupkin. The program replaced 'a' with 'uaa'.



Exercise 3:

Problem # 1. Let's write a GUI application that encode our passwords and store it in data.text. Application contains one textbox for Key and another textbox for data (both can contain only ASCII characters). Application also contains two buttons: "Load" and "Save", first button loads the data from data.txt (if it exists, if not exception need to be

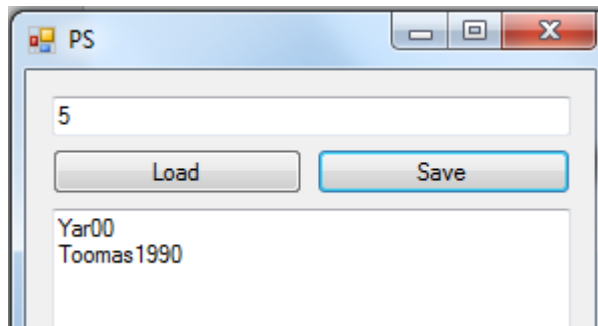
handled) and decode it with entered Key and "Save" encode data with entered Key and save it to data.txt.

Try to save with one key, and load with other. Look in data.txt contents.

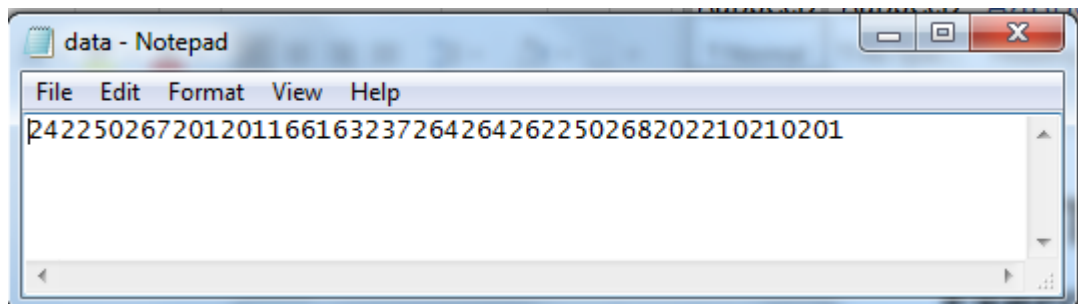
a) Try to use Cyrillic characters, what happens? Why? What we need to do?

ANSWERS: Cyrillic character not in ASCII table and utf-8 characters have another longer int representation. We must put exception handler in this code.

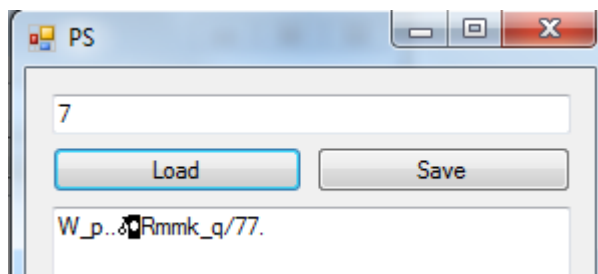
Note: How this application work – If I entered Key 5 and Password “Yar00 Toomas1990” and click on Save button.



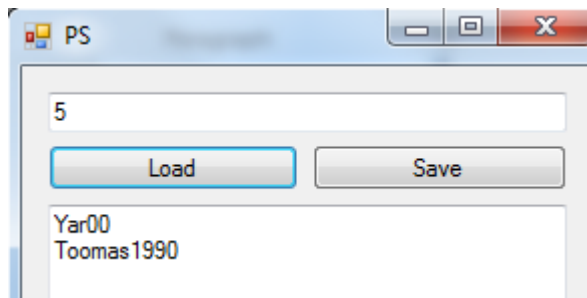
I opened the data.txt file how these passwords are encoded.



I ran the application again to put wrong Key “7” and click on Load button.

