

**МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ**

**Національний технічний університет України**

**«Київський політехнічний інститут імені Ігоря Сікорського»**

**Кафедра інформаційних систем та технологій**

**Звіт**

**з лабораторної роботи № 8**

**«Робота з рядками мовою C++»**

**з дисципліни**

**«Програмування – 1. Основи програмування»**

**Варіант № 7**

**Перевірила:**

**ст. викладач, Проскура Світлана  
Леонідівна**

**Виконав:**

**Фундерат Денис**

**Студент гр. ІС-12 , ФІОТ**

**1 курс,**

**залікова книжка № ІС-1230**

**Київ 2021**

## Лабораторна робота № 7

### Тема: Робота з рядками мовою C++

#### Мета :

1. Ознайомитися з особливостями роботи з рядками;
2. Опанувати технологію застосування та обробки рядків;
3. Навчитися розробляти алгоритми та програми із застосуванням рядків.

#### Завдання

7. У рядку символів визначити слова-паліндроми, які однаково читаються справа наліво і навпаки. Знайти найбільшу довжину цих слів. Якщо таких слів немає, то з кожного слова на непарній позиції вилучити голосні та надрукуйте їх.

```
#include <iostream>

#include <vector>

using namespace std;

// У рядку символів визначити слова-паліндроми, які однаково читаються справа наліво і
навпаки.

// Знайти найбільшу довжину цих слів.

// Якщо таких слів немає, то з кожного слова на непарній позиції вилучити голосні та
надрукуйте їх.

void palindromeSearch();

void lengthCounting();

void palindromeNotFound();

//const char str[] = "lorem ipsum dolor sit amet"; // without palindrome words
```

```
const char str[] = "lorom tenet olso edle olololo ele temp bed dad gg"; // with palindrome
words

vector<char> arr;

vector<char> vowels = {'a', 'e', 'i', 'o', 'u'};


int main() {

    cout << "hole string: " << str << endl;

    palindromeSearch();

    if (arr.size() == 0) palindromeNotFound();

    else lengthCounting();

    return 0;
}


void palindromeSearch() {

    vector<char> palindWords;

    int k;

    int count = 0;

    for (int i = 0; i <= strlen(str); i++) { // перебираемо по буквах рядок

        if (*(str+i) == ' ' || i == strlen(str)) {

            k = palindWords.size();
```

```

        count = 0;

        for (int j = 0; j < palindWords.size(); j++) {

            k--;

            if (palindWords[j] == palindWords[k]) count++;

        }

        if (count == palindWords.size()) {

            for (int j = 0; j < palindWords.size(); j++) {

                arr.push_back(palindWords[j]);

            }

            arr.push_back(' ');

        }

        if (i != strlen(str)) palindWords = {};

    }

    else if (*(str+i) != ' ' || i != strlen(str)) {

        palindWords.push_back(*(str+i));

    }

}

}

```

```

void lengthCounting() {

    vector<int> wordLen_arr;

    int wordLen = 0;

```

```
cout << "palindrome words: ";

for (int i = 0; i < arr.size(); i++) {

    cout << arr[i]; // виводимо в консоль слова-паліндроми
```

```
    if (arr[i] == ' ') { // визначаємо довжину кожного слова
```

```
        wordLen_arr.push_back(wordLen);
```

```
        wordLen = 0;
```

```
    }
```

```
    else wordLen++;
```

```
}
```

```
cout << endl;
```

```
int bigInt;
```

```
for (int i = 0; i < wordLen_arr.size(); i++) { // визначаємо найбільшу довжину серед слів
```

```
    if (i == 0) bigInt = wordLen_arr[i];
```

```
    if (bigInt < wordLen_arr[i+1] && i != wordLen_arr.size() - 1) {
```

```
        bigInt = wordLen_arr[i+1];
```

```
    }
```

```
}
```

```
cout << "the biggest length: " << bigInt << " "; // виведення в консоль найбільшої довжини
```

```
vector<char> bigInt_word;
```

```

for (int i = 0; i < arr.size(); i++) { // визначення слова з найбільшою довжиною

    if (arr[i] == ' ') {

        if (bigInt_word.size() == bigInt) break;

        else bigInt_word = {};

    }

    else bigInt_word.push_back(arr[i]);

}

for (int i = 0; i < bigInt_word.size(); i++) { // виведення слова з найбільшою довжиною

    cout << bigInt_word[i];

}

}

```

```

void palindromeNotFound() {

    vector<char> vowels_cur;

    int count = 0;

    for (int i = 0; i < strlen(str); i++) {

        if (*(str+i) == ' ' || i == strlen(str) - 1) {

            count++;

            if (count % 2 != 0) {

```

```
        for (int j = 0; j < vowels.size(); j++) { // визначаємо чи є символ голосною
буквою

            if (*(str+i) == vowels[j]) vowels_cur.push_back(*(str+i));

        }

        for (int j = 0; j < vowels_cur.size(); j++) { // виводимо в консоль
визначені голосні

            cout << vowels_cur[j] << " ";

        }

        cout << endl;

    }

    else if (count % 2 == 0 && i != strlen(str) - 1) vowels_cur = {};

}

else if (*(str+i) != ' ' || i != strlen(str) - 1) {

    for (int j = 0; j < vowels.size(); j++) {

        if (*(str+i) == vowels[j]) vowels_cur.push_back(*(str+i));

    }

}

}

}
```

## Результат виконання:

```
hole string: lorom tenet olso edle olololo ele temp bed dad gg
palindrome words: tenet olololo ele dad gg
the biggest length: 7 olololo
Process finished with exit code 0
```

```
hole string: lorem ipsum dolor sit amet
o e
o o
a e

Process finished with exit code 0
```