**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ   
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«СИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ТЕЛЕКОММУНИКАЦИЙ И ИНФОРМАТИКИ»**

**ХАБАРОВСКИЙ ИНСТИТУТ ИНФОКОММУНИКАЦИЙ (ФИЛИАЛ)**

**СРЕДНЕЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАНИЕ**

ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №14

по МДК 01.01 Системное программирование

на тему

Применение рекурсивных функций.

Выполнил: А.В. Медведев

Проверил: Н.А. Насонова

Группа: ПКС-320

2023

Составление программ.

Для начала работы подключим библиотеки для работы с основными функциями. И подключим пространство имён std.

#include <iostream>

#include <fstream>

#include <cmath>

using namespace std;

iostream — заголовочный файл с классами, функциями и переменными для организации ввода-вывода.

fstream – заголовочный файл из стандартной библиотеки C++, включающий набор классов, методов и функций, которые предоставляют интерфейс для чтения/записи данных из/в файл.

cmath — заголовочный файл стандартной библиотеки разработанный для выполнения простых математических операций.

iomanip - заголовочный файл стандартной библиотеки, в котором описаны манипуляторы при работе с потоковыми операциями.

В методе main опишем использования русского алфавита.

setlocale(LC\_ALL, "Russian");

Для удобства проверки, задания выполненны в одном файле .cpp.

1. Ввести последовательность чисел (окончание ввода – 0) и вывести их в обратной последовательности. Входные данные взять из текстового файла.
2. Используя cout <<x лишь при x=0..9, написать рекурсивную программу печати десятичной записи целого положительного числа n.
3. Напишите рекурсивную функцию, которая возвращает среднее из n элементов массива чисел.
4. Найти первые N чисел Фибоначчи двумя способами: с помощью рекурсии и с помощью итерации. Сравнить эффективность алгоритмов.
5. Написать функцию сложения двух чисел, используя только прибавление единицы.
6. Написать функцию умножения двух чисел, используя только операцию сложения.
7. Вычислить сумму элементов одномерного массива.
8. Найти НОД (наибольший общий делитель) двух натуральных чисел.
9. Проверить, является ли фрагмент строки с *i*-го по *j*-й символ палиндромом.
10. Вычислить произведение элементов одномерного массива.
11. Написать функцию сортировки массива методом простого выбора.
12. Подсчитать количество цифр в заданном числе.
13. Написать функцию, проверяющую правильность имени в языке Pascal.
14. Написать функцию определения, является ли заданное натуральное число простым.

#include <iostream>

#include <fstream>

#include <vector>

#include <math.h>

#include <cmath>

using namespace std;

// Задание 1 : Вывести в обратном порядке из файла

void reverseOrder(ifstream& infile) {

int num;

infile >> num;

if (num != 0) {

reverseOrder(infile);

cout << num << " ";

}

}

// Задание 2 : Числовое представление

void decimalRepresentation(int n) {

if (n < 10) {

cout << n;

}

else {

decimalRepresentation(n / 10);

cout << n % 10;

}

}

// Задание 3 : Среднее по массиву

double averageArray(int arr[], int n) {

if (n == 1) {

return arr[0];

}

else {

return ((arr[n - 1] + averageArray(arr, n - 1) \* (n - 1)) / n);

}

}

// Задание 4 : Числа Фиббоначи

void fibRecursive(int n, int& prev, int& curr) {

if (n == 0) {

curr = 0;

prev = 1;

}

else {

int tempPrev, tempCurr;

fibRecursive(n - 1, tempPrev, tempCurr);

curr = tempPrev + tempCurr;

prev = tempCurr;

}

}

void fibIterative(int n) {

int prev = 0, curr = 1;

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

int temp = prev;

prev = curr;

curr = temp + curr;

}

cout << prev << " ";

}

// Задание 5 : Сложить два числа

int add(int a, int b) {

if (b == 0) {

return a;

}

else {

return add(a ^ b, (a & b) << 1);

}

}

// Задание 6 : Умножить два числа

int multiply(int a, int b) {

if (b == 0) {

return 0;

}

else if (b % 2 == 0) {

return multiply(a + a, b / 2);

}

else {

return add(a, multiply(a + a, (b - 1) / 2));

}

}

// Задание 7 : Сумма элементов массива

int sumArray(int arr[], int n) {

if (n == 0) {

return 0;

}

else {

return arr[n - 1] + sumArray(arr, n - 1);

}

}

// Задание 8 : НОД

int gcd(int a, int b) {

if (b == 0) {

return a;

}

else {

return gcd(b, a % b);

}

}

// Задание 12 : Палиндром

bool isPalindrome(string str, int i, int j) {

if (i >= j) {

return true;

}

if (str[i] != str[j]) {

return false;

}

return isPalindrome(str, i + 1, j - 1);

}

// Задание 13 : Произведение элементов

int arrayProduct(int arr[], int n) {

if (n == 0) {

return 1;

}

return arr[n - 1] \* arrayProduct(arr, n - 1);

}

// Задание 14 : Сортировка массива

void selectionSort(int arr[], int n) {

if (n <= 1) {

return;

}

int maxIdx = 0;

for (int i = 1; i < n; i++) {

if (arr[i] > arr[maxIdx]) {

maxIdx = i;

}

}

swap(arr[maxIdx], arr[n - 1]);

selectionSort(arr, n - 1);

}

// Задание 15 : Количество цифр в заданном числе

int countDigits(int n) {

if (n == 0) {

return 0;

}

return 1 + countDigits(n / 10);

}

// Задание 16 : Правильность имён на Паскале

bool isValidPascalName(string name) {

if (name.length() == 0) {

return false;

}

if (!isupper(name[0])) {

return false;

}

for (int i = 1; i < name.length(); i++) {

if (!islower(name[i]) && !isdigit(name[i])) {

return false;

}

}

return true;

}

// Задание 18 : Простое число

bool isPrime(int n, int i = 2) {

if (n <= 2) {

return (n == 2) ? true : false;

}

if (n % i == 0) {

return false;

}

if (i \* i > n) {

return true;

}

return isPrime(n, i + 1);

}

// Задание 19 : Цепочка построения домино

void dominoChain(int dominoes[], int n, int last, bool used[], string& chain) {

if (chain.length() == n \* 2) {

cout << chain << endl;

return;

}

for (int i = 0; i < n \* 2; i++) {

if (!used[i]) {

int a = dominoes[i / 2];

int b = dominoes[i / 2 + (i % 2 ? -1 : 1)];

if (a == last) {

used[i] = true;

chain.push\_back('0' + a);

chain.push\_back('0' + b);

dominoChain(dominoes, n, b, used, chain);

chain.pop\_back();

chain.pop\_back();

used[i] = false;

}

else if (b == last) {

used[i] = true;

chain.push\_back('0' + b);

chain.push\_back('0' + a);

dominoChain(dominoes, n, a, used, chain);

chain.pop\_back();

chain.pop\_back();

used[i] = false;

}

}

}

}

int main() {

setlocale(LC\_ALL, "RU");

ifstream infile("input.txt");

// Задание 1 : Вывести в обратном порядке из файла

cout << "Задание 1 : Вывести в обратном порядке из файла" << endl;

reverseOrder(infile);

cout << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

// Задание 2 : Числовое представление

cout << "Задание 2 : Числовое представление" << endl;

int n;

cout << "Введите положительное число: ";

cin >> n;

cout << "Десятичная запись: ";

decimalRepresentation(n);

cout << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

// Задание 3 : Среднее по массиву

cout << "Задание 3 : Среднее по массиву" << endl;

int Array[] = { 1, 2, 3, 4, 5 };

int n2 = sizeof(Array) / sizeof(Array[0]);

cout << "Среднее по массиву : " << averageArray(Array, n2) << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

// Задание 4 : Числа Фиббоначи

cout << "Задание 4 : Числа Фиббоначи" << endl;

cout << "Используя рекурсию: ";

int prev = 0, curr = 0;

for (int i = 0; i < 10; i++) {

fibRecursive(i, prev, curr);

cout << curr << " ";

}

cout << endl;

cout << "Используя итерации: ";

fibIterative(10);

cout << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

// Задание 5 : Сложить два числа

cout << "Задание 5 : Сложить два числа" << endl;

int a, b;

cout << "Введите два числа для сложения: ";

cin >> a >> b;

cout << "Сумма: " << add(a, b) << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

// Задание 6 : Умножить два числа

cout << "Задание 6 : Умножить два числа" << endl;

cout << "Введите два числа для умножения: ";

cin >> a >> b;

cout << "Произведение: " << multiply(a, b) << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

// Задание 7 : Сумма элементов массива

cout << "Задание 7 : Сумма элементов массива" << endl;

int MyArray[] = { 10, 20, 30, 40, 50 };

int n3 = sizeof(MyArray) / sizeof(MyArray[0]);

cout << "Сумма элементов: " << sumArray(MyArray, n3) << endl;

// Задание 8 : НОД

cout << "Задание 8 : НОД" << endl;

int a2, b2;

cout << "Введите два положительных числа: ";

cin >> a2 >> b2;

cout << "НОД: " << gcd(a2, b2) << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

// Задание 12 : Палиндром

cout << "Задание 12 : Палиндром" << endl;

string str = "racecar";

int i = 0;

int j = str.length() - 1;

if (isPalindrome(str, i, j)) {

cout << str.substr(i, j - i + 1) << " палиндром " << endl;

}

else {

cout << str.substr(i, j - i + 1) << " не палиндром " << endl;

}

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

// Задание 13 : Произведение элементов массива

cout << "Задание 13 : Произведение элементов массива" << endl;

int ByMyG[] = { 1, 2, 3, 4, 5 };

n = sizeof(ByMyG) / sizeof(ByMyG[0]);

int product = arrayProduct(ByMyG, n);

cout << "Произведение элементов массива: " << product << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

// Задание 14 : Сортировка массива

cout << "Задание 14 : Сортировка массива" << endl;

int LoP[] = { 5, 3, 8, 6, 7, 2 };

int len = sizeof(LoP) / sizeof(LoP[0]);

selectionSort(LoP, len);

cout << "Отсортированный массив ";

for (int i = 0; i < n2; i++) {

cout << LoP[i] << " ";

}

cout << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

// Задание 15 : Количество цифр в заданном числе

cout << "Задание 14 : Количество цифр в заданном числе" << endl;

int num = 12345;

int numDigits = countDigits(num);

cout << "Количество цифр числа " << num << " равно " << numDigits << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

// Задание 16 : Правильность имён на Паскале

cout << "Задание 16 : Правильность имён на Паскале" << endl;

string name = "ValidName1";

if (isValidPascalName(name)) {

cout << name << " возможное имя на языке Паскаль" << endl;

}

else {

cout << name << " так писать на Паскале нельзя! " << endl;

}

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

// Задание 18 : Простое число

cout << "Задание 18 : Простое число" << endl;

int num2 = 17;

if (isPrime(num2)) {

cout << num2 << " простое число " << endl;

}

else {

cout << num2 << " непростое число " << endl;

}

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

return 0;

}

