**ASSIGNMENT 1 FRONT SHEET**

|  |  |  |  |
| --- | --- | --- | --- |
| **Qualification** | **BTEC Level 5 HND Diploma in Computing** | | |
| **Unit number and title** | **Unit 1: Programming** | | |
| **Submission date** | 21/10/2021 | **Date Received 1st submission** |  |
| **Re-submission Date** |  | **Date Received 2nd submission** |  |
| **Student Name** | Do Huu Duy | **Student ID** | GCC200018 |
| **Class** | GCC0903 | **Assessor name** | Nguyen Hung Dung |
| **Student declaration**  I certify that the assignment submission is entirely my own work and I fully understand the consequences of plagiarism. I understand that making a false declaration is a form of malpractice. | | | |
|  |  | **Student’s signature** | huuduy |

**Grading grid**

|  |  |  |
| --- | --- | --- |
| P1 | M1 | D1 |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **❒ Summative Feedback: ❒ Resubmission Feedback:** | | |
| **Grade:** | **Assessor Signature:** | **Date:** |
| **Lecturer Signature:** | | |

# Assignment Brief 1 (RQF)

**Higher National Certificate/Diploma in Computing**

|  |  |
| --- | --- |
| **Student Name/ID Number:** |  |
| **Unit Number and Title:** | **Unit 1: Programming** |
| **Academic Year:** | **2021 – 2022** |
| **Unit Assessor:** | **Vinh Hoang** |
| **Assignment Title:** | **Problem solving with algorithms** |
| **Issue Date:** | **01 April 2021** |
| **Submission Date:** |  |
| **Internal Verifier Name:** |  |
| **Date:** |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Submission Format:** | | | |
| *Format:*   * The submission is in the form of an individual written report. This should be written in a concise, formal business style using single spacing and font size 12. You are required to make use of headings, paragraphs and subsections as appropriate, and all work must be supported with research and referenced using the Harvard referencing system. Please also provide a bibliography using the Harvard referencing system.   *Submission*   * Students are compulsory to submit the assignment in due date and in a way requested by the Tutor. * The form of submission will be a soft copy posted on <http://cms.greenwich.edu.vn/>. * Remember to convert the word file into PDF file before the submission on CMS.   *Note:*   * The individual Assignment *must* be your own work, and not copied by or from another student. * If you use ideas, quotes or data (such as diagrams) from books, journals or other sources, you must reference your sources, using the Harvard style. * Make sure that you understand and follow the guidelines to avoid plagiarism. Failure to comply this requirement will result in a failed assignment. | | | |
| **Unit Learning Outcomes:** | | | |
| **LO1** Define basic algorithms to carry out an operation and outline the process of programming an application | | | |
| **Assignment Brief and Guidance:** | | | |
| **Assignment scenario**  **You have applied for a post as a trainee with a software development company and have been invited for an interview. You have been asked to demonstrate your problem solving and basic programming skills. To do this you have to prepare a report on using algorithms to solve problems.**  **You need to explain, using examples, how algorithms are used to solve simple business problems and the steps needed to be followed to produce a working program solution. You should make clear your assumption about your program. The problems to be solved will involve basic procedural programming instructions - sequence instructions (input, output and assignment statements), loops, conditional statements. Problems should be analysed and designed by the use of flowchart and demonstrated by the use of modules (procedures).**  **Tasks:**   * **State your simple business problems to be solved.** * **Analyse the problem and design the solutions by the use of suitable methods.** * **Demonstrate the compilation and running of a program** * **Evaluate how the problem is solved from the designed algorithm to the execution program written by a specific programming language.** | | | |
| **Learning Outcomes and Assessment Criteria (Assignment 1):** | | | |
| Learning Outcome | Pass | Merit | Distinction |
| LO1 | **P1** Provide a definition of what an algorithm is and outline the process in building an application. | **M1** Determine the steps taken from writing code to execution. | **D1** Examine the implementation of an algorithm in a suitable language. Evaluate the relationship between the written algorithm and the code variant. |

**Table of Contents**

[Assignment Brief 1 (RQF) 1](#_Toc85705329)

