**Individual Lab Exam : BSc 3rd Year**

In this project you will be creating a Word Processor that takes in input as a json and throws out the processed string. The following table illustrates the options that can be applied on the given string. The input to your Spring Application would look like the Json given below. You will have to implement ANY ONE of the options of your choice to secure a 100 in your lab exam.

Example input :

{

“data” : “Sairam”,

“options” : [“-c”, “upper”]

}

| **Option** | **Description** |
| --- | --- |
| -c <case> | Converts characters from the English alphabet to the specified letter case. The parameter case can be one of two string values: "upper" or "lower". The "upper" string parameter converts the characters to uppercase while the "lower" string parameter converts the characters to lowercase, regardless of what its original letter case was. **SHOULD NOT USE ANY PRE EXISTING STRING METHODS TO IMPLEMENT THIS.** |
| -e <shift\_amount> | Encodes the text using a cipher that replaces each alphabetic character with a new character, ch, that is at shift\_amount positions from ch in the English alphabet (wrapping from "z" to "a" or, similarly, from "Z" to "A"). shift\_amount is an integer parameter in the inclusive range of ﹣󠀭25 to 25. The cipher respects capitalization and ignores non-English alphabetic characters (i.e. special characters or numbers). |
| -x <substring> | Exclude (or skip) the lines of the input file containing the string parameter substring. The search for substring literal is case-sensitive. Expect a newline character (System.*lineSeparator*()) in the input string. |
| -a | Encodes characters with its ASCII value followed by a space. Only printable ASCII characters (those in the inclusive range 32 to 126) should be converted. This option takes no parameters. Hint : Use character arrays instead of Strings. |
| -p <prefix> | Adds the string parameter prefix at the beginning of each line. A line is considered when it is separated by new line characters. |

#### 

IMPLEMENT ANY ONE OF THE OPTIONS PROVIDED ABOVE TO SECURE A 100 IN YOUR LAB EXAM.

string.charAt()

char[]

string.split()

Example input :

{

“data” : “Sairam”,

“options” : [“-c”, “upper”]

}

Example output : This is the json your spring application would throw.

{

“output” : “SAIRAM”

}

Example input :

{

“data” : “SaiRam”,

“options” : [“-c”, “lower”]

}

Example output : This is the json your spring application would throw.

{

“output” : “sairam”

}

Example input :

{

“data” : “Sairam”,

“options” : [“-a”]

}

Example output : This is the json your spring application would throw.

{

“output” : “83 97 105 114 97 109”

}

Example input :

{

“data” : “SaiRam”,

“options” : [“-e”, “0 to 25 or -25 to 0”]

}

Example output : This is the json your spring application would throw.

{

“output” : “TbjSbn”

}

Example input :

{

“data” : “SaiRam”,

“options” : [“-e”, “2”]

}

Example output : This is the json your spring application would throw.

{

“output” : “UckTco”

}

Example input :

{

“data” : “SaiRam \n abcsairam \n cdeSairam”,

“options” : [“-x”, “abc”]

}

Example output : In this case the second line which contains the substring abc gets deleted. \n is the new line character

{

“output” : “SaiRam \n cdeSairam”

}

Example input :

{

“data” : “SaiRam\n This is a lab test”,

“options” : [“-p”, “abs=c”]

}

Example output : This is the json your spring application would throw.

{

“output” : “- Sairam\n - This is a labtest”

}

SaiRam

This is a lab test

This is the way it should look.

- Sairam

- This is a lab test