|  |  |  |
| --- | --- | --- |
| Министерство науки и высшего образования  Российской Федерации | | |
| Федеральное государственное бюджетное  образовательное учреждение высшего образования | | |
| «Новосибирский государственный технический университет» | | |
|  | | |
| Кафедра прикладной математики | | |
|  | | |
| Практическое задание № 4 | | |
| по дисциплине «Основы программирования» | | |
|  | | |
| **Обработка текстовой информации** | | |
|  | | |
|  | Факультет: | ПМИ |
| Группа: | ПМ-93 |
| Бригада: | 2 |
| Студенты: | Воробьева Вероника, |
|  | Назаров Максим, |
|  | Чипкинеева Юлия |
| Преподаватели: | Тракимус Юрий Викторович, |
|  | Хиценко Валентина Павловна |
|  | | |
| Новосибирск | | |
| 2020 | | |

1. **Условие задачи**

Дана непустая последовательность слов (не более 50), в каждом слове не более восьми символов. Слова разделены пробелом, за последним словом точка. Найти те слова, которые отличаются от последнего слова и удовлетворяют следующему условию: слово симметрично.

1. **Анализ задачи**

Входные данные:

Выходные данные:

n [0;50]

Решение:

при

повторять пока :

если Аi  Ak и Аi - симметричното:

Bj = Аi,

j = j +1

i = i + 1

1. **Алгоритм решения задачи**

Вывод B, j

Конец

-

+

Начало

Ввод А

i = 0,

j = 0

i<(k-1)?

,

j = j +1

i = i +1

1. **Текст программы**

#include <stdio.h>

#include <conio.h>

#include <string.h>

#include <string>

#include <windows.h>

int IsSymmetricStr(char s[]);

const int M = 51;

const int N = 9;

void main()

{

char A[M][N];

int k, i, j;

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

printf\_s("\n Введите последовательность: ");

i = 0; j = 1;

while ((i < M) && (j == 1))

{

scanf\_s("%s", A[i], N + 1);

if (A[i][strlen(A[i]) - 1] == '.') { j = 0; }

i++;

}

i--;

if (A[i][strlen(A[i]) - 1] == '.') { A[i][strlen(A[i]) - 1] = '\0'; }

k = i;

for (j = 0; j < k; j++) {

if (strcmp(A[j], A[k])!= 0) {

if (IsSymmetricStr(A[j]) == 1) {printf\_s("%s ", A[j]);}

}

}

printf\_s("\n");

}

int IsSymmetricStr(char s[]) {

int b = 1;

int i = 0, k;

k = strlen(s);

while ((i < k / 2) && (b == 1)) {

if (s[i] != s[k - i - 1]) { b = 0; }

i++;

}

return (b);

}

1. **Наборы тестов**

|  |  |  |
| --- | --- | --- |
| **№** | **Входные данные** | **Назначение теста** |
| **1** | А = {как.} | Проверить работу программы для минимального количества элементов в А |
| **2** | А = {ара в а к ара.} | Проверить работу программы для элементов с минимальным количеством символов |
| **3** | А = {ппппп птп стстс кек как с кек.} | Проверить работу программы для элементов А различного количества символов |
| **4** | А = {с аааааааа с.} | Проверить работу программы для элемента с максимальным количеством символов |

1. **Результаты работы программы**

|  |  |
| --- | --- |
| **№** | **Ввод/вывод программы** |
| **1** | А = {как.}; |
| **2** | А = {ара в а к ара.};  в а к |
| **3** | А = { ппппп птп стстс кек как с кек.};  ппппп птп стстс как с |
| **4** | А = {с аааааааа с.};  аааааааа |