[ **Chapter 1 - State your simple business problems to be solved (P)** 4](#_Toc85705330)

[**1.1** **Definition of algorithms, characteristics of algorithm.** 4](#_Toc85705331)

[**1.2** **Explain the steps of building an application.** 5](#_Toc85705332)

[**1.3** **Represent a small and simple problem.** 7](#_Toc85705333)

[ **Chapter 2 - Analyze the problem and design the solutions by the use of suitable methods. (P)** 7](#_Toc85705334)

[**1.1** **Analyze the business problem and turn it into application.** 7](#_Toc85705335)

[**1.2** **Design the algorithm to solve the problem by using suitable diagrams such as Flowchart.** 8](#_Toc85705336)

[ **Chapter 3 - Demonstrate the compilation and running of a program (M)** 9](#_Toc85705337)

[**1.1** **Demonstrate how the application is implemented by using suitable programming language (C#).** 9](#_Toc85705338)

[**1.2** **explain how the source code is compiled and run in C#** 11](#_Toc85705339)

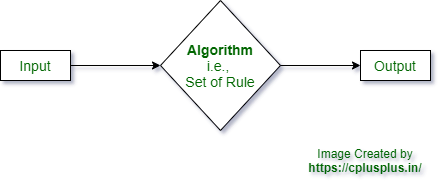
[ **Chapter 4 - Evaluate how the problem is solved from the designed algorithm to the execution program written by a specific programming language. (D)** 13](#_Toc85705340)

[**1.1** **Test plane.** 13](#_Toc85705341)

[**1.2** **Evaluate how the problem is solved from the designed algorithm to the execution program written by a specific programming language.** 16](#_Toc85705342)

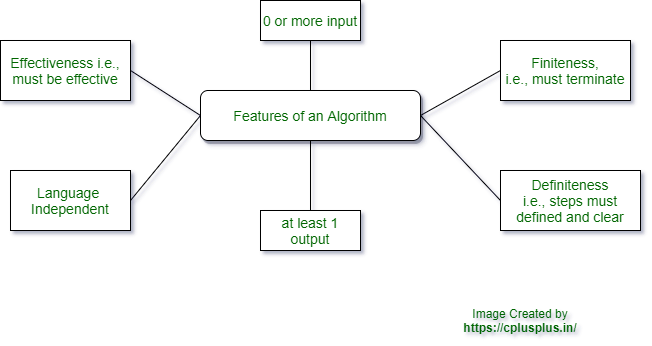
[References 17](#_Toc85705343)

* **Chapter 1 - State your simple business problems to be solved (P)**
  1. **Definition of algorithms, characteristics of algorithm.**
* **Algorithm** is a step-by-step procedure which is used to solve a problem. It is important Computer Science and Software Engineering. We can improve our program efficiency in cost and time by choosing appropriate algorithm and data structures for a particular program. (cplusplus, 2021)

