**Основное задание:**

Предложение. Хранит строку переменной длины представляющую собой предложение.

Методы: добавить слово, удалить слово, вставить слово, количество букв, количество слов, самое длинное слово,самое короткое слово, есть ли в предложении заданное слово, слово под заданным номером, равны ли два предложения.

**Выполнил:** Пазов В.С.

**Проверил:** Дзюба Е.А.

**Выполненное:**

1) Классы. Диаграммы классов.  
2) Конструкторы-деструкторы. Три конструктора: по умолчанию, с параметром, копирования. (для своего класса делать)

**Код:**

// Предложение.

// Хранит строку переменной длины представляющую собой предложение.

// Методы: добавить слово, удалить слово, вставить слово, количество букв,

// количество слов, самое длинное слово, самое короткое слово,

// есть ли в предложении заданное слово, слово под заданным номером, равны ли два предложения.

// Выполненное:

// 1)Классы. Диаграммы классов.

// 2) Конструкторы-деструкторы. Три конструктора: по умолчанию, с параметром, копирования. (для своего класса делать)

#include <iostream>

using namespace std;

class Sentence {

private:

string sentence;

public:

Sentence() {

sentence = "Hello, world!";

cout << endl << "Конструктор без параметров: ";

}

Sentence(string str) {

sentence = str;

cout << endl << "Конструктор с параметрами: ";

}

Sentence(const Sentence& other) {

sentence = other.sentence;

}

void Print() {

cout << endl << this->sentence << endl;

}

void Add\_Word() {

cout << endl << "Добавление слова:" << endl;

string word\_to\_add;

int begin\_end = 1;

cout << "Введите слово которое вы хотите добавить: " << endl;

cin >> word\_to\_add;

cout << "Добавить слово в начало текста - 1, если в конец - 2\n" << endl;

cin >> begin\_end;

switch(begin\_end) {

case 1:

word\_to\_add = word\_to\_add;

sentence = word\_to\_add + " " + sentence;

cout << sentence << endl;

break;

case 2:

sentence += " " + word\_to\_add;

cout << sentence << endl;

break;

default:

cout << "Неправильно введено число" << endl;

break;

}

}

void Delete\_Word() {

cout << endl << "Удаление слова: " << endl;

string new\_sentence = sentence;

char word\_to\_delete(' ');

int count\_spaces = 0;

int total\_spaces = 0;

cout << "Введите номер слова которое вы хотите удалить: " << endl;

cin >> total\_spaces;

total\_spaces - 2;

for (int i = 0; i < new\_sentence.length(); i++) {

if (new\_sentence[i] != word\_to\_delete) {

count\_spaces == count\_spaces;

} else {

count\_spaces++;

if (count\_spaces == total\_spaces) {

new\_sentence[i++];

}

while (new\_sentence[i] != word\_to\_delete && new\_sentence[i] != NULL) {

new\_sentence.erase(i, 1);

}

}

}

cout << "Исходное предложение: " << sentence << endl;

cout << "Предложение с удаленным словом: " << new\_sentence << endl;

}

void Number\_Of\_Letters() {

cout << endl << "Подсчет количества букв:" << endl;

int count\_letters = 0;

char Space(' ');

for (int i = 0; i < sentence.length(); i++) {

if (sentence[i] != Space) {

count\_letters++;

}

}

cout << "Исходное предложение: " << sentence << endl;

cout << "Количество букв: " << count\_letters << endl;

}

void Number\_Of\_Words() {

cout << endl << "Подсчет количества слов:" << endl;

char Space(' ');

int count\_word = 0, new\_const = 0;

for (int i = 0; i < sentence.length(); i++) {

if (sentence[i] == Space && sentence[i + 1] == Space) {

sentence[i++];

count\_word++;

} else {

if (sentence[i] == Space) {

count\_word++;

}

}

}

if (sentence[0] == Space) {

count\_word--;

}

new\_const = sentence.length();

if (sentence[new\_const - 1] != Space) {

count\_word++;

}

cout << "Исходное предложение: " << sentence << endl;

cout << "Количество слов: " << count\_word << endl;

}

void Longest\_Word() {

cout << endl << "Подсчет длины слов:" << endl;

char Space(' ');

int mass[10];

int start\_id, count\_letters = 0, max\_count = 0, count\_letters\_2 = 0, otd;

int j = 0;;

for (int i = 0; i < sentence.length(); i++) {

otd = sentence[i];

count\_letters = 0;

while (sentence[i] != Space && i < sentence.length()) {

sentence[i++];

count\_letters++;

otd = sentence[i - count\_letters];

}

if (otd != Space) {

mass[j] = count\_letters;

if (mass[j - 1] < mass[j]) {

max\_count = i - count\_letters;

}

mass[j++];

}

}

cout << "Исходное предложение: " << sentence << endl;

cout << endl << "Самое длинное слово: " << endl;

for (int i = max\_count; i < sentence.length(); i++) {

while (sentence[i] != Space && i < sentence.length()) {

cout << sentence[i];

sentence[i++];

}

}

}

};

int main() {

setlocale(LC\_ALL, "RUS");

Sentence withoutParams;

withoutParams.Print();

Sentence withParams("word1 word22 word333");

withParams.Print();

withParams.Add\_Word();

withParams.Delete\_Word();

withParams.Number\_Of\_Letters();

withParams.Number\_Of\_Words();

withParams.Longest\_Word();

cout << endl;

cout << endl <<"Завершение программы" << endl;

cout << endl;

return 0;

};

**Результат:**

