

Exercise 02 - Palindrome Pointer

Skills trained: Pointer Arithmetics

About this exercise

This exercise will be familiar for some of you. The main difference to the original exercise of last semester is that this time you must use pointer arithmetics.

Exercise pointer arithmetics, palindrome

Read a string and find out whether it is a palindrome.

Detailed exercise

A palindrome is a word or a sentence which reads the same backward and forward. Read a line with at most 80 characters (if it is longer, use the first 80 characters). Then use one or multiple loops in which you use pointers to find out whether the string is a palindrome.

Hints

- Please do not use “masked” array notation. Eg `*(array + i)` is not pointer arithmetics!
- To read the line, use `fgets` with the appropriate arguments.
- To find out whether a character is a lower case letter, remember that characters are numbers and therefore, comparisons like `ch >= 'A'` are possible.
- To convert lower case letters to upper case letters subtract 32 (see ASCII table).
- You can use a loop with a left and a right index and make sure to step over symbols like commas and dots.
- You can save a lot of time by implementing some procedures, for instance to check whether a character is a letter or a symbol.

Examples

Example 1

Input:

This is not a palindrome!

Output:

not a palindrome

Example 2

Input:

A man, a plan, a canal, Panama!

Output:

palindrome

Example 3

Comment: Empty lines or lines only containing one letter are considered to be palindromes.

Input:

!

Output:

palindrome