****

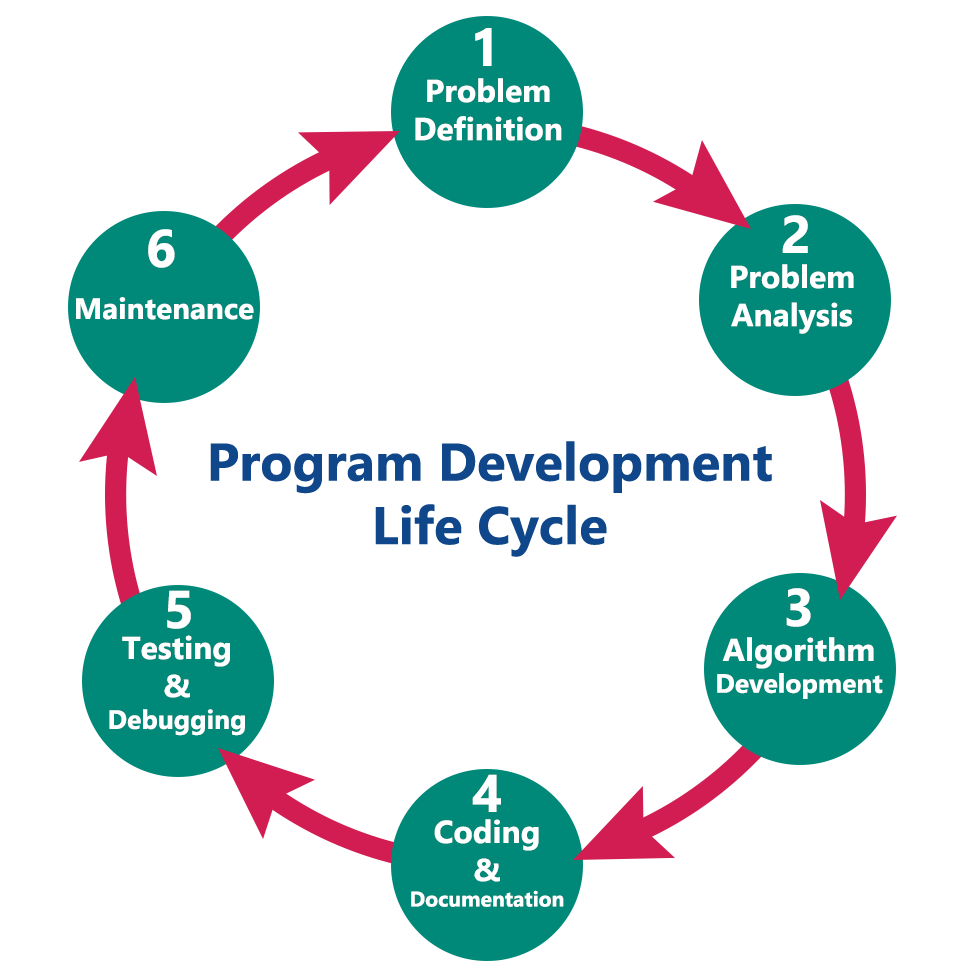
Picture 1: Algorithm diagram.

* **Definition of algorithm:**“An algorithm is a well-defined computational procedure, which takes some value (or set of values) as input and produces some value, or a set of values, as output”.
* **characteristics of algorithm.**

****

Picture 2: Characteristics diagram of the algorithm.

* **Finiteness: means it must always terminate after a finite number of steps.**
* **Definiteness**: means each step must be precisely defined and clear.
* **Input**: means it has zero or more inputs, i.e., an algorithm can run without taking any input.
* **Output**: means it has one or more outputs, i.e., an algorithm must produce at least one output.
* **Effectiveness**: means it is also generally expected to be effective.
* **Independent:** An algorithm should be and **unambiguous** and independent of any programming code, i.e., **language independent**.
  1. **Explain the steps of building an application.**
* **Problem Definition**
* **Problem Analysis**
* **Algorithm Development**
* **Coding & Documentation**
* **Testing & Debugging**
* **Maintenance** (mql5, 2021)

****

Picture 3: Diagram of the steps to build an application.

* **Problem Definition:**

In this phase, we define the problem statement and we decide the boundaries of the problem. In this phase we need to understand the problem statement, what is our requirement, what should be the output of the problem solution. These are defined in this first phase of the program development life cycle.

* **Problem Analysis:**

In phase 2, we determine the requirements like variables, functions, etc. to solve the problem. That means we gather the required resources to solve the problem defined in the problem definition phase. We also determine the bounds of the solution.

* **Algorithm Development:**

During this phase, we develop a step by step procedure to solve the problem using the specification given in the previous phase. This phase is very important for program development. That means we write the solution in step by step statements.

* **Coding & Documentation:**

This phase uses a programming language to write or implement actual programming instructions for the steps defined in the previous phase. In this phase, we construct actual program. That means we write the program to solve the given problem using programming languages like C, C++, Java, Forex programming languages like our (mt4 and/or mt5) language(s), etc.,

