Logic test explanation

In `alpha_string.py` file, in `server` folder, i applied the following logic:

1. First, I consider that for every set of Roman letters there is, necessarily, a non-Roman letter separating each sequence.
    - So, for the following string: `AXXBLX` , the separator letter is `A` and `B` .
2. I wrote down all roman letters in a python dictionary, like this:

```
VALUE_ROMAN_CHARACTERS = {
    "I": 1,
    "V": 5,
    "X": 10,
    "L": 50,
    "C": 100,
    "D": 500,
    "M": 1000,
    "IV": 4,
    "IX": 9,
    "XL": 40,
    "XC": 90,
    "CD": 400,
    "CM": 900
}
```

3. Given a sequence with pattern: $(S_k s_n S_{k+1})^*$, where:
    - S_k is k-th roman letters set;
    - s_n is a separator letter between S_k and S_{k+1};

`if` : $S_k \in [I, V, X, L, C, D, M, IV, IX, XL, XC, CD, CM]$

`then` : $value =$ VALUE_ROMAN_CHARACTERS[S_k]

(That is, the set has only one of the Roman letters present in the dictionary)

`else` :

$$value = \sum_{i=0}^{len(S_k)} VRC[S_k[i]]$$

, where $VRC =$ VALUE_ROMAN_CHARACTERS

(That is, the sum of the conversions of each Roman letter in the set)