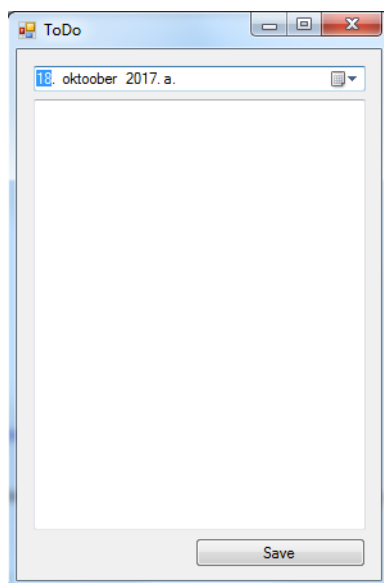


The file is not properly decoded because wrong Key is entered. Now, correct Key is entered Key “5” and click on Load button.



The proper data is encoded "Yar00 Toomas1990".

Problem # 2. Create simple GUI application that helps handle to-do list. In the application has three elements: textbox, datepicker and "Save" button. In the application user can choose date, write what to do and save it on this date. You need to get datepicker value, and convert it to file name like "DD-MM-YY.txt", then save data to it. Same process in reverse works if datepicker date was changed. Don't forget handle exceptions.



Exercise 4:

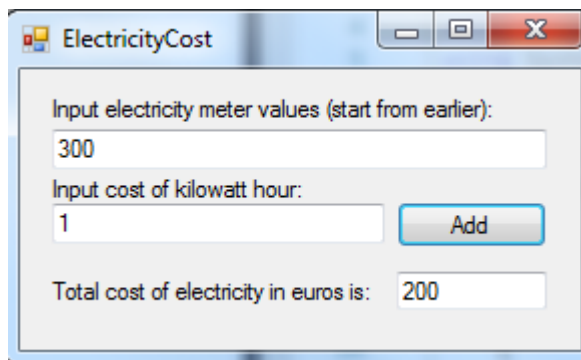
Problem # 1. Try to implement an Application for electricity cost calculation. The Application takes these inputs: electricity meter value, cost per kilowatt hour. User needs to add all months' electricity meters' values (starting from earliest) and press add button,

if needed, they can change the price per kilowatt. After every addition cost between last and new value is calculated.

For example:

first month my meter shows 100, and price per kilowatt is 1 euro, next month it shows 300, then $300 - 100 = 200$ kwh, $200 \text{ kwh} * 1 \text{ euro per kwh} = 200$ euros (add it to total cost), in next month my meter value is 700, then $700 - 300 = 400$ kwh, calculate cost and add it to total cost and so on.

Hints: Two constructors could be used one with optional string. Two methods could be declared for files read and write, and all other must work over that two. Another solution could be read content in to array and manipulate it as array.



Problem # 2. Create contacts book that can store up to 1000 contacts. Contact book must have 3 tabs: first for add contact (name, phone, email), second for search by name and third tabs shows all contacts. All contacts must be stored in file. All functionalities of the application need to be implemented in separated class named "ContactDB" (ContactDB contains following methods: addContact, getAllContacts, searchByName, and other), all object variables must be private and accessed over accessors. All files must store in same folder with application and do not need to search weird solutions to organize your path. For example, if you use method `System.IO.File.ReadAllLines(path)`, path must be for example "myfile.dat".

Form1

Add Search All Contacts

Name
Paul

Phone
47884598

E-mail
Paul@hotmail.ee

Add Contact

Form1

Add Search All Contacts

Insert name
Paul Kaas

Search by name

Paul Kaas 47884598 Paul@hotmail.ee

Form1

Add Search All Contacts

James Boot 5541104558
james.boot@gmail.com
Toomas Tamm 8998855
toomas.tamm@yahoo.com
Andrei Ivanov 4774488
andrei3667@hotmail.ee
Anton Krek 5547878 mymail@hotmail.ee
Yar 547554558 Yar@abc.com
Paul Kaas 47884598 Paul@hotmail.ee

Exercise 5:

Problem # 1. Create a simple program that calculates area. Use class name for calculation (Area) which will contain only constructor that takes two arguments. Calculation of result should be presented before "Area" and after that object should be disposed. In short, area should be displayed before object destroyed (removed from memory).

HINT: In Main method use "using" keyword and Area must inherit IDisposable.

Problem # 2. Write simple Application that calculate Car Ownership Cost.

