|  |
| --- |
|  |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** |
| Институт искусственного интеллекта (ИИИ)  Кафедра промышленной информатики (ПИ) |

**ОТЧЁТ ПО**

**ПРАКТИЧЕСКОЙ РАБОТЕ №4**

по дисциплине «Информатика»

|  |  |  |  |
| --- | --- | --- | --- |
| Студент группы | Давлетов Э.А., ИВБО–21–24 |  |  |
|  | *(Ф.И.О., учебная группа)* |  | *(подпись студента)* |
| Руководитель лаб. работы |  |  |  |
|  | *(Ф.И.О.)* |  | *(подпись руководителя)* |
| Лабораторная работа представлена к защите | «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2024г. |  |  |
|  |  |  |  |

Москва, 2024г.

Задача 1

#include <iostream>

#include <fstream>

using namespace std;

int main() {

ofstream zap("D:\\somedocuments\\informatika\\test5.txt");

if (!zap.is\_open()) {

cout << "File wasn't open. Try again" << endl;

}

else {

char nums[64];

cin.getline(nums, 63);

zap << nums;

zap.close();

}

//

//

ifstream test5("D:\\somedocuments\\informatika\\test5.txt");

float buff2 = 0;

float counter = 0;

if (!test5.is\_open()) {

cout << "File wasn't open. Try again" << endl;

}

else {

while (!test5.eof()) {

test5 >> buff2;

counter += buff2;

}

cout << counter << endl;

test5.close();

}

}

Задача 2

#include <iostream>

using namespace std;

int sign(float x) {

if (x > 0) {

return 1;

}

else if (x < 0) {

return -1;

}

else {

return 0;

}

}

int main() {

float number;

cout << "Enter number: ";

cin >> number;

int result = sign(number);

cout << "Sign num: " << result << endl;

return 0;

}

Задача 3

#include <iostream>

#include <cmath>

using namespace std;

double a, b, c, r;

double rectangle(double a, double b) {

if ((a <= 0) or (b <= 0)) {

return -1;

}

return a \* b;

}

double triangle(double a, double b, double c) {

double ab = a + b;

double bc = b + c;

double ac = a + c;

if ((a <= 0) or (b <= 0) or (c <= 0) or (ab <= c) or (bc <= a) or (ac <= b)) {

return -1;

}

double p = (a + b + c) / 2;

return sqrt(p \* (p - a) \* (p - b) \* (p - c));

}

double circle(double r) {

if (r <= 0) {

return -1;

}

double pi = 3.14159;

return pi \* pow(r, 2);

}

int main() {

int selector;

cout << "1 - rectangle, 2 - triangle, 3 - circle: ";

cin >> selector;

switch (selector)

{

case 1:

cout << "a = ";

cin >> a;

cout << "b = ";

cin >> b;

if (rectangle(a, b) == -1) {

cout << "Wrong data";

}

else {

cout << "S of rectangle = " << rectangle(a, b) << endl;

}

break;

case 2:

cout << "a = ";

cin >> a;

cout << "b = ";

cin >> b;

cout << "c = ";

cin >> c;

if (triangle(a, b, c) == -1) {

cout << "Wrong data";

}

else {

cout << "S of triangle = " << triangle(a, b, c) << endl;

}

break;

case 3:

cout << "r = ";

cin >> r;

if (circle(r) == -1) {

cout << "Wrong data";

}

else {

cout << "S of circle = " << circle(r) << endl;

}

break;

}

return 0;

}

Задача 4

#include <iostream>

#include <cstdlib>

using namespace std;

int main() {

const int rows = 13;

const int columns = 35;

char flag[rows][columns]{};

for (int x = 0; x < rows; x++) {

for (int y = 0; y < columns; y++) {

if ((x < 6) and (y < 8)) {

flag[x][y] = '\*';

}

else {

flag[x][y] = '\_';

}

}

}

for (int x = 0; x < rows; x++) {

for (int y = 0; y < columns; y++) {

cout << flag[x][y];

}

cout << endl;

}

return 0;

}

Задача 5

#include <cmath>

#include <iostream>

#include <string>

#include <vector>

using namespace std;

int main()

{

int size = 100; // ширина графика

int height = 21; // высота от 21 иначе будет ошибка

vector<string> sinGraph(height, string(size, ' ')); // пустое поле

for (int i = 0; i < size; i++) { // точки графика

sinGraph[(round(5 \* sin(i / 9) + 10))][i] = '\*'; // от 1 до 10 меняется амплитуда

}

for (auto& s : sinGraph) { // принт графика

cout << s << '\n';

}

}

Задача 6

#include <iostream>

#include <cstdlib>

using namespace std;

int RomanLetters(char rome) {

switch (rome)

{

case 'I': return 1;

case 'V': return 5;

case 'X': return 10;

case 'L': return 50;

case 'C': return 100;

case 'D': return 500;

case 'M': return 1000;

default: return 0;

}

}

int RomanDecode(string rome) {

int res = 0;

int previousVal = 0;

for (int i = rome.length() - 1; i >= 0; i--) {

char currentSym = rome[i];

int currentVal = RomanLetters(currentSym);

if (currentVal < previousVal) {

res -= currentVal;

}

else {

res += currentVal;

}

previousVal = currentVal;

}

return res;

}

int main() {

string romanNumber;

int defaultNumber;

bool stopper = true;

while (stopper == true) {

cout << "roman number: ";

cin >> romanNumber;

defaultNumber = RomanDecode(romanNumber);

cout << "10cc num: " << defaultNumber << endl;

cout << "Enter 0 to stop, 1 to continue: ";

cin >> stopper;

}

return 0;

}

Задача 7

#include <iostream>

using namespace std;

int rng(int s, int m, int i, int c) {

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

s = (s \* m + i) % c;

cout << s << endl;

}

return s;

}

int main() {

int m0, i0, c0, m1, c1, i1;

int s = 0;

int cases;

cout << "Choose small(0) or big(1) numbers (0 or 1): ";

cin >> cases;

switch (cases)

{

case 0:

m0 = 37;

i0 = 3;

c0 = 64;

cout << rng(s, m0, i0, c0) << endl;

break;

case 1:

m1 = 25173;

i1 = 13849;

c1 = 65537;

cout << rng(s, m1, i1, c1) << endl;

break;

}

return 0;

}

Задача 8

#include <iostream>

using namespace std;

int main()

{

double matrix\_a[3][4] = { // 3x4

{5, 2, 0, 10},

{3, 5, 2, 5},

{20, 0, 0, 0} };

double matrix\_b[4][2] = { // 4x2

{1.20, 0.50},

{2.80, 0.40},

{5.00, 1.00},

{2.00, 1.50} };

double matrix\_c[3][2] = { // 3x2

{0,0},

{0,0},

{0,0} };

for (int x = 0; x < 3; x++) { // 3 строки

for (int y = 0; y < 2; y++) { // 2 столбца

for (int i = 0; i < 4; i++) {

matrix\_c[x][y] += matrix\_a[x][i] \* matrix\_b[i][y];

}

cout << matrix\_c[x][y] << " ";

}

cout << endl;

}

cout << "=============" << endl;

// 1) какой продавец выручил больше всего денег с продажи, какой – меньше;

int maxSeller = 0;

int minSeller = 0;

double maxMoney = 0;

double minMoney = 10000000;

for (int x = 0; x < 3; x++) { // прохожу по строкам матрицы 0 (1) столбца

if (matrix\_c[x][0] > maxMoney) {

maxSeller = x + 1;

maxMoney = matrix\_c[x][0];

}

if (matrix\_c[x][0] < minMoney) {

minSeller = x + 1;

minMoney = matrix\_c[x][0];

}

}

cout << "Seller " << maxSeller << " MaxMoney: " << maxMoney << endl;

cout << "Seller " << minSeller << " MinMoney: " << minMoney << endl;

cout << "=============" << endl;

// 2) какой получил наибольшие комиссионные, какой – наименьшие

maxSeller = 0;

minSeller = 0;

double maxCom = 0;

double minCom = 10000000;

for (int x = 0; x < 3; x++) { // прохожу по строкам матрицы 1 (2) столбца

if (matrix\_c[x][1] > maxCom) {

maxSeller = x + 1;

maxCom = matrix\_c[x][1];

