

Лабораторная работа № 1

Отладка простейших программ: определение площадей и объемов геометрических фигур, ввод данных с клавиатуры, программ по использованию встроенных функций вывод на экран сообщений и т.д.

Гвоздиков Данила Алексеевич
М33-105Бк-20

Анализ требований

Программа должна выполнять простейшие вычисления по переданным на вход данным. Во всех программа вводимые данные передаются в терминал при помощи клавиатуры, после чего программа должна вывести результат.

Во всех программах, где результат представляет вещественное число, результат округляется до 4-го знака после запятой.

- 1) По введенному значению радиуса определять длину и площадь окружности.

Радиус - вещественное число (число с плавающей точкой),

Длина и площадь окружности - вещественные числа.

Длина вычисляется по формуле $L = 2 * \Pi * R$, где L - длина строки, R - радиус окружности

Площадь окружности вычисляется по формуле $S = \Pi * R^2$, где S - площадь окружности, R - радиус окружности.

Ограничения: введенные данные должны быть больше 0.

- 2) Определить объем куба, цилиндра, конуса.

Для определения каждого объема, необходимо получить недостающие элементы уравнения:

- Объем куба $V = a^3$, нужно ввести длину стороны куба - целое число.
- Объем цилиндра $V = h * \Pi R^2$, нужно ввести радиус основания и высоту цилиндра - два целых числа.
- Объем конуса $V = 1/3 * \Pi R^2 h$, нужно ввести радиус основания и высоту конуса - два целых числа.

Ограничения: введенные данные должны быть больше 0.

- 3) Определить принадлежит ли точка с координатами X и Y прямоугольнику с координатами X1, X2, Y1, Y2. На вход должны подаваться координаты прямоугольника и координаты точки - целые числа. Условия по которым проверяется принадлежность: $X \geq X_1, X \leq X_2, Y \geq Y_1, Y \leq Y_2$. На вывод должен передаваться текстовый ответ «принадлежит» или «не принадлежит»
Ограничения: введенные данные должны быть больше 0; $X_1 < X_2; Y_1 > Y_2$;

- 4) Определить площадь поверхности дома. Высота дома - A, ширина - D, окно - B*C. На каждой стене по два окна. Судя по описанию данным - дом представляет из себя квадратное строение. Следовательно, площадь одной стены дома будет считаться по формуле $S_w = A * D - 2(B * C)$. Площадь фундамента и крыши по формуле: $S_r = D^2$. Площадь всего дома считается по формуле $S = 4S_w + 2S_r = 4AD - 8BC + 2D^2$.
На вход передаются целые числа: A, D, B, C. Вывод - вещественное число - площадь поверхности.

- 5) Определить сумму трех чисел. На вход подается три целых числа, на выходе выводится их сумма.

- 6) Найти минимум из трех чисел. На вход передается три целых числа, на выходе выводится минимальное из всех троих чисел.

Определение спецификаций

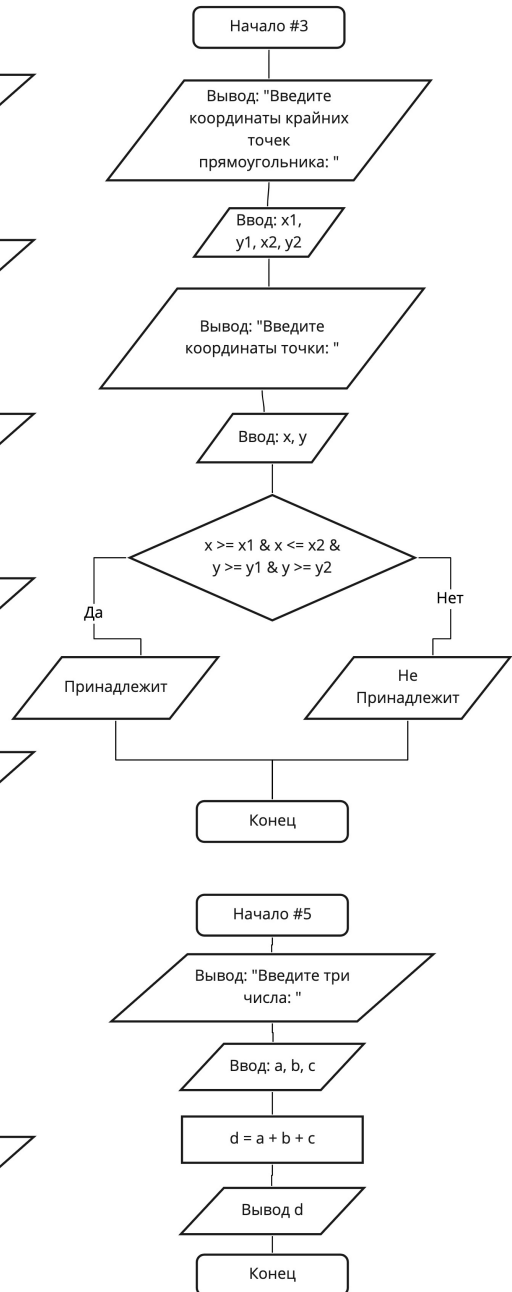
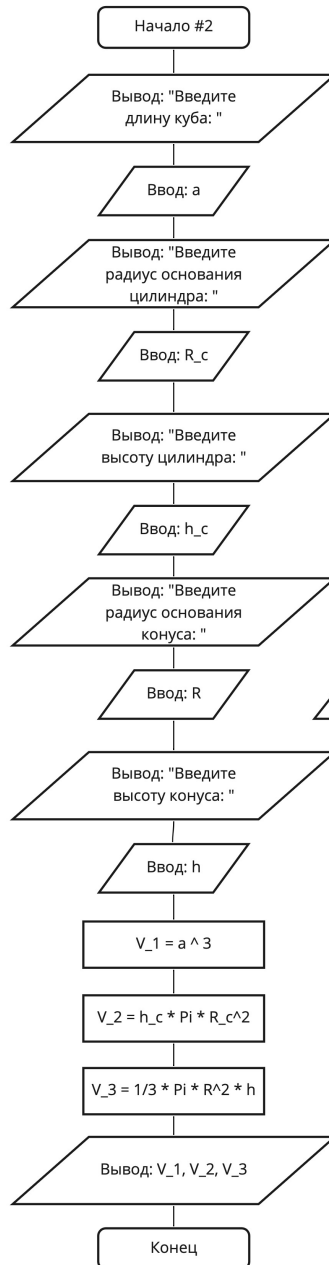
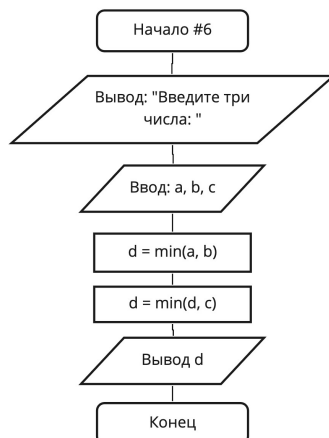
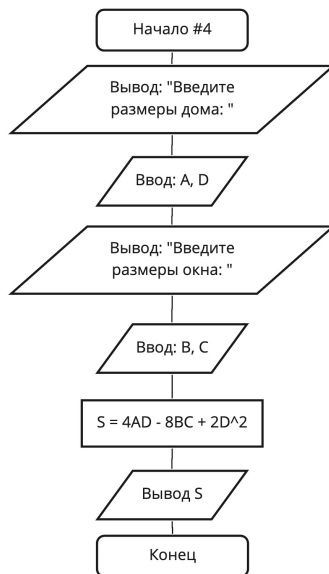
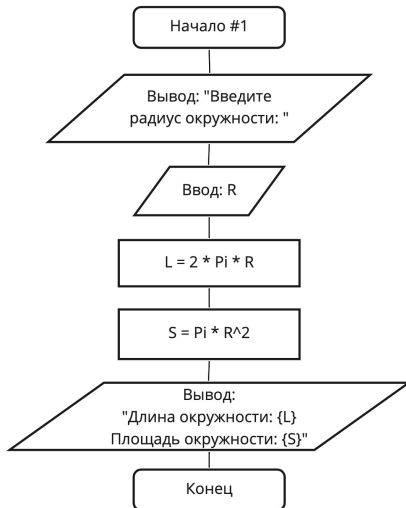
Необходима проверка входных данных: порядок ввода должен быть строго определен для каждой задачи.

Ограничения памяти - не более 1 килобайта для каждой программы.

Ограничение по времени выполнения подсчетов - не больше 1 секунды.

Ограничение по сложности - без ограничений

Проектирование



Исходный текст программы

```
//  
// Created by Danila Gvozdikov on 24.12.2020.  
//  
#include <iostream>  
#include <cmath>  
#include <stdexcept>  
#include <string>  
  
using namespace std;  
  
void throwError(const string& message) {  
    throw runtime_error(message);  
}  
  
void throwLeqZero(const int value, const string& valueName) {  
    if (value <= 0) {  
        throwError(valueName + " должна(ен) быть больше 0.");  
    }  
}  
  
int inputInt(const string& desc = "") {  
    string input;  
    cin >> input;  
  
    int result = 0;  
    try {  
        result = stoi(input);  
    } catch (exception err) {  
        throw runtime_error(desc);  
    }  
    return result;  
}  
  
// Base class for program  
// All programs below inherits base program but should provide:  
// - variables to input and variables for results  
// - method InputValues() to input required data  
// - method ValidateValues() to validate input that throws runtime_error when data is  
// incorrect.  
// - method ProcessValues() to count result  
// - const method OutputResult() to print result  
class TProgram {  
public:  
    void run() {  
        while (true) {
```

```

        try {
            this->InputValues(); // input required variables
        } catch (runtime_error) {
            cout << "Некорректный тип введенных данных, ожидалось целое число." <<
endl
                << "Попробуйте еще раз." << endl;
            continue;
