|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **Информатика и системы управления**

КАФЕДРА **Компьютерные системы и сети (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.03.01 Информатика и вычислительная техника**

**Отчет**

|  |  |
| --- | --- |
| **по домашнему заданию №** | 2 |

**Название:**

Программирование на С++

**Дисциплина:** Объектно-ориентированное программирование

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-22Б |  |  | С.В. Астахов |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  |  |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2020

**Задание 1**

Написать программу, определяющую, сколько из чисел в диапазоне от 0 до n (n вводится с клавиатуры), сами являясь палиндромами, при возведении в квадрат, образуют палиндромы.

Например: 222 -> 484 .

Вывести на печать найденные числа, их квадраты и количество найденных чисел. При программировании использовать функции.

**Исходный код**

(Unit1.pas)

*#include <iostream>*

*#include <stdlib.h>*

*using namespace std;*

*int checker(char\* sa, bool\* cond);*

*int main()*

*{*

*puts("Enter number");*

*int n;*

*scanf\_s("%d", &n);*

*char\* s1, \* s2;*

*s1 = new char[10];*

*s2 = new char[40];*

*for (int i = 0; i <= n; i++) {*

*\_itoa\_s(i, s1, 10, 10);*

*\_itoa\_s(i \* i, s2, 40, 10);*

*bool fl = true;*

*checker(s1, &fl);*

*checker(s2, &fl);*

*if (fl) {*

*printf("nice num: %d %d \n", i, i \* i);*

*}*

*}*

*delete[] s1;*

*delete[] s2;*

*}*

*int checker(char \*sa, bool \* cond)*

*{*

*for (int j = 0; j <= strlen(sa) / 2; j++) {*

*if (\*(sa + j) != \*(sa + strlen(sa) - j - 1)) {*

*\*cond = false;*

*}*

*}*

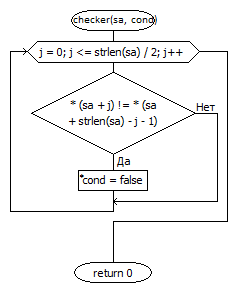
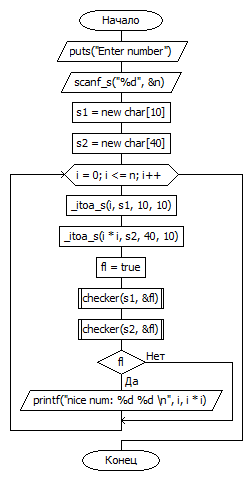
*return 0;*

*}*

**Тесты**

|  |  |  |
| --- | --- | --- |
| Входные данные | Ожидаемые выходные данные | Выходные данные |
| 22 | nice num: 0 0  nice num: 1 1  nice num: 2 4  nice num: 3 9  nice num: 11 121  nice num: 22 484 | nice num: 0 0  nice num: 1 1  nice num: 2 4  nice num: 3 9  nice num: 11 121  nice num: 22 484 |
| 20 | nice num: 0 0  nice num: 1 1  nice num: 2 4  nice num: 3 9  nice num: 11 121 | nice num: 0 0  nice num: 1 1  nice num: 2 4  nice num: 3 9  nice num: 11 121 |
| 100 | nice num: 0 0  nice num: 1 1  nice num: 2 4  nice num: 3 9  nice num: 11 121  nice num: 22 484  nice num: 101 10201  nice num: 111 12321  nice num: 121 14641  nice num: 202 40804  nice num: 212 44944 | nice num: 0 0  nice num: 1 1  nice num: 2 4  nice num: 3 9  nice num: 11 121  nice num: 22 484  nice num: 101 10201  nice num: 111 12321  nice num: 121 14641  nice num: 202 40804  nice num: 212 44944 |

**Схема алгоритма**

****

**Задание 2**

Дана последовательность строк. Каждая строка состоит из слов, разделенных пробелами, в конце – точка. Слова в строке образуют пары: каждое первое слово – заменяемое, каждое второе – замещающее. Написать программу, обеспечивающую ввод строк и их корректировку. Корректировка заключается в замене всех заменяемых слов замещающими. Вывести на печать исходную и скорректированную последовательности строк.

