Белорусский государственный технологический университет

Факультет информационных технологий

Кафедра программной инженерии

Лабораторная работа 13

По дисциплине «Основы алгоритмизации и программирования»

На тему «Обработка символьной информации»

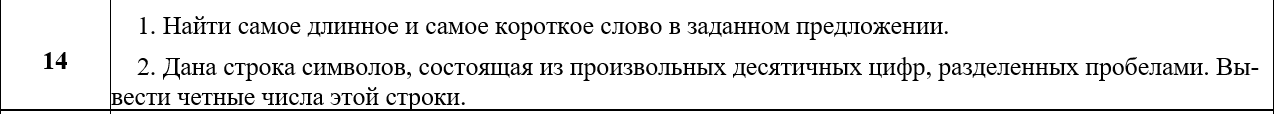
Выполнила:

Студентка 1 курса 7 группы

Шинкевич Марина Дмитриевна

Преподаватель: асс. Андронова М.В.

2023, Минск



#include <iostream>

#include <string>

#include <Windows.h>

using namespace std;

int main() {

setlocale(LC\_ALL, "rus");

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

string sentence;// объявляет переменную типа string

getline(cin, sentence);//считывает строку и сохраняет ее в переменной sentence.

string min\_word = "", max\_word = "";//переменные для хранения минимальной и максимальной длины слов,а также самих слов.

int min\_length = sentence.length(), max\_length = 0;

int length = 0, start = 0;

for (int i = 0; i <= sentence.length(); i++) {//Цикл for перебирает каждый символ во введенном предложении, начиная с индекса 0 и заканчивая индексом, равным длине предложения

if (sentence[i] == ' ' || i == sentence.length()) {//На каждой итерации цикла проверяется, является ли текущий символ пробелом или является ли индекс текущего символа равным длине предложения.

length = i - start;//вычислить длину этого словам

if (length > max\_length) {//длина сравнивается с минимальной и максимальной длиной

max\_length = length;

max\_word = sentence.substr(start, length);// подстрока, соответствующая текущему слову, извлекается из исходной строки sentence с использованием функции substr, и сохраняется в переменную

}

if (length < min\_length) {

min\_length = length;

min\_word = sentence.substr(start, length);

}

start = i + 1;

}

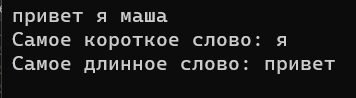
}

cout << "Самое короткое слово: " << min\_word << endl;

cout << "Самое длинное слово: " << max\_word << endl;

return 0;

}



#include <iostream>

#include <string>

#include <Windows.h>

using namespace std;

int main() {

setlocale(LC\_ALL, "rus");

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

string input;

getline(cin, input);//считывание введенной строки с помощью функции getline и сохранение в переменную input

int numbers[6];//объявление массива

int count = 0;

for (int i = 0; i < input.length(); i++) {//цикл for, который перебирает каждый символ во введенном предложении

if (isdigit(input[i])) {//проверка, является ли текущий символ цифрой.

numbers[count] = input[i] - '0';//преобразование символа-цифры в числовое значение и сохранение его в массив

count++;

}

}

int\* ptr = numbers;// объявление указателя ptr, который указывает на начало массива numbers.

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

if (\*ptr % 2 == 0) {//проверка, является ли текущее число четным.

cout << "Чётный:" << \*ptr << " ";//вывод сообщения о том, что число четное, и само число.

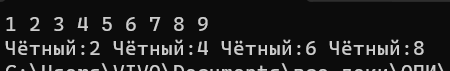
}

ptr++;

}

return 0;

}



Дополнительные задания:

1) В заданной последовательности слов найти все слова, имеющие заданное окончание.

#include <iostream>

#include <string>

#include <Windows.h>

using namespace std;

int main() {

setlocale(LC\_ALL, "ru");

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

const int maxSize = 100;

int n = 0;

char arr[maxSize];

char symbol;

cout << " Введите слова : " << endl;

cin.getline(arr, maxSize); // считывание строки из ввода пользователя в массив arr.

cout << "Введите окончание: " << endl;

cin >> symbol;

int a = strlen(arr); //вычисление длины строки в массиве arr и сохранение в переменной a.

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

n++;

if (arr[i] == ' ' || arr[i] == '\0') {//условие проверки, если текущий символ равен пробелу или концу строки.

n--;// уменьшение значения переменной

if (arr[i - 1] == symbol) {//условие проверки, если предыдущий символ равен символу symbol.

for (int g = i - n; arr[g] != '\0' && arr[g] != ' '; g++) {//начало цикла с итерацией от i-k до конца строки или до пробела.

cout << arr[g];

}

cout << ' ';

}

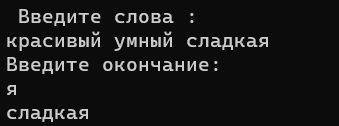
n = 0;

}

}

return 0;

}



3) Из заданного предложения удалить те слова, которые уже встречались в предложении раньше

#include <iostream>

#include <vector>

#include <Windows.h>

using namespace std;

int main() {

setlocale(LC\_ALL, "ru");

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

char arr[] = "Меня зовут зовут катя катя";

int k = 0;

for (int i = 0; arr[i] != '\0'; i++) {//В цикле подсчитывает количество слов в строке и выделяет память для массива строк words.

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

k++;

}

}

k++;

string\* words = new string[k];//создается динамический массив words типа string с размером k.

int j = 0;

for (int i = 0; i < k; i++) {//икл for, чтобы перебирать каждый элемент в массиве words

while (arr[j] != ' ' && arr[j] != '\0') {//цикл while добавляет символы из массива arr в элемент массива words[i], пока не встретится пробел или конец строки

words[i] = words[i] + arr[j];

j++;

}

j++;

}

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

bool rep = 0;

for (int g = 0; g < i; g++) {//переменная g используется для сравнения текущего элемента с предыдущими элементами массива.

if (words[i] == words[g]) {

rep = 1;//если повторение найдено, то rep=1

break;

}

}

if (rep == 0) { //Проверяет, есть ли повторяющиеся слова в массиве words

cout << words[i] << ' ';

}

}

delete[] words;// Освобождает память, выделенную для массива words.

return 0;

}



4) Отредактировать заданное предложение, удаляя из него все слова с нечетными номерами и переворачивая слова с четными номерами. Пример: HOW DO YOU DO преобразовать в OD OD.

#include <iostream>

#include <string>

using namespace std;

int main() {

setlocale(LC\_ALL, "ru");

const int Size = 100;

char arr[Size];

cout << "Предложение: " << endl;

cin.getline(arr, Size);

int p = 0;//Переменная p будет использоваться для подсчета количества слов в предложении.

int a = strlen(arr); //находит длину строки в массиве arr с помощью функции strlen и сохраняет результат в переменной a

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

if (arr[i] == ' ' || arr[i] == '\0') {// проверяет, является ли текущий элемент массива пробелом или концом строки.

p++;

if (p % 2 == 0) { //проверяет, является ли значение переменной p четным числом.

for (int g = i - 1; arr[g] != '\0' && arr[g] != ' '; g--) {//перебирает символы в обратном порядке от текущего элемента до пробела или конца строки.

cout << arr[g];

}

cout << ' ';

}

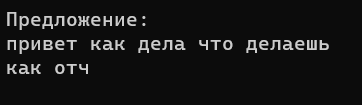
}

}

cout << '\n';

return 0;

}

****