Introduction to Data Structures

Jinkyu Lee

Dept. of Computer Science and Engineering,
Sungkyunkwan University (SKKU)

Homework 2B

- 40 points for coding evaluation
  - Submission format
    - File name: yourid_HW2B.c
      - Example: 2000123456_HW2B.c
    - File type: .c (NOT .cpp)
  - Submission site: https://icampus.skku.edu
    - Week 5: [Homework] 2B (code)

- 1 point for report
  - The report is not evaluated in detail but evaluated as Pass/Fail
  - Template: Homework Report Template.docx
  - Submission format
    - File name: yourid_HW2B.pdf
      - Example: 2000123456_HW2B.pdf
    - File type: .pdf
  - Submission site: https://icampus.skku.edu
    - Week 5: [Homework] 2B (report)

- Due date
  - 10/13 23:59 (no late submission accepted)

Rules for homework

- You should follow instructions.
  - Complier
    - You will get no/less point if your program cannot be complied with the specified complier
  - Input/output format
    - You will get no/less point if TA's automatic evaluation program cannot parse your input or output.
  - Permitted modification scope
    - You will get no/less point if you modify code outside of the permitted modification scope
  - All other rules
    - You will get severe penalty or no/less point if you violate the given rules.

Complier and input/output rules for homework

- Every implementation homework will be evaluated by TA’s automatic evaluation program with the following complier.
  - Complier: GCC 7.X, 8.X, 9.X or 10.X
    - https://gcc.gnu.org/
  - You will get no/less point if your program cannot be complied with GCC 7.X, 8.X, 9.X or 10.X.
    - For example, do not rely on visual studio.
  - You can use standard library such as *stdlib.h* and *math.h*.

- Input/output format
  - You will get no/less point if TA’s automatic evaluation program cannot parse your input or output according to the following rules.
  - Use `stdin` and `stdout`

Problem

- Problem: Pseudo-Palindrome Checker
  - A palindrome is a sequence of characters that reads the same backward as forward.
  - Write a program that prints (i) a sequence of characters that reads backward, and (ii) the number of characters which do not match between backward and forward.
    - We use the upper-case alphabets from ‘A’ to ‘Z’ only.
    - However, we use one more special character ‘*’. If the character is ‘*’, it is regarded as matched with any character. When there is at least on ‘*’, the output number should be the minimum number.
  - The number of input characters is not more than one hundred.
  - You should use
    - Queue (including Deque),
    - Stack, or
    - Queue (including Deque) and stack.

Input/Output

- Input (no space)

```
ABCDA
```

- Output (no space)

```
ADCBA
2
```

Input/Output

- Input (no space)

```
ABC*A
```

- Output (no space)

```
A*CBA
0
```

Template

- There is no template.

Evaluation

- Evaluation
  - TA will test several cases.
  - For each test case,
    - If your C code results in an answer within 10 seconds on a platform with average computing power,
      - If your output is perfect for both (i) a sequence of characters that reads backward and (ii) the number of characters which do not match between backward and forward,
        - You get 100%.
      - Else,
        - You get 0%.
    - Else,
      - You get 0%.