# C++ Advanced – Exam 2 (21 Apr 2019)

Write C++ code for solving the tasks on the following pages.

Code should compile under the C++11 standard.

Submit your solutions here: https://judge.softuni.bg/Contests/1608/CPlusPlus-Advanced-Retake-21-Apr-2019

Any code files that are part of the task are provided under the folder **Skeleton**.

Please follow the exact instructions on uploading the solutions for each task.

## Task 1 – Cipher

Your task is to write program, which decodes/parses a hidden message out of two provided data buffers following a special rule of orders. You are given the main() function, which reads three string values (as whole rows) of memory.

- The first string value indicate your command buffer. The buffer may only contain the letters 'l', 'r' and and digits in range [0-9] inclusive;
  - 'I' stands for 'left';
  - 'r' stands for 'right';
  - The digit represents an index out of either the 'left' or 'right' data buffers;
  - The syntax is always a letter ('I' or 'r') followed by a single digit [0-9];
  - Example: I2r0I9r8I3r3
- The second string value contains your 'left' data buffer (as a single row of data).
- The third string value contains your 'right' data buffer (as a single row of data).
- Note: commands buffer, left data buffer and right data buffer may be empty.
- Note: 2 commands buffer may have indexes listed that are not present in your data buffers!

You should implement the functions decryptMessage() and printResult() in another .cpp file. (For example Cipher.cpp)

The decryptMessage() function should construct a new string following the commands from your command buffer. (See more detailed description below).

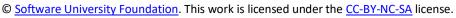
As a result of decryptMessage() – you should print a status message depending on the received ErrorCode in printResult() followed by a newline.

You should print:

- For successful message decoding the decoded message;
- For partial message decoding all part of the message that was successfully decoded followed by "Warning, buffer out-of-bound detected";
- For an empty command buffer "No input provided";

Note: again, an empty data buffer is not an error and should be treated normally.





















#### Example:

```
command - "r0l3l3l2"

leftBuffer - "lemon"

rightBuffer - "bye"

The result is decryptedMessage - "boom"
```

### Example 2:

```
command - "r0r1r1r6"

leftBuffer - "" (empty line)

rightBuffer - "oida"
```

The result is decryptedMessage – "oi Warning buffer out-of-bound detected"

Your task is to study the code and implement the function so that the code accomplishes the task described.

You should submit a single .zip file for this task, containing ONLY the files you created.

The Judge system has a copy of the other files and will compile them, along with your file, in the same directory.

### Restrictions

You are free to implement another function/functions that are used internally by the **decryptMessage()** and **printResult()**.

The command buffer will only contain the letters 'l' or 'r' and only ASCII digits from '0' to '9' inclusive;

The command buffer and data buffer size will be in the range [0, SIZE\_T\_MAX] inclusive;

Note: 'size\_t' and 'unsigned long long integer' are the same thing;

### **Examples**

Input	Output
TheBigBang bazinga	No input provided
r3r5r3r51916 whoElseBut Quagmire	giggite
r3r0r9	si Warning, buffer out-of-bound detected
it'sValid	















