

.NET Technology

(503112)

By Mai Van Manh

Lab 01: CSharp Language Overview

Mỗi bài học thực hành đều đi kèm với một số tập tin PDF hướng dẫn các kiến thức có liên quan đến các bài tập. Sinh viên cần xem các tập tin hướng dẫn (và video nếu có) trước khi thực hiện các bài tập bên dưới. Sinh viên chỉ nộp lại source code (.cs, .txt; và static contents, media) của từng bài tập, không nộp toàn bộ project. Với bài tập giao diện (Forms, Web) sinh viên cần cung cấp thêm ảnh screenshot từng giao diện của bài tập. Dữ liệu mỗi bài tập cần được đặt trong một thư mục riêng (trừ khi bài tập chỉ có 1 file duy nhất), toàn bộ bài tập cần được đặt trong thư mục có dạng 520H1234_NguyenVanA và được nén lại dưới dạng zip/rar trước khi nộp. Bài nộp không có thông tin sinh viên hoặc nộp toàn bộ source code của project sẽ không được tính.

Exercise 1: Simple calculator

You need to create a program that can perform the four basic arithmetic operations: addition, subtraction, multiplication, and division. The program should take two numbers as input and output the result of the operation. For example, if the user enters 10 and 5, then choose addition, the program should output 15.

Implement thorough input validation to ensure the following:

- Both input values are numeric.
- For division, the second input number is non-zero.
- Input values are within a reasonable range to avoid overflow or unrealistic results.
- The program handles invalid inputs gracefully, providing appropriate feedback to the user.

Exercise 2: Class Library Application

Create a class library program that has a method to calculate the factorial of a number. The method should take an integer as an input and return the factorial of that number. Compile the class library program as a DLL file.

Create another console program that uses the DLL file to calculate the factorial of a number. The console program should prompt the user to enter a series of integers. For each integer, the console program should call the method in the DLL file to calculate the factorial of the integer and print the result to the console. The console program should repeat prompting the user for integers until the user enters the value -1.

Exercise 3: Guessing game

You need to create a program that randomly generates a number between 1 and 100. The user will then try to guess the number. The program should provide feedback to the user after each guess, and it should keep track of the user's score.

For example, if the user guesses 50 and the correct number is 60, the program should output "Too high!" The program should continue until the user guesses the correct number. When the game is over, show how many guesses the user has made.

Exercise 4: List of words

You need to create a program that reads a list of words from a file. The program should then print the words to the console. The program should also be able to search for a specific word in the list. For example, if the file contains the words "cat", "dog", "fish", and "mouse", the program should be able to print the word "cat" if the user enters "cat" as input.