

# Data Structures (in C++)

2021 Fall

- PA 1. String Decoder -

Jinsun Park

Visual Intelligence and Perception Lab., CSE, PNU

# Problem Definition

## ■ String Decoder

- You will be given an input text file containing a number of encoded strings
- The rule to encode a string is defined as follows:

$$S_{enc} = n\{S_{unit}\}$$

- $S_{enc}$  : The encoded result
- $n$  : A *hexadecimal* number for the repetition ( $n \in [0, f]$ )
- $S_{unit}$  : A unit string to be repeated

## ■ Valid Input Characters

- $S_{unit}$  can contain: **0123456789abcdefghijklmnopqrstuvwxyz.,!:@#\$%^&\***
- $n$  must be the one of the followings: **0123456789abcdef**
- $n$  must not be the one of the followings: **ghijklmnopqrstuvwxyz.,!:@#\$%^&\***

# Problem Definition

## ■ String Decoder

- You will be given an input text file containing a number of encoded strings
- The rule to encode a string is defined as follows:

$$S_{enc} = n\{S_{unit}\}$$

- $S_{enc}$  : The encoded result
- $n$  : A *hexadecimal* number for the repetition ( $n \in [0, f]$ )
- $S_{unit}$  : A unit string to be repeated

## ■ Example results

Input	Output
3{hww}#	hwwhwwhww#
6{01b}5{x}	01b01b01b01b01bxxxxx
cws4{3h,}	cws3h,3h,3h,3h,
3{poc}l?po{5{y@}}	pocpocpocl?p
y{s9}3{*,}3{5*r}	ERROR: Invalid input

# Problem Definition

## ■ Requirements

- Implement your algorithm in a file named “*string\_decoder.cpp*”

- Your implementation will be compiled by the following command:

```
g++ string_decoder.cpp -o string_decoder
```

- You should read each line from the input file “*encoded.txt*”
- Each line from the input should be correctly decoded
- The decoded strings should be saved into an output file named “*decoded.txt*”
- Your program should get an input file name from the standard input stream
  - The command for the execution should be: `./string_decoder INPUT_FILE_PATH`
  - This command should generate a file “*decoded.txt*”

# Problem Definition

## ■ Notes

- Implement a string decoder using **class**
- Only the characters in [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, f] are allowed in front of the “{”
- If an input is invalid, the output should be “**ERROR: Invalid input**”
- You may use **recursion** and **STL containers** for the implementation

## ■ Evaluation Criteria

- Submitted implementations will be evaluated based on the “*decoded.txt*” file that is:
  - 1) generated using the released “*encoded.txt*” file
  - 2) generated using a hidden test file “*encoded\_eval.txt*”
- Readability
  - Please put detailed comments in your program as much as possible
- Plagiarism
  - **DO NOT COPY CODES FROM THE INTERNET**
  - **DO NOT COPY YOUR FRIEND’S CODES**