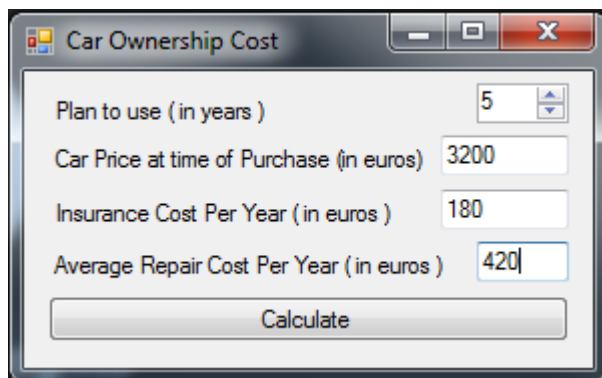
User has to put following input in order to calculate the Car's Ownership Cost:

- How many years plan to use the Car?
- How much to pay to purchase the Car?
- Average Insurance Cost per year.
- Average Repair Cost per year.

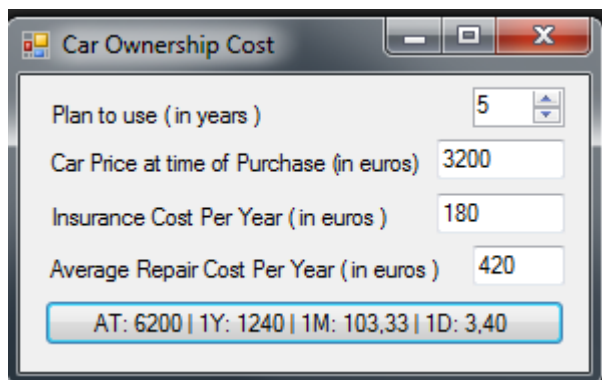
After pressing the "calculate" button, the result should be shown on the same button.

Results should be shown in this format: AT: 344 | 1Y: 100 | 1M: 50 | 1D: 5.

Note: Create a separate class for all calculations' functionality.



The screenshot shows a window titled "Car Ownership Cost" with four input fields and a "Calculate" button. The inputs are: "Plan to use (in years)" with a value of 5, "Car Price at time of Purchase (in euros)" with 3200, "Insurance Cost Per Year (in euros)" with 180, and "Average Repair Cost Per Year (in euros)" with 420.



The screenshot shows the same window after the calculation. The "Calculate" button now displays the results: "AT: 6200 | 1Y: 1240 | 1M: 103,33 | 1D: 3,40".

Assignment 1:

Whenever we write a code or play with database, some random datasets are needed for testing purpose. The random generated data should be unique and required in big amount. In order to generate such kind of data. It requires to develop a simple CSV generation software.

CSV is very simple (just use comma to separate the values) spreadsheet and data import/export format. CSV is supported by most of the systems as data export/import

format (we can export CSV in MySQL, CRM systems, Accounting Software etc). You can read more about CSV from this link: https://en.wikipedia.org/wiki/Comma-separated_values

First of all, it requires to create additional class in our project (Project -> Add Class...) and name it something like "RandomData". In this class we need to implement following methods:

- getName() - Generate random first and last name combination and return it as string.
- getDate() - Generates date (from 100 days ago to 20 years ago) and return it as string.
- getPhone() - Generates random phone number.
- getCity() - Return random city as string.

If you want as addition, you can implement method getEmail(), that generates random e-mail address and return it as string (here you can use random number cast as ASCII character and cycles).

To implement these methods, you need to use string arrays and random int generation method for selecting random index from array (use Next(range start, range end) method of class Random:

```
Random rnd = new Random();
```

The class with all methods are implemented, let's create our application elements.

Our GUI consists of 5 comboboxes for datatype selection, every combobox is related to one column of our CSV file. Application also has a field with rows count, preview screen and "Generate CSV" button.

We select type of random data for every column, enter the number of rows and press "Generate CSV" button, after that we see our CSV preview in preview window. At this moment CSV is saved to application folder (you can name it something like output.csv).