* **Testing & Debugging:**

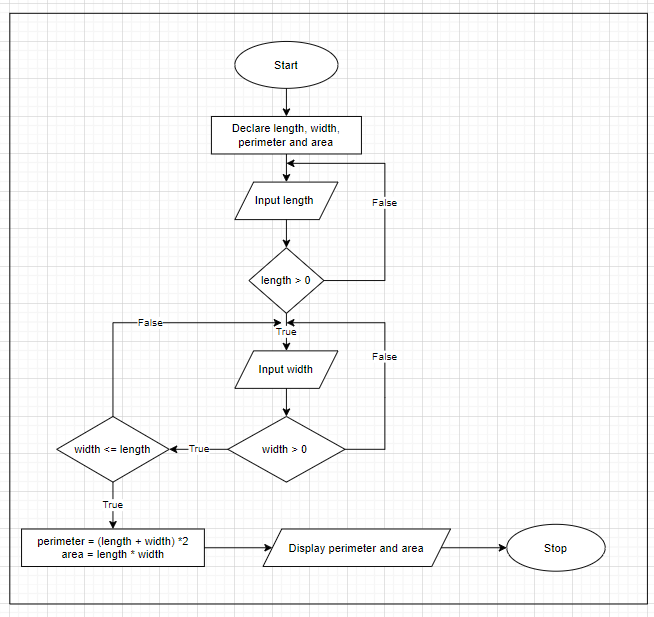
During this phase, we check whether the code written in previous step is solving the specified problem or not. That means we test the program whether it is solving the problem for various input data values or not. We also test that whether it is providing the desired output or not.

* **Maintenance:**

During this phase, the program is actively used by the users. If any enhancements found in this phase, all the phases are to be repeated again to make the enhancements. That means in this phase, the solution (program) is used by the end user. If the user encounters any problem or wants any enhancement, then we need to repeat all the phases from the starting, so that the encountered problem is solved or enhancement is added.

* 1. **Represent a small and simple problem.**
* In this part, I will choose a small and simple problem to represent and perform it.
* I choose a small problem, that is calculate perimeter and area of a rectangle.
* In this problem the algorithm can solve it easier and faster.
* **Chapter 2 - Analyze the problem and design the solutions by the use of suitable methods. (P)**
  1. **Analyze the business problem and turn it into application.**
* For this problem, I will declare four variables with float data to perform. Four variables have names is width, length, perimeter and area.
* Specifying the input requirements:
* I will input length and width of the rectangle.
* Specifying the output requirements.
* I will print perimeter and area of the rectangle after being calculated.
* Specifying the processing requirements.
* The first: I will input the length of the rectangle and check it. If user enter the length less than 0, the program will require the user re-enter the length of the rectangle. The length of the rectangle must be larger than 0.
* The second: After the user enter the length exactly. Then the user will continue entering the width of the rectangle and check it. If the user enters the width less than 0 and the width equal to the length, the program will require the user re-enter the width. The width must be larger than 0 and less than the length.
* The third: After entering the length and width of the rectangle exactly. The program will execute to calculate perimeter and area of the rectangle with the following formula.
* Perimeter = (length + width) \* 2.
* Area = length \* width.

* 1. **Design the algorithm to solve the problem by using suitable diagrams such as Flowchart.**

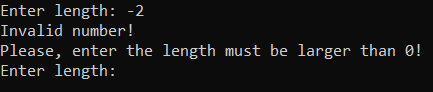
****

Picture 4: Flowchart diagram.

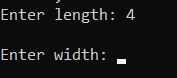
* **Chapter 3 - Demonstrate the compilation and running of a program (M)**
  1. **Demonstrate how the application is implemented by using suitable programming language (C#).**
* When the program will be executed. It will print a command asking the user to enter the length of the rectangle.



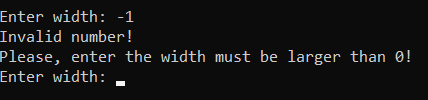
* Then, the user will enter the length. But if the user enters the length is less than 0. The program will print a command “Invalid number!” and require the user to re-enter the length of the rectangle that must be larger than 0.



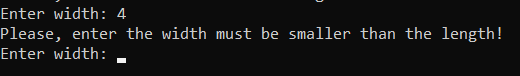
* If the user enters the length of the rectangle exactly. The program will continue printing a command asking the user to enter the width of the rectangle.



