Lab 4.5.6 Text manipulation: plain-text formatting

Objectives

Familiarize the student with:

- problems with manipulating free-form text.

Scenario

When dealing with plain-text mediums, the formatting options are fairly limited compared to modern WYSIWYG rich-text editors.

A common practice is to emphasize words or text fragments using the asterisk character ("*") or the underline character ("_").

Write a program that will change this style of formatting according to the following rules:

- all alphabet characters surrounded by asterisks will be made upper-case, and the asterisks will be removed, i.e. "this is *it*" will be changed to "this is IT";
- all characters surrounded by underscores will be separated by additional spaces, and the underscores will be changed to spaces, i.e. "is _this_ it?" will be changed to "is t h i s it?";
- an underscore or asterisk will be ignored if no matching character has been found until the end of the string;
- an underscore will be ignored if an asterisk has been encountered before a matching underscore;
- an asterisk will be ignored if an underscore has been encountered before a matching asterisk.

```cpp
#include <string>
#include <iostream>

int main()
{
    std::string sentence;
    std::getline(std::cin, sentence);

    // manipulate the sentence here

    std::cout << sentence << "\n";
}}
```

Example input

```
What do you *mean*?
```

Example output

```
What do you MEAN?
```

Example input

```
This is _really_ important!
```

Example output

```
This is  r e a l l y  important!
```

Example input

```
*This* one _might *be _quite tricky_*, if you know what I mean.
```

Example output

```
THIS one might be  q u i t e   t r i c k y , if you know what I mean.
```