**Исходный код**

*#include <iostream>*

*int main()*

*{*

*int n = 0;*

*char\* mstr[30];*

*for (int i = 0; i < 30; i++) {*

*mstr[i] = NULL;*

*}*

*char\* p, \* q,\*tr,\*st, \*buf;*

*buf = NULL;*

*q = NULL;*

*puts("Enter kol-vo of strings \n");*

*scanf\_s("%d", &n, 10);*

*char\* sp;*

*sp = NULL;*

*sp = new char;*

*gets\_s(sp,1);*

*for (int i = 0;i < n;i++) {*

*mstr[i] = new char[30];*

*gets\_s(mstr[i], 30);*

*}*

*puts("\n ECHO \n");*

*for (int i = 0;i < n;i++) {*

*printf("\n %s", mstr[i]);*

*}*

*puts("\n \n CCORRECTED \n");*

*for (int i = 0;i < n;i++) {*

*tr = mstr[i];*

*while (tr != NULL) {*

*p = strtok\_s(tr, " ", &q);*

*q = strtok\_s(q, " ", &tr);*

*if ((p == NULL) || (q == NULL)) {*

*break;*

*}*

*st = tr;*

*printf\_s("%s %s ", q, q);*

*}*

*puts("\n");*

*delete[] mstr[i];*

*// delete*

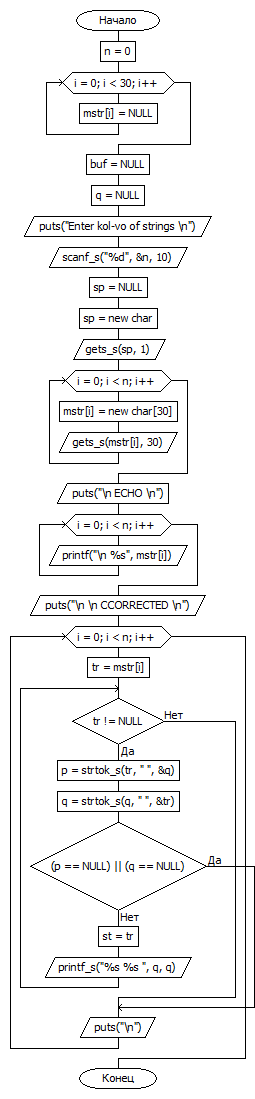
*}*

*}*

**Тесты**

|  |  |  |
| --- | --- | --- |
| Входные данные | Ожидаемые выходные данные | Выходные данные |
| 1  удалить оставить | ECHO  удалить оставить  CCORRECTED  оставить оставить | ECHO  удалить оставить  CCORRECTED  оставить оставить |
| 3  удалить оставить удалить2  1 2 3 4  s1 s2 s3 s4 s5 | ECHO  удалить оставить удалить2  1 2 3 4  s1 s2 s3 s4 s5  CCORRECTED  оставить оставить  2 2 4 4  s2 s2 s4 s4 | ECHO  удалить оставить удалить2  1 2 3 4  s1 s2 s3 s4 s5  CCORRECTED  оставить оставить  2 2 4 4  s2 s2 s4 s4 |
| 1  один еще тест программы | ECHO  один еще тест программы  CCORRECTED  еще еще программы программы | ECHO  один еще тест программы  CCORRECTED  еще еще программы программы |

**Схема алгоритма**

****

**Задание 3**

Организовать программным способом символьные файлы F и G. Определить совпадают ли компоненты этих файлов. Если нет, то получить номер первой компоненты (символа), с которой начинаются различия. В случае, когда один из файлов имеет N компонент (N>=0) и повторяет начало другого (более длинного) файла, ответом должно быть число N+1. Вывести на экран исходные файлы и результат сравнения.

**Исходный код**

*#include <iostream>*

*using namespace std;*

*int main()*

*{*

*FILE\* f, \* g;*

*char\* s1;*

*fopen\_s(&f, "f.dat", "w+t");*

*fopen\_s(&g, "g.dat", "w+t");*

*s1 = new char[50];*

*\*s1 = ' ';*

*puts("Enter string 1");*

*s1 = gets\_s(s1, 50);*

*for (int i = 0; i < strlen(s1); i++) {*

*putc(s1[i], f);*

*}*

*puts("Enter string 2");*

*s1 = gets\_s(s1, 50);*

*for (int i = 0; i < strlen(s1); i++) {*

*putc(s1[i], g);*

*}*

*rewind(f);*

*rewind(g);*

*char ch1, ch2;*

*int n = -1;*

*int k = 1;*

*bool fl = false;*

*ch1 = getc(f);*

*ch2 = getc(g);*

*while ((ch1 != EOF) && (ch2 != EOF) && (ch1 == ch2)) {*

*k++;*

*ch1 = getc(f);*

*ch2 = getc(g);*

*if (ch1 != ch2) {*

*fl = true;*

*}*

*}*

*if (ch1 != ch2) {*

*fl = true;*

*}*

*if (fl) {*

*printf("Mismatch on symb: %d", k);*

*}*

*else {*

*puts("Files matches");*

*}*

*fclose(f);*

*fclose(g);*

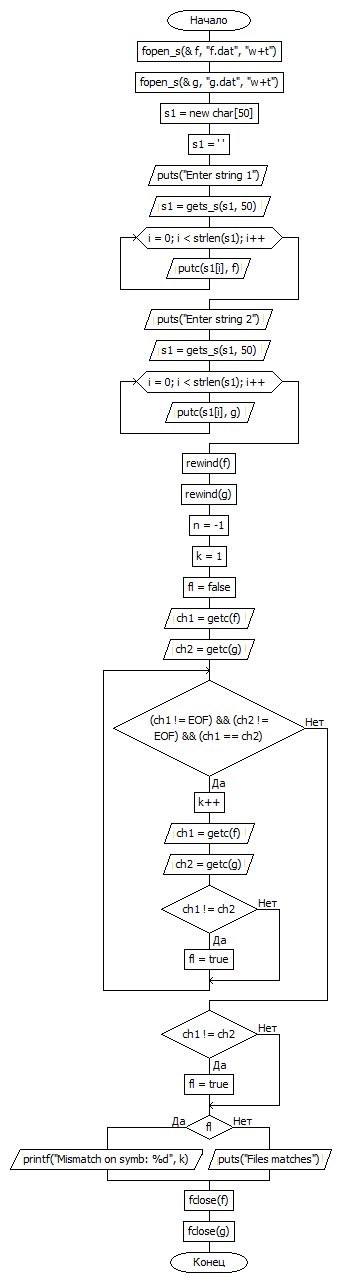
*delete[] s1;*

*}*

**Тесты**

|  |  |  |
| --- | --- | --- |
| Входные данные | Ожидаемые выходные данные | Выходные данные |
| qwerty  qwerty | Files matches | Files matches |
| qwerty  qwen | Mismatch on symb: 4 | Mismatch on symb: 4 |
| qwerty  qwe | Mismatch on symb: 4 | Mismatch on symb: 4 |

**Схема алгоритма**

****

**Вывод**

* Работа с файлами и функциями в С++ не имеет значительных отличий от Delphi. Работа со строками осуществляется за счет специальных функций. Отличительной особенность строк в С++ является то, что формально это указатели на символ.