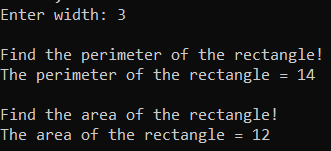
* Then, the user will enter the width. But if the user enters the width is less than 0. The program will print a command “Invalid number!” and require the user to re-enter the width of the rectangle that must be larger than 0.



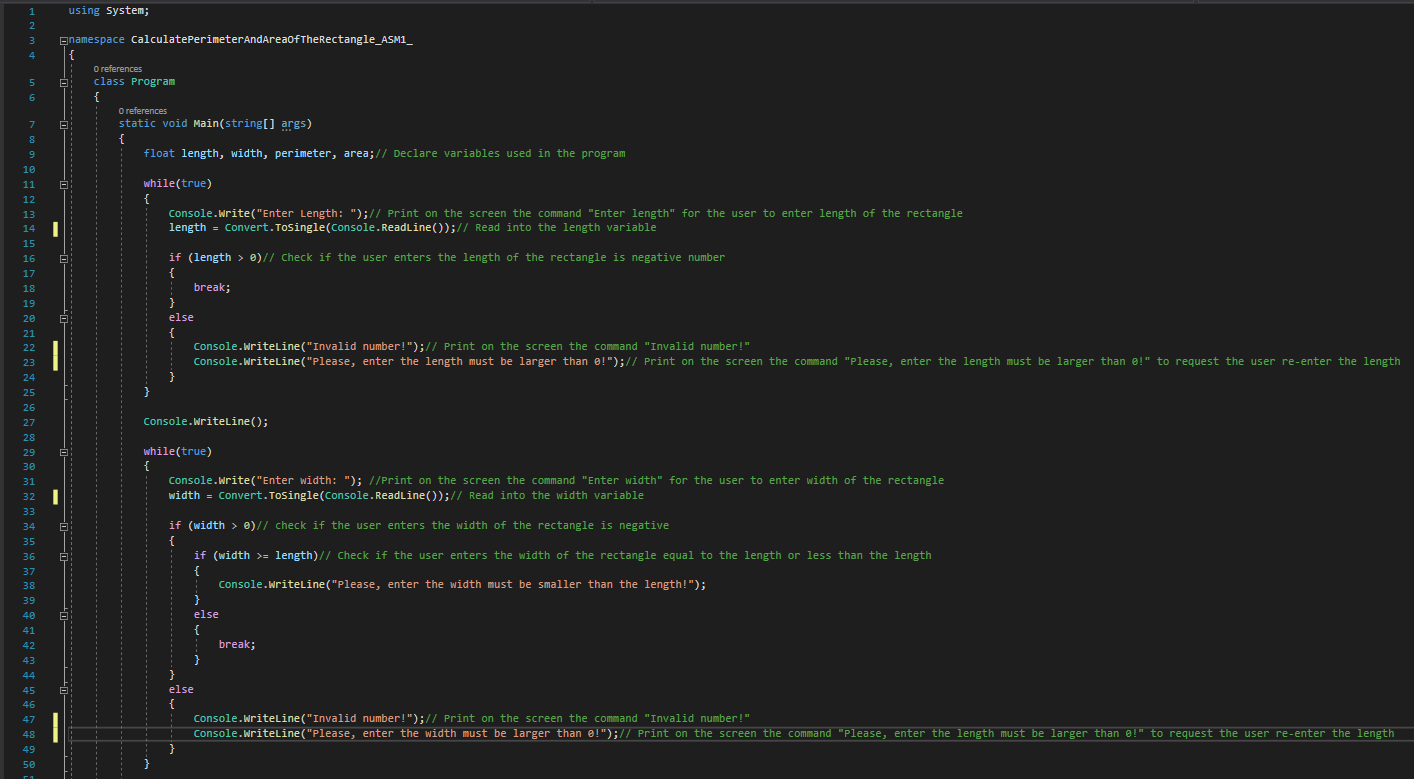
* Now the user enters the width is larger than 0. But if the user enters the width equal to the length. The program will require the user entering the width that must be less than the length of the rectangle.

