Министерство науки и высшего образования Российской Федерации Федеральное государственное автономное образовательное учреждение высшего образования

**Пермский национальный исследовательский политехнический университет**

Факультет Электротехнический Кафедра ИТАС

Специальность Мехатроника и Робототехника

**ОТЧЁТ**

**о лабораторной работе №6**

Строки

|  | Выполнил:  Студент группы МИР-21-2Б  Степанова И.В.  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Проверил:  Доцент кафедры ИТАС  Полякова О.А.  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| --- | --- |

**Пермь 2022**

Цель лабораторной работы: изучение символьных и строковых переменных и способов их обработки на языке Си.

Постановка задачи:

Задана строка, состоящая из символов. Символы объединяются в слова. Слова друг от друга отделяются одним или несколькими пробелами. В конце текста ставится точка. Текст содержит не более 255 символов. Выполнить ввод строки, используя функцию gets(s) и обработку строки в соответствии со своим вариантом.  
 Анализ решения:  
Для начала попробуем получить строку при помощи функции Gets(s) и у нас ничего не выйдет, т.к. такая функция отсутствует в с++. Уже в 2016 году данная функция была удалена из C++ из-за её проблемы с безопасностью. По этому мы заменим её на похожую функцию getline().

После получения строки можем нарезать ее на слова используя цикл.

После сравниваем каждое слово и находим самое короткое и самое длинное

Код программы:

#include <iostream>

#include <string>

#include <vector>

using namespace std;

int main()

{

string mas, g, prom;

bool running = true;

char p[255];

int words = 0, b = 0, min = 1000, max = 0, min\_pos, max\_pos;

vector <string> word;

getline(cin, mas, '.');

for (int i = 0; i < 255; i++) {

if (mas[i] != ' ') {

g.push\_back(mas[i]);

}

else {

word.push\_back(g);

g.clear();

words++;

}

if (mas[i + 1] == NULL) {

word.push\_back(g);

g.clear();

words++;

goto next\_l;

}

}

next\_l:

cout << "\n";

for (int i = 0; i < words; i++) {

prom = word[i];

if (prom.length() < min) {

min = prom.length();

min\_pos = i;

}

else if (prom.length() > max) {

max = prom.length();

max\_pos = i;

}

prom.clear();

}

cout << word[max\_pos] << "- it's longest word! \n";

cout << word[min\_pos] << "- it's smallest word! \n";

return 0;

}

Скрины решения:

