

# COMP 201 - Fall 2023

## Assignment 1 - Strings in C

Assigned: 28 February 2024 23:59, Due: 14 March 2024 23:59

Doğa Kukul (dkukul19@ku.edu.tr) is the lead person for this assignment.



## 1 Introduction

In this assignment, you'll dive deep into the fundamentals of dynamic memory management, file input/output (I/O) operations, and string manipulation using C. These skills are crucial for processing and managing real-world data effectively. Through the lens of a thrilling espionage scenario, you'll be tasked with encrypting and decrypting secret messages exchanged between undercover agents and their commanding officer.

## 2 The Scenario

In a world shadowed by intrigue and covert operations, secure communication is the backbone of intelligence work. You're a codebreaker in the heart of this clandestine world, decoding the secrets of the friend and foe alike. Provided with several text files containing coded messages, your mission is to use your programming prowess to unveil their contents—some need encryption for secure transmission, while others require your keen insight to decode.

### 3 Files Overview

message1.txt: A conversation between Agent Zero and the Chief in plain text—a risky format that demands encryption for security.

message2\_encrypted.txt: An encrypted dispatch from an allied intelligence base awaits your decryption skills.

message3\_encrypted.txt: Another agency has sent a coded message; the encryption index is unknown, challenging you to decrypt it using linguistic analysis.

message4\_encrypted.txt: A captured communique from adversaries; decrypt it to uncover their plans, using a known frequency heuristic.

### 4 Logistics

This is an individual project. All handins are electronic. Clarifications and corrections will be announced on Blackboard.

### 5 Handout Instructions

#### 5.1 How to start

I Accept the GitHub Classroom assignment using the link: <https://classroom.github.com/a/MQoD4cwW>

II Clone the GitHub repository created for you to a Linux machine in which you plan to do your work (We advice you to do your work on our linux servers [linuxpool.ku.edu.tr]. See Section 9 for details.)

```
$ git clone https://github.com/COMP-201-Spring-2024/assignment-1-strings-in-c-USER.git
```

(Replace USER with your GitHub username that you use to accept the assignment)

III **[IMPORTANT]** After cloning the repository, you are required to write the following honor code in a new file called "honor.txt" and commit & push it: "I hereby declare that I have completed this assignment individually, without support from anyone else." You can use the following command to create "honor.txt" file and write the honor code in it:

```
$ echo "I hereby declare that I have completed this assignment individually,  
without support from anyone else." > honor.txt
```

#### 5.2 Main Task

In the environment that you set up, create a main.c file. You will do all of your coding in this file. To run your code, simply run the "make" command in the directory you're working on.

#### 5.3 How to Read a File

- First, open the text file using the `fopen()` function.
- Then, use `fgets()` function to read text from the stream, and store it as a string. The newline or EOF character stops `fgets()` so you can check the newline or EOF file character to read the whole line. **Note:** `fgets()` also reads endline characters, and you should ignore newline character at the end of the sentence.
- When you are done with the file, close the file with the `fclose()` function.

- For more information, you can refer to these links: [Link 1](#), [Link 2](#)

## 6 Tasks

### Task 1: Secure Agent Zero's Communication

Agent Zero has reported vital intelligence in `message1.txt`. However, its plain text format is a glaring vulnerability. Your first mission is to encrypt this message. Prompt for a shift value (N), then apply a Caesar cipher, shifting each letter N places forward in the alphabet. Save the secured message as `message1_encrypted.txt`. Remember, the art of secrecy is in the subtlety of your shift.

Note: If N is 3, the letter A becomes D, the letter K becomes N, and the letter Z becomes C. You can handle letter casing as you wish, you shouldn't change non-alphabet characters.

### Task 2: Revealing Allied Secrets

A message (`message2_encrypted.txt`) from an allied base carries essential intelligence, encrypted with a shift of 5. Decrypt this message to ensure the information flows securely within our network. The decrypted contents should be stored in `message2_decrypted.txt`.

### Task 3: Deciphering the Unknown

The challenge intensifies with `message3_encrypted.txt`, a message whose key is lost in the ether. By identifying the most frequent letter and assuming it represents 'e', decrypt the message, revealing the hidden text. Store your findings in `message3_decrypted.txt`. This task tests your ability to find light in the darkness.

### Task 4: Unmasking the Enemy

Intercepted enemy communication (`message4_encrypted.txt`) provides a unique challenge: its most common word is "the". Use this clue to break their code, piecing together their scheme. Your successful decryption will be your triumph, recorded in `message4_decrypted.txt`.

### Task 5: Reflect on Your Journey

In the quiet aftermath, reflect on the tools of your trade. Comment on the Caesar cipher's effectiveness at the end of your code, discussing its strengths and weaknesses. Optionally, if inspiration strikes, propose an alternative encryption method that could better guard the secrets of the shadows.

## 7 Evaluation

Your score will be computed out of a maximum of 105 points based on the following distribution:

**Effective Use of Version Control Points.** You are required to push your changes to the repository frequently. If you only push the final version, even if it is implemented 100% correctly, you will lose a fraction of the grade because you are expected to learn to use Version Control Systems effectively. You do not have to push every small piece of change to Github but every meaningful change should be pushed. For example, each of the functions coded and tested can be one commit. **For each task, there should be at least one commit (with proper commit message) that includes just modifications on that task.**

Task Description	Points
Task 1	15
Task 2	5
Task 3	30
Task 4	30
Task 5	10
Style points	5
Effective use of version control	5
Optional encryption method	5

Table 1: Scoring Distribution

**Style Points.** Finally, we've reserved 5 points for a subjective evaluation of the style of your solutions and your commenting. Your solutions should be as clean and straightforward as possible. Your comments should be informative, but they need not be extensive.

**Important Note:** We use automated plagiarism detection to compare your assignment submission with others and also the code repositories on GitHub and similar sites. Moreover, we plan to ask randomly selected 10% of students to explain their code verbally after the assignments are graded. And one may lose full credit if he or she fails from this oral part.

## 8 Handin Instructions

As with Assignment 0, we use GitHub for the submissions as follows. Note that we want you to get used to using a version management system (Git) in terms of writing good commit messages and frequently committing your work so that you can get most out of Git.

I Commit all the changes you make: `$ git commit -a -m "commit message"`

**Note:** Use meaningful commit messages as in huge projects they become really helpful.

Try to gain this habit from early on.

II Push your work to GitHub servers: `$ git push origin main`

## 9 How to use linuxpool.ku.edu.tr linux servers <sup>1</sup>

I Connect to KU VPN (If you are connected to the KU network, you can skip this step.)

See for details: <https://confluence.ku.edu.tr/kuhelp/ithelp/it-services/network-and-wireless/vpn-access>

II Connect to linuxpool.ku.edu.tr server using SSH (Replace USER with your Koç University username):

`$ ssh USER@linuxpool.ku.edu.tr`

(It will ask your password, type your Koç University password.)

III When you are finished with your work, you can disconnect by typing: `$ exit`

Your connection to the server may drop sometimes. In that case, you need to reconnect.

We advice you to watch the following video about the usage of SSH, which is used to connect remote servers, and SCP, which is used to transfer files between remote servers and your local machine: <https://www.youtube.com/watch?v=rm6pewTcSro>

<sup>1</sup>For details, please see the guide on linuxpool that we have announced on Blackboard

## 10 Academic Integrity

All work on assignments must be done individually unless stated otherwise. You are encouraged to discuss with your classmates about the given assignments, but these discussions should be carried out in an abstract way. That is, discussions related to a particular solution to a specific problem (either in actual code or in the pseudocode) will not be tolerated. In short, turning in someone else's work, in whole or in part, as your own will be considered as a violation of academic integrity. Please note that the former condition also holds for the material found on the web as everything on the web has been written by someone else. See [Koç University - Student Code of Conduct](#).

## 11 Late Submission Policy

**You may use up to 7 grace days (in total) over the course of the semester for the assignments.** That is you can submit your solutions without any penalty if you have free grace days left. Any additional unapproved late submission will be punished (1 day late: 20% off, 2 days late: 40% off) and **no submission after 2 days will be accepted (grace days included).**