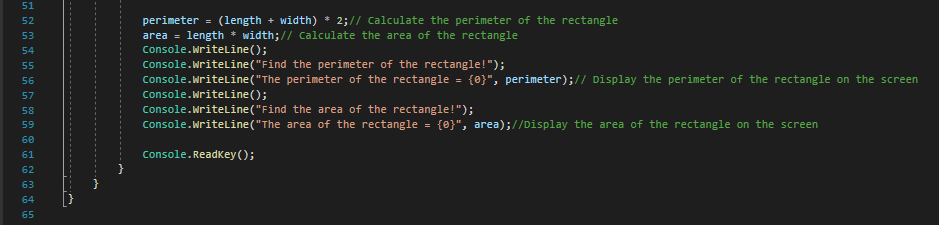


When the user enters the width of the rectangle exactly. The program will execute and calculate the perimeter and area of the rectangle, then print on the screen.

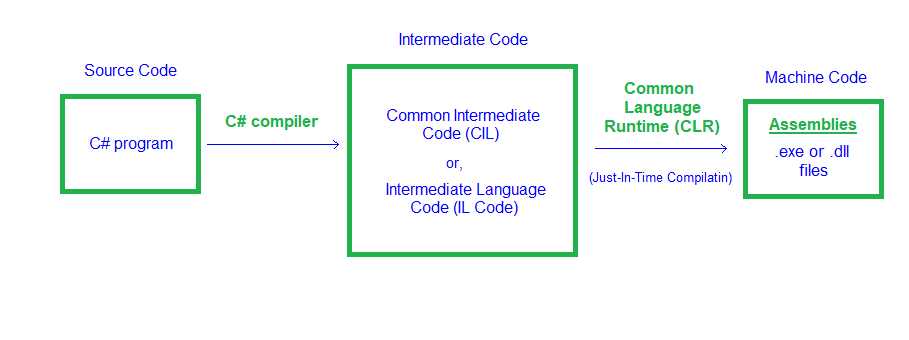


* The source code:





* 1. **explain how the source code is compiled and run in C#**
* Step 1: Write a C# code.
* Step 2: Compile the code using a C# compiler.
* Step 3: Now compiler checks if the code contains an error or not. If no error is found then the compiler move to the next step. Or if the compiler found errors, then it will immediately tell the developer that an error is found in the given line, so that the developer can correct them. After correcting the error when you again run the code the compiler again check for the errors, if no error found then it will move to the next step or if an error is found then the compiler gives a message to the developer. In C# there are two types of errors:
* **Compiler error:**Errors that occur when the developer violates the rules of writing syntax are known as Compile-Time errors. This compiler error indicates something that must be fixed before the code can be compiled. All these errors are generally detected by the compiler and thus are known as compile-time errors. For example, missing semicolon, missing parenthesis, etc.
* **Runtime error:** Errors that occur during program execution(run-time) after successful compilation is called run-time errors. One of the most common run-time errors is division by zero also known as Division error. These types of errors are hard to find as the compiler doesn’t point to the line at which the error occurs.
* Step 4: Languages such as Java or C# are not directly converted or compiled into machine-level language or machine instructions. These languages need to be converted to an intermediate code first, which is a partially half compiled code. For C#, the source code is converted to an intermediate code which is known as [**Common Intermediate Language (CIL)**](https://www.geeksforgeeks.org/common-language-runtime-clr-in-c-sharp/)or**Intermediate Language Code (ILC or IL code)**. This CIL or IL Code can run on any operating system because C# is a *Platform Independent*Language.
* Step 5: After converting the C# source code to Common Intermediate Language (CIL) or Intermediate Language Code (ILC or IL code, the intermediate code needs to be converted to machine understandable code. C# uses the *.NET Framework* and as part of this .NET Framework, the *Virtual Machine component* manages the execution of programs written in any language that uses the. NET Framework. This virtual machine component is known as [**Common Language Runtime (CLR)**](https://www.geeksforgeeks.org/common-language-runtime-clr-in-c-sharp/)which translates the CIL or IL code to native code or machine understandable code or machine instructions. This process is called the[**Just-In-Time (JIT) Compilation**](https://www.geeksforgeeks.org/what-is-just-in-time-jit-compiler-in-dot-net/)or**Dynamic Compilation**which is the way of compiling code during the execution of a program at run time only.
* Step 6:  Once the C# programs are compiled, they’re physically packaged into **Assemblies**. An assembly is a file that contains one or more namespaces and classes. As the number of classes and namespaces in program grows, it is physically separated by related namespaces into separate assemblies. Assemblies typically have the file extension **.exe**or **.dll,**depending on whether they implement applications or libraries respectively, where EXE stands for *Executable*and DLL stands for *Dynamic Link Library*. An EXE (Executable) file represents a program that can be executed and a DLL (Dynamic Link Library) file includes code (Eg: Library) that can be reused across different programs.
* Step 7: Now, the C# compiler returns the output of the given c# code.



