

Частное образовательное учреждение высшего образования
“ЮЖНЫЙ УНИВЕРСИТЕТ (ИУБиП)”

Индивидуальное задание №1

по дисциплине: «Практикум по программированию»

Тема 1. Обработка строк
Вариант 15.

Работу выполнил:
студент группы ИД 401
Недвигин С. В.

Проверил:
преподаватель к.т.н. Сафонова С. А.

Задание: переписать в новую строку те слова, длина которых четная, в 2-х вариантах: с использованием встроеного типа и с использованием класса string

Листинг кода.

```
#include <iostream>
#include <sstream>
#include <vector>

int main() {
    std::string inputString = "The quick brown fox jumps over the lazy dog";
    std::istringstream iss(inputString);
    std::vector<std::string> words;

    do {
        std::string word;
        iss >> word;
        words.push_back(word);
    } while (iss);

    std::string resultString;
    for (const std::string& word : words) {
        if (word.length() % 2 == 0) {
            resultString += word + " ";
        }
    }

    std::cout << "input: " << inputString << std::endl;
    std::cout << "result: " << resultString << std::endl;

    return 0;
}

#include <iostream>
#include <sstream>
#include <vector>

int main() {
    std::string inputString = "The quick brown fox jumps over the lazy dog";
    std::istringstream iss(inputString);
    std::vector<std::string> words;
```

```

do {
    std::string word;
    iss >> word;
    words.push_back(word);
} while (iss);

std::string resultString;
for (const std::string& word : words) {
    if (word.length() % 2 == 0) {
        resultString += word + " ";
    }
}

std::cout << "input: " << inputString << std::endl;
std::cout << "result: " << resultString << std::endl;

return 0;
}

```

Скриншот выполнения работы представлен на рисунках 1 и 2.

```

main.cpp
1 #include <iostream>
2 #include <sstream>
3 #include <vector>
4
5 int main() {
6     std::string inputString = "The quick brown fox jumps over the lazy dog";
7     std::istringstream iss(inputString);
8     std::vector<std::string> words;
9
10    do {
11        std::string word;
12        iss >> word;
13        words.push_back(word);
14    } while (iss);
15
16    std::string resultString;
17    for (const std::string& word : words) {
18        if (word.length() % 2 == 0) {
19            resultString += word + " ";
20        }
21    }
22
23    std::cout << "input: " << inputString << std::endl;
24    std::cout << "result: " << resultString << std::endl;
25
26    return 0;
27 }

```

```

Console
Run
input: The quick brown fox jumps over the lazy dog
result: over lazy
10s on 09:43:31, 01/29

```

Рисунок 1

The image shows a C++ IDE with two panels. The left panel displays the source code for a program named `main.cpp`. The code defines a `CustomString` class with a constructor, a `processAndPrint` method, and a `main` function. The `processAndPrint` method uses `std::istringstream` to split the input string into words and prints them with a space between every second word. The `main` function initializes a `CustomString` object with the string "The quick brown fox jumps over the lazy dog" and calls `processAndPrint`. The right panel shows the console output, which matches the expected result: "input: The quick brown fox jumps over the lazy dog" followed by "result: over lazy".

```
1 #include <iostream>
2 #include <sstream>
3
4 class CustomString {
5 public:
6     CustomString(const std::string &str) : content(str) {}
7
8     void processAndPrint() {
9         std::istringstream iss(content);
10        std::string word;
11
12        std::cout << "input: " << content << std::endl;
13        std::cout << "result: ";
14
15        while (iss >> word) {
16            if (word.length() % 2 == 0) {
17                std::cout << word << " ";
18            }
19        }
20
21        std::cout << std::endl;
22    }
23
24 private:
25    std::string content;
26 };
27
28 int main() {
29     std::string inputString = "The quick brown fox jumps over the lazy dog";
30     CustomString customStr(inputString);
31     customStr.processAndPrint();
32
33     return 0;
34 }
```

Console Output:

```
input: The quick brown fox jumps over the lazy dog
result: over lazy
```

Рисунок 2

Вывод: в ходе выполнения работы освоены программы простейшей структуры; приобретены навыки проектирования и обработки строк в языке программирования C++. Работа с встроенным типом строки (`std::string`) позволила эффективно разбивать текст на отдельные слова и проводить операции над ними. Также использование класса строки (`string`) продемонстрировало принципы объектно-ориентированного программирования.