}

if (matrix\_c[x][1] < minCom) {

minSeller = x + 1;

minCom = matrix\_c[x][1];

}

}

cout << "Seller " << maxSeller << " MaxCom " << maxCom << endl;

cout << "Seller " << minSeller << " MinCom " << minCom << endl;

cout << "=============" << endl;

// 3) чему равна общая сумма денег, вырученных за проданные товары

double sumMoney = 0;

for (int x = 0; x < 3; x++) {

sumMoney += matrix\_c[x][0];

}

cout << "SumMoney = " << sumMoney << endl;

// 4) сколько всего комиссионных получили продавцы

double sumCom = 0;

for (int x = 0; x < 3; x++) {

sumCom += matrix\_c[x][1];

}

cout << "SumCom = " << sumCom << endl;

// 5) чему равна общая сумма денег, прошедших через руки продавцов

double result = 0;

for (int x = 0; x < 3; x++) {

result += matrix\_c[x][0] + matrix\_c[x][1];

}

cout << "Result summ = " << result << endl;

return 0;

}

Задача 9

#include <iostream>

#include <string>

#include <algorithm>

#include <cmath>

using namespace std;

int convertToDestich(const string& num, int base) { // в десятич из исходной

int result = 0;

int power = 1; // Степень основания

for (int i = num.length() - 1; i >= 0; i--) { // с конца идем

char digit = num[i];

int digitValue;

if (digit >= '0' && digit <= '9') {

digitValue = digit - '0'; // от 0 до 9

}

else if (digit >= 'A' && digit <= 'Z') {

digitValue = digit - 'A' + 10; // от A - Z для основания больше 10

}

else if (digit >= 'a' && digit <= 'z') {

digitValue = digit - 'a' + 10; // от a - z маленьких

}

else {

throw invalid\_argument("Invalid character in the number.");

}

result += digitValue \* power;

power \*= base; // Увеличиваем степень основания

}

return result;

}

bool isValidNumber(const string& num, int base) {

for (char digit : num) {

int digitValue;

if (digit >= '0' and digit <= '9') {

digitValue = digit - '0'; // от 0 до 9

}

else if (digit >= 'A' and digit <= 'Z') {

digitValue = digit - 'A' + 10; // от A - Z

}

else if (digit >= 'a' and digit <= 'z') {

digitValue = digit - 'a' + 10; // от a - z

}

else {

return false; // такого нет символа

}

// проверка что цифра не больше основания

if (digitValue >= base) {

return false; // фолс!!

}

}

return true; // норм всё1

}

string convertFromDestich(int num, int base) { // из десятич в новое

string result = "";

while (num > 0) { // пока больше нуля

int remainder = num % base;

if (remainder < 10) {

result += char(remainder + '0'); // 0 - 9

}

else {

result += char(remainder - 10 + 'A'); // A - Z

}

num /= base; // тут число мы срезаем

}

if (result.empty()) {

return "0"; // если нолик

}

reverse(result.begin(), result.end()); // ревёрс строки

return result;

}

int main() {

setlocale(LC\_ALL, "Russian");

string inputNumber;

int oldBase, newBase;

cout << "Введите число: ";

cin >> inputNumber;

cout << "Введите старое основание системы счисления: ";

cin >> oldBase;

cout << "Введите новое основание системы счисления: ";

cin >> newBase;

if (!isValidNumber(inputNumber, oldBase)) {

cout << "Ошибка: введённое число содержит недопустимые цифры для системы счисления с основанием " << oldBase << endl;

return 1; // проверка

}

try { // через трай кетчи

// в десятичную систему

int decimalValue = convertToDestich(inputNumber, oldBase);

// из десятичной системы в новое основание

string result = convertFromDestich(decimalValue, newBase);

// тут результаты

cout << "Число " << inputNumber << " в системе счисления с основанием " << oldBase

<< " равно " << result << " в системе счисления с основанием " << newBase << endl;

}

catch (const exception& e) {

cout << "Ошибка: " << e.what() << endl;

}

return 0;

}