HINT:

As one of the solution, in this application is that you need run cycle for rows and run cycle for columns where you choose type with conditional statement. If statement is true, you add generated value to array (generate with class we made before). After all row cycles

complete, write this array (`System.IO.File.WriteAllLines()`) to file named `output.csv`. Catch exception if directory not writable (`UnauthorizedAccessException`). May be you need remove last comma from current row, you can use `String.Remove()` method, but if not any column is selected, you need to handle `ArgumentOutOfRangeException` exception.

Assignment 2:

The online translation services are very come and useful such as Google Translator and many more. Let's try to implement some very basic translator. For this purpose, we use freely available dictionary files in txt format. You can download these files (The Dictionary Files (February 19, 1999)) from this link <http://www.june29.com/idp/IDPfiles.html>. These files are not as good, but these files are enough to build a simple translator.

Now, we need to create an application for handling these files, this application must contain the fields for select dictionary file, textbox for enter word to translate, button "translate" and result window.

All these dictionary files must be stored beside our assembly if we have a folder named "Dictionaries". If the folder is empty, program says that "Dictionaries not found" and if something is wrong with directory (not readable, not found etc) program says "Cannot open directory".

If application is launched, select dictionary file from drop-down list, enter word and press "Translate", so we get translation in result window but if word is not found, then program will return to first match what contain this word in substring and if this is also not found, so then program will print "Not found" in result window.

HINT: Look in dictionary file structure, how words can be separated?

Translator

Select dictionary:

Word to translate:

Translation result:

Translator

Select dictionary:

Word to translate:

Translation result:

Translator

Select dictionary:

Word to translate:

Translation result:

Translator

Select dictionary:

Word to translate:

Translation result:

Assignment 3:

Maybe you heard something about Morse code (still used for emergency communication).

Read more about the Morse Code: https://en.wikipedia.org/wiki/Morse_code.

It does not seem very user friendly, the same robust alternative of Morse is "Tap Code" or "knock code". The "Tap" is very simple to learn and use. Read more about "Tap Code" https://en.wikipedia.org/wiki/Tap_code.

As you noticed that the general element of it is in matrix with characters and for taping character you first tap row and then column. But here we have one thing that can be reduced from the counting.

In the communication, we can optimize it for certain language. For optimization, we need to analyze some text in needed language, count what characters are most often used in particular language and put it upper side of our tap code table (Shown in below Figure).

Let's write a program that asks for file path (ex. book.txt), analyze it, count all characters and suggest tap table as given below for based on language on stats collected.

Use Generic Collection, Dictionary and LINQ. Program also have list of symbols to ignore inside (what tap table not need to contain such as "#!?:;[]()<>*. \n\t\r0123456789-").

More text should be better to optimize the tap table for specific language. The sample text (book.txt) is also uploaded. Following results are generated based on given text (book.txt).

Insert full path to book.txt file:
book.txt

Character statistics:

a - 957
e - 789
i - 626
s - 555
l - 414
u - 407
n - 393
t - 389
k - 364
m - 255
d - 219
r - 189
, - 166
v - 157
p - 156
o - 155
j - 140
h - 136
g - 122
ä - 115
b - 79
ö - 75
ü - 61
- - 14
ö - 8

Suggested tap code table:

	1	2	3	4	5
1	a	e	i	s	l
2	u	n	t	k	m
3	d	r	,	v	p
4	o	j	h	g	ä
5	b	ö	ü	-	ö

FEEDBACK

To evaluate the results of the practical package, the benefits that it brought to the students, I needed to get feedback from them. In my case, the standard course feedback in the educational information system (ÖIS) was not suitable because it contains a general opinion about the course, which includes an opinion about the lectures and the teacher, and, since my thesis is about exercises and assignments, I needed feedback only about them. In addition, I do not have access to this information.

