Problem statement

Statement: General statement

Write an `.exe` program in assembly language that reads a **string** of maximum 255 characters, or until the *enter* key is pressed and a **letter** followed by the *enter* key, which replaces all occurrences of that letter in the string with the character `#`. the resulting string is then printed to the console. For additional credit, modify the program so that the letter read from the keyboard is deleted from the string.

Examples

Replacing a character

INPUT: `Assembly language is fun to learn, write and teach.[enter]a[enter]`
OUTPUT: `#ssembly l#ngu#ge is fun to le#rn, write #nd te#ch.`

Removing a character

INPUT: `Assembly language is fun to learn, write and teach.[enter]a[enter]`
OUTPUT: `ssembly lnguge is fun to lern, write nd tech.`

Requirements

Requirements: Nonfunctional requirements

- You shall use the data segment provided in the template. You **may not** define additional variables in your program, other than the ones that are already declared in the data segment.
- You shall incorporate in your solution the macros and procedures provided alongside the template.
- The input string may contain any character from the standard keyboard. You shall only process the characters that correspond to letters (either lowercase or uppercase).
- It is encouraged to define your own macros and procedures in order to solve the problem (if possible). In this case, each macro or procedure

should be roughly documented with its intended behaviour.

General rules

You may use any resources available to solve the problem (except AI generational tools).

You may be asked questions about your implementation. Be prepared to answer them.

You have a total of **75 minutes** to solve the problem and provide a working solution. A working solution is a program that compiles and executes, even if it does not perform the required operation.

The total time should be divided in two parts: a **30 minutes** session, followed by a **10-15 minutes** break and then another **35 minutes** session.

The total time may be extended with an additional **30 minutes**, if less then 2 teams manage to complete the implementation within the initial **75 minutes**.

You should be able to describe the behaviour of the program in an algorithmic manner.

The solution shall be submitted on GitHub, Teams or via email at the address Tudor.Coroian@cs.utcluj.ro. One submission/team is enough. Submission time **is not** included in the **75 minutes** for solving the problem.

You may not ask for help from students in other teams during the two sessions of implementation. You are encouraged to collaborate within your own team.