**CMP1902M Object Oriented Programming 2022/23**

**Assignment 1: RESIT Report**

[*Expand the sections as necessary*]

Name: Adam Meleshko

Student ID: 27159808

**Code Review**

1. **Your code review** *(~400 words):   
   Provided code has some problems: logic, error handling, comments and use of words.   
   Starting with the latest, programmer should never make typos in the text that is shown to the user, because then the user understands that program was done by not a professional and as it can be seen, programmer made a spelling mistake in the word «commenter» in the first message to user. Additionally, the usage of the phrase «oh dear» in the last else branch is not suitable, since it is better for text to be in formal or semi-formal style.   
   Regarding comments, there are no comments, which is a big problem since it will be harder for another developer to understand meaning of the program. Even though this program is very simple, every programmer should develop this habit of commenting and explaining different parts of code which is both useful for the programmer himself/herself and other people who will work on a project.   
   Another major problem is absence of normal error handling, for instance, if user introduce non number string to the question about the mark, the program will stop executing by error. In such case, it is better if program inform user that input only can be number, not string, and give user another try. Moreover, when user introduces mark smaller that 0 or greater than 100, the program says that the score is wrong, however does not inform in which range it should be.   
   Finally, the biggest drawback of this program is logic. The if else branches which should be connected, actually are not. In case user introduces mark from 0 to 40, program will inform user about the fail score, but the program will go further to second if-else branch and move to else, thus the message about the wrong test score will appear, when it should not. Furthermore, this code does not cover the case when student has mark 0, the program will say that the input is wrong when actually it is not.   
   The main advice for the programmer is to test his/her program with different input data, to check if logic of execution is correct, also by this programmer will be able to find typos in messages to user if there are any. Regarding error handling, the recommendation is to use try-catch section for handling unexpected errors and personal advice is to use while loop, when the mark is received, to check if written string is number or not, to avoid errors of marks being non number string. Additionally, in my opinion it is better to check firstly if the mark given is correct, and only after show the message about the score of the student. Talking about comments, programmer should always comment the code, at least the way he/she thinks, because when coder will come back to the project, there is a chance of not understanding what code does, so it is highly recommended to comment code.*
2. **Reflection on code review: How do you feel about giving/receiving code reviews?** *(~400 words)*

*Personally I believe that every programmer, especially beginners, should both give and receive code reviews, since it is very useful for the progress.  
Giving code reviews is important for understanding the way other people think, because problems can be solved in different ways, so looking at the problem from another angle will make the programmer better, and also trying to understand the way someone thinks develop programmer’s logic so it makes him/her more coding aware. Additionally, in the future it will be easier to apply solutions of this problem to another tasks. Furthermore, it is better to learn from other’s mistakes, thus if programmer see some mistake in someone’s code, especially if it ruins the program, then he/she will intentionally try to omit this kind of mistakes. Therefore, reviewing someone’s code is positive both for gaining experience about how things should and should not be done.  
Receiving code reviews also is a good practice, because people can often make some obvious mistakes and even do not mention them. Therefore, when someone reviews your code and points at some mistakes, you know what to do to improve the code. Another problem is that usually people may think that they are doing something right even not mentioning that initially they are wrong. Everyone had such situation when they thought that they know how to solve some problem at the very beginning, started programming but eventually got the program which did not work properly. It is all because we start thinking in the wrong way at the very beginning, but we do not understand it, and this is where someone help with fresh point of view is needed. Moreover, person who review your code could offer a better solution (for instance faster), thus make you more experienced. So, receiving reviews of your code can point at mistakes that were not mentioned and offer more sophisticated solutions of a task.   
To sum up, both giving and receiving code reviews is a positive practice of programming and it is universally accepted that practice makes perfect. For this reason, it helps programmer master his/her skills to become better.*