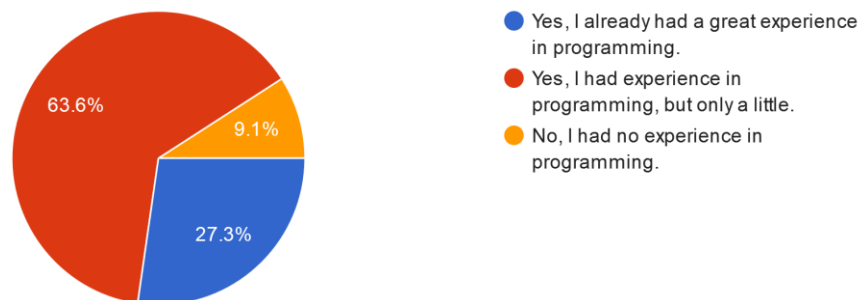
Therefore, I have compiled the necessary questions and upon the completion of a course students were given a questionnaire, which they had to respond. The questionnaire consisted of seven questions with response variants and a one free-form question. I got responses from 11 (eleven) students. Statistics are listed below.

QUESTIONNAIRE

Question 1:

Did you have any experience in programming before enrolling in a course?

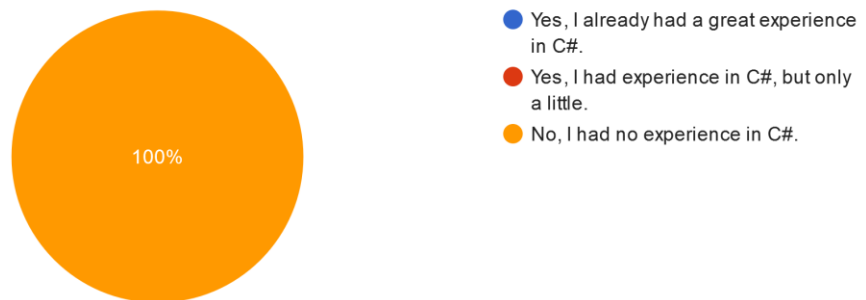
11 responses



Question 2:

Did you have experience specifically in C# before taking the course?

11 responses

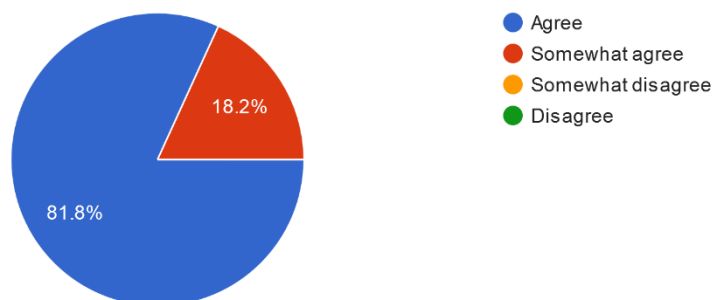


The purpose of questions 1 and 2 was to evaluate preliminary programming experience of a student. As we can see, most of students (10 out of 11) were already familiar with programming, but none of them were familiar with C#.

Question 3:

Exercises and Assignments correspond to the objectives of the course.

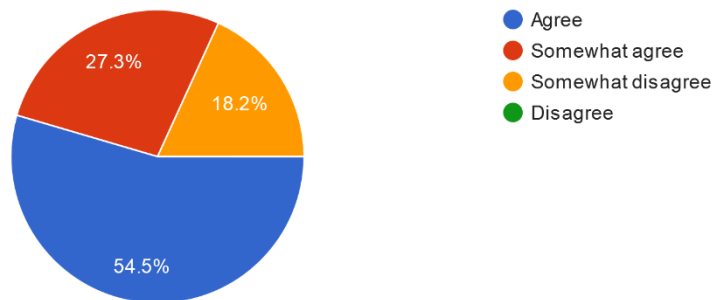
11 responses



Question 4:

Exercises and Assignments are easy to understand.

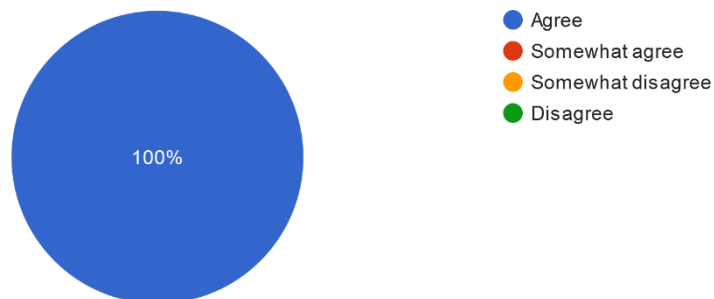
11 responses



Question 5:

Exercises and Assignments develop programming skills.

11 responses

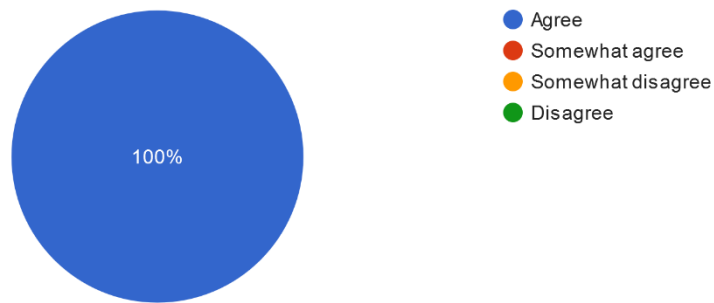


The purpose of Questions 3, 4 and 5 was to indicate what students thought about tasks and how they coped with them. As we can see, students almost completely agreed that tasks correspond the objectives of the course, bigger half of them (6 out of 11) completely understood the tasks and 5 of them had some difficulties. But everyone agreed that tasks improve programming skills.

Question 6:

Exercises and Assignments evaluation policy was transparent.

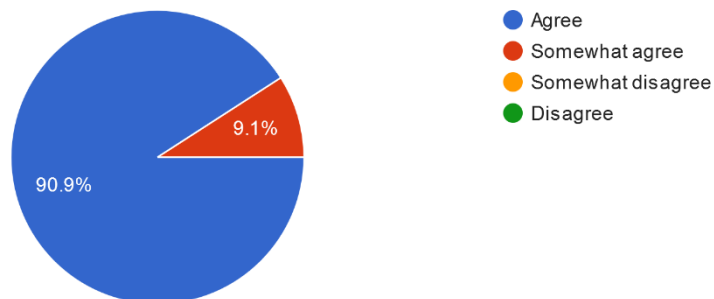
11 responses



Question 7:

Evaluation comments were useful.

11 responses



Questions 6 and 7 were aimed at understanding what students thought about evaluation. As we can see, all of them agreed that they understood the criteria for evaluating their work and most of them (10 out of 11) found the evaluation commentaries useful.

Question 8:

This question was not mandatory and could be answered in a free form. Just an opportunity for students to express their thoughts more precisely. So here are comments of 7 students.

Your comments on exercises and assignments. (NOT on lectures and/or teacher)

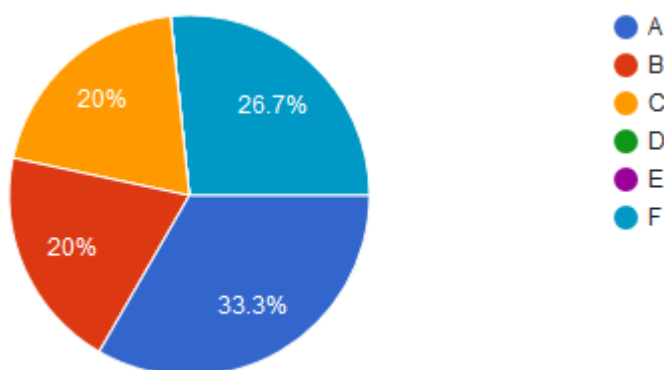
7 responses

Like
There was TOO MANY excercises and assignments
clear language and tasks
they were somewhat tricky but interesting to complete
Tasks were a little harder, than tasks during the lectures
very good and nice
very nice course, thanks

CONCLUSION

Obtained statistics show that my practical package was useful for students, that it raised their programming skills and, in general, most of them liked it.

Also, with the help of my supervisor, I obtained the statistics on final students' grades:



This indicates, that, out of 15 students, 5 got A (Excellent), 3 got B (Very Good) and 3 got C (Good). No one got D or E, but, unfortunately, 4 students Failed the course.

All in all, the course can be called successful.

FUTURE IMPROVEMENT

One of the possible improvements mentioned by the students was to add some Game Based Learning (teaching method that allows learners to explore different parts of games as a form of learning), because that way will add some entertainment element to the learning process, which can result in raising motivation and engagement.

REFERENCES

- Course OIS page 2017. Õppekava "Infotehnoloogiliste süsteemide arendus (136638)" sisu 2017/2018 sisseastunutele. Available at <https://ois2.ut.ee/#/courses/P2NC.01.084/version/sv-2018-autumn-fulltime-c5-en/details>, accessed May 18, 2019
- City, University of London 2019. Available at <https://www.city.ac.uk/study/courses/short-courses/courses/.net-object-oriented-programming-using-c>, accessed May 18, 2019
- Malmö University, 2019. Available at <https://edu.mah.se/en/course/da204b>, accessed May 18, 2019
- University of Washington. Available at <https://www.pce.uw.edu/courses/foundations-of-programming-c-sharp-and-dot-net>, accessed May 18, 2019
- Tallinn University of Technology, 2019. Available at <https://www.ttu.ee/asutused/tartu-kolledz/?id=154956&course=IAG0010&oview=1>, accessed May 18, 2019
- Queens University, Centre for Teaching and Learning, 2019. Case-Based Learning. Available at <https://www.queensu.ca/ctl/teaching-support/instructional-strategies/case-based-learning>, accessed May 20, 2019
- Queens University, Centre for Teaching and Learning, 2019. Field-Based Learning. Available at <https://www.queensu.ca/ctl/teaching-support/instructional-strategies/field-based-learning>, accessed May 20, 2019
- Queens University, Centre for Teaching and Learning, 2019. Inquiry-Based Learning. Available at <https://www.queensu.ca/ctl/teaching-support/instructional-strategies/inquiry-based-learning>, accessed May 20, 2019
- Queens University, Centre for Teaching and Learning, 2019. Problem-Based Learning. Available at <https://www.queensu.ca/ctl/teaching-support/instructional-strategies/problem-based-learning>, accessed May 20, 2019
- Education Corner. Learning Pyramid. Available at <https://www.educationcorner.com/the-learning-pyramid.html>, accessed May 20, 2019

Adapted from Stepien, W.J. and Gallagher, S.A. 1993. "Problem-based Learning: As Authentic as it Gets." Educational Leadership. 50(7) 25-8 and Barrows, H. (1985) Designing a Problem Based Curriculum for the Pre-Clinical Years.

Dr. Izza Umbrin, 2014, "Difference between Problem Based Learning (PBL) and Case Based Learning (CBL). Available at <https://www.slideshare.net/izzaumbrin/difference-between-problem-based-learning-pbl-and-case-based-learning-cbl>, accessed May 20, 2019

David Kolb, Experiential Learning Model. Available at <https://www.sandiego.edu/student-affairs/assessment/field-experience.php>, accessed May 20, 2019

Viktoriia Pavlova, 5 Phases of Inquiry-Based Learning Cycle. Available at <https://visual.ly/community/infographic/education/5-phases-inquiry-based-learning-cycle>, accessed May 20, 2019

BK101, "Basic Knowledge 101", Learning Methods - Thinking Styles - Teaching Methods. Learning Pyramid. Available at <https://www.basicknowledge101.com/subjects/learningstyles.html>, accessed May 20, 2019

Microsoft C# Guide. Available at <https://docs.microsoft.com/en-us/dotnet/csharp/>, accessed May 18, 2019

Tutorialspoint C# Guide. Available at <https://www.tutorialspoint.com/csharp/index.htm>, accessed May 18, 2019

Game-Based Learning. Available at <https://www.brighthubeducation.com/teaching-methods-tips/129304-advantages-of-game-based-learning/>, accessed May 18, 2019

APPENDICES

APPENDIX 1. Weekly course plan

P2NC.01.084 Visual Programming

About Visual Programming

This course is not a beginner course on Programming with C#. I am expecting basic knowledge of Object Oriented Programming. It is intended to provide an introduction to the C# language and the world of .NET programming.

Course Contents and Weekly Schedule Autumn 2017			
Week	Date	Contents	Deliverables
3	19.09.2018	Introduction about course, .Net Framework Overview, CLR, CTS, CLS, Stack vs Heap, Type Categories .Net Framework Architecture, Namespace, C# Language, Programming with C#, CLI	
	20.09.2018		
4	26.09.2018	C# Tool and IDEs, Introduction to object-orientation and C#, Visual Studio .Net Development Environment and Keyboard Shortcuts, Get start with visual studio and C#	
	27.09.2018		
5	03.10.2018	C# History, Datatype, Variables, Operations, and Expression Hands-On Programs 1-3	Exercise # 1 Deadline: 09.10.2018
	04.10.2018		
6	10.10.2018	Demo on Window Form, Hands-on Problems, Exercise Problems	
	11.10.2018		
7	17.10.2018	Making Decisions and Performing Iterations in C#	Exercise # 2 Deadline: 23.10.2018
	18.10.2018		
11	14.11.2018	Working with Methods and Handling Exceptions in C#	Exercise # 3 Deadline: 20.11.2018 Assignment # 1 Deadline: 25.11.2018
	15.11.2018		
12	21.11.2018	Introduction about Complex Data Structure, Working with Array, Working with Enumerations (enum), Working with Structures (struct)	
	22.11.2018		
13	28.11.2018	Introduction about Object-Oriented Programming (OOP) in C# (Class, Encapsulation, Static Methods and Static Class, Anonymous Class)	Exercise # 4 Deadline: 04.12.2018
	29.11.2018		
14	05.12.2018	Inheritance and Resource Management in C# (Garbage Collection) Collection in C#	Exercise # 5 Deadline: 09.12.2018 Assignment # 2 Deadline: 16.12.2018 Project Proposal: Deadline: 09.12.2018
	06.12.2018		
15	12.12.2018	Project Proposal/Mockup Presentations Project Discussion Exam Discussion/Revision	Assignment # 3 Deadline 19.12.2018 Mockup/Prototype Presentation (5 min)
	13.12.2018		

16	19.12.2018	Final Project Presentations	Project Presentation
	20.12.2018	Exam # 1 at 12.15 – 14.00	
	21.12.2018	Exam # 2 at 12.15 – 14.00	

Course Book:

Microsoft Visual C# 2013 Step by Step

Reference Book:

Professional C# 5.0 and .NET 4.5.1

C# Programming Guide: <https://msdn.microsoft.com/en-us/library/67ef8sbd.aspx>

C# Tutorials: [https://msdn.microsoft.com/en-us/library/aa288436\(v=vs.71\).aspx](https://msdn.microsoft.com/en-us/library/aa288436(v=vs.71).aspx)

C# Basic Tutorial: <http://www.tutorialspoint.com/index.htm>

Grading Points:

Activity	Description	Total Points
Exercises	5 Exercises * 4 points	20
Assignments	3 Assignments * 8 points	24
Project	5 Proposal + 5 Mockup Presentation + 30 Final Project	40
Final Exam		16
Bonus	Active Participation in Class and Exercises	10
Total		110

Deliverable Submission:

Assignments: 1 day late 50% penalty. After 1 day 100% penalty.

Exercises: No late submission is accepted. 100% penalty.

APPENDIX 2. Questionnaire link

<https://goo.gl/forms/Zeb9PfNfNfiE2zzI3>

LICENSE

Non-exclusive license to reproduce thesis and make thesis public

I, Kristofer Velbaum (date of birth: 10.09.1997),

1. herewith grant the Tartu University Narva College of a free permit (non-exclusive license) to:
 - 1.1. reproduce, for the purpose of preservation and making available to the public, including for addition to the DSpace digital archives until expiry of the term of validity of the copyright, and
 - 1.2. make available to the public via the web environment of the University of Tartu, including via the DSpace digital archives until expiry of the term of validity of the copyright, of my thesis

Title, “Practical package improvement and development for the P2NC.01.084 Visual Programming course”, supervised by *Dr. Yar Muhhamad*

2. I am aware of the fact that the author retains these rights.
3. I certify that granting the non-exclusive license does not infringe the intellectual property rights or rights arising from the Personal Data Protection Act.

Narva, **20.05.2019**