

Department of Electrical and Computer Engineering

EECE 2140: COMPUTING FUNDAMENTALS FOR ENGINEERS

Spring 2026

Dr. Fatema Nafa

InClassWork#02

Learning Objectives:

By the end of this lab, students should be able to:

- Declare and use variables of type int, char, bool, string.
- Write and debug code using if/else, for, while, do...while, and switch.
- Trace a loop and predict its behavior.
- Fix simple logical and off-by-one errors.

Submission Requirements

For each submission, **two files** are required: one **PDF file** and one **C++ (.cpp) file**. The details for each are provided below.

1. Technical Report (PDF)

Your PDF report must include:

- Pseudocode for the problem.
- A clear explanation of your solution logic.
- Execution snapshots showing program output.
- **File name format:**
AssignmentXX_LastName.pdf
(Example: Assignment02_Smith.pdf)

2. C++ Code File(s) (.cpp)

Submit your actual C++ source code files:

- **File name format:**
AssignmentXX_ProblemYY_LastName.cpp
(Example: Assignment02_Problem01_Smith.cpp)
- Submit a separate .cpp file

Problem:

In this lab, you'll work with a simple “secret code”. Each number represents a letter:
 $1 : A, 2 : B, \dots, 26 : Z$. any other value should be treated as an **invalid code**.

Write a program that:

1. Ask the user how many numbers are in the secret sequence.
2. Repeatedly asks for each number.
3. For each number:
 - o If it's between **1 and 26**, convert it to an uppercase letter ($1 : A$, etc.).
 - o Otherwise, treat it as an **invalid code** and decode it as **?**.
4. At the end, print the decoded message.

Tasks

Task 1: Compile, run and observe

1. **Compile:**
2. **Use this test case:**
 - o Length: 5
 - o Codes: 8 5 12 12 15

Expected message: HELLO
3. **Observe the issues:**
 - o Wrong letters?
 - o Extra character?
 - o Problems when length is 0 or negative?

Write down at least two wrong behaviors.

Task 2: Fix using if, for, and basic logic

Fix the following issues:

1. **Length check**
 - o If length ≤ 0 , print an error message and **do not** decode anything.
2. **Loop bounds**
 - o Update the for loop so that it runs exactly length times.

3. Off-by-one error in letter mapping

- Ensure 1 : 'A', 2 : 'B', ..., 26 : 'Z'.

re-run the test case they get HELLO correctly.

Task 3: Add input validation with while

Use a **while loop** to validate the length:

- If the user enters length ≤ 0 , keep asking:

Task 4: Add a repeat menu with do...while and switch

Now turn it into a small “menu loop”:

1. Wrap the decoding logic inside a **do...while** loop so the user can decode multiple messages in one run.
2. Use a **switch** to handle the user’s menu choice:
3. Behavior:
 - 1: Run the length and sequence input and decode as you implemented.
 - 2: Print a short message explaining 1–26 and A - Z, other numbers ?.
 - 0: Exit.
 - Any other number: print Invalid option and show menu again.