Picture 5: Compilation process of C# code

So, this is how the whole process of compilation and execution of a C# code takes place. (geeksforgeeks, 2021)

* **Chapter 4 - Evaluate how the problem is solved from the designed algorithm to the execution program written by a specific programming language. (D)**
  1. **Test plane.**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Test case** | **Function** | **Test Data** | **Expected output** | **Actual output** | **Result** |
| 1. | Verify that the length of the rectangle entered exactly when user enters valid length. | calculate perimeter and area of rectangle | Enter length: 4 | After the user enters the length exactly. The program will continue to execute and ask the user enters the width of the rectangle. | After the user enters the length exactly. The program will continue to execute and ask the user enters the width of the rectangle. | Pass |
| 2. | Verify that the program has error when the user enters the length of the rectangle is less than 0. | calculate perimeter and area of rectangle | Enter length: -4 | When the user enters the length of the rectangle incorrectly. The program will notify “Invalid numbers!” and ask the user re-enters the length. | When the user enters the length of the rectangle incorrectly. The program will notify “Invalid numbers!” and ask the user re-enters the length. | Pass |
| 3. | Verify that the width of the rectangle entered exactly when user enters valid width | calculate perimeter and area of rectangle | Enter width: 3 | When the user enters the width of the rectangle exactly. The program will continue to execute and calculate the perimeter and the area of the rectangle and then will print on the screen the perimeter and area of the rectangle. | When the user enters the width of the rectangle exactly. The program will continue to execute and calculate the perimeter and the area of the rectangle and then will print on the screen the perimeter and area of the rectangle. | Pass |
| 4. | Verify that the program has error when the user enters the width of the rectangle is less than 0 | calculate perimeter and area of rectangle | Enter width: -3 | When the user enters the width of the rectangle incorrectly. The program will notify “invalid number!” and ask the user re-enters the width. | When the user enters the width of the rectangle incorrectly. The program will notify “invalid number!” and ask the user re-enters the width. | pass |
| 5. | Verify that the program has error when the user enters the width of the rectangle equal to the length of the rectangle. | calculate perimeter and area of rectangle | Enter width: 4 | When the user enters the width equal to the length of the rectangle. The program will notify error and print “Please, enter the width must be smaller than the length” | When the user enters the width equal to the length of the rectangle. The program will notify error and print “Please, enter the width must be smaller than the length” | Pass |

* 1. **Evaluate how the problem is solved from the designed algorithm to the execution program written by a specific programming language.**
* Pros: The program brief and easy to use. The program works fast and convenient. It is easy for user to calculate. It doesn’t take long.
* Cons: The program will have error when the user enters the length and the width of the rectangle is characters. In addition, the program just uses to the calculate perimeter and area of the rectangle.
* Improvement: In the future, I will fix in case the program has errors when the user enters the characters. I will do more functions like multiplication table, calculate perimeter and area of circles, parallelograms, triangles, and so on.

# References

cplusplus, 2021. *characteristics-or-features-of-an-algorithm.* [Online]   
Available at: https://www.cplusplus.in/characteristics-or-features-of-an-algorithm/

geeksforgeeks, 2021. *how-c-sharp-code-gets-compiled-and-executed.* [Online]   
Available at: https://www.geeksforgeeks.org/how-c-sharp-code-gets-compiled-and-executed/  
[Accessed 31 August 2021].

mql5, 2021. *mql5.com.* [Online]   
Available at: https://www.mql5.com/en/blogs/post/713281