Data Structures 2020

Homework #3

- Being able to send encoded messages during World War II was very important to the Allies. The messages were always sent after being encoded with a known password. Having a fixed password was of course insecure, thus there was a need to change it frequently. However, a mechanism was necessary to send the new password.
- One of the mathematicians working in the cryptographic team had a clever idea that was to send the password hidden within the message itself. The interesting point was that the receiver of the message only had to know the size of the password and then search for the password within the received text.

Homework #3

- A password with size N can be found by searching the text for the most frequent substring with N characters. After finding the password, the password can be used to decode the message.

- Your mission has been simplified as you are only requested to write a program that, given the size of the password and the encoded message, determines the password following the strategy given above.

Input & Output

- Standard/Console Input
  - Input consists of two lines with the size of the password, $3 \le N \le 30$, followed by the text representing the encoded message whose size is less than 5 mega bytes. To simplify things, the input text only includes lower-case alphabet letters.
  - Sample input

```
3
accacacea
```

- Standard/Console Output
  - As output your program should print the password string.
  - Sample output

```
cac
```

Note

- Length-3 substrings and their frequencies

```
accacacea → acc, cca, cac, aca, cac, ace, cea
acc: 1
cca: 1
cac: 2
aca: 1
ace: 1
cea: 1
```

- Size of all length-30 strings

```
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaa ~
zzzzzzzzzzzzzzzzzzzzzzzzzzzzzz
```

→ $0 \sim 26^{30}-1$ (= 2,813,198,901,284,745,919,258,621,029,615,971,520,741,375)

Due Date

- Soft deadline: **December 4, 2020**
- Hard deadline: December 6, 2020
  - But, will be deducted 20% per one day from your original score

Submission date	Deduction rate
December 5	20%
December 6	40%
December 7	100%

Notice

- You may use `hash`, `unordered_map`, `hash_map`, etc in STL.
- You should observe the format of input & output exactly.

- You should submit a compressed file (**HW3_your-ID.zip**) containing the following two files to the web-site (https://klas.kw.ac.kr/).
  - **HW3_your-ID.hwp/doc/pdf** // report document
  - **HW3_your-ID.cpp/cc** // source code

Notice (cont'd)

- Source code
  - It should be compiled in **Visual Studio 2010 or higher, or g++**
    - **You should note your environment in your report.**
  - Your name and student ID should be noted at the top of your source files in the form of comment.
- Report
  - Free format
  - But, it must include several examples for testing your program and your own discussion.
  - It will be an important factor for getting a good score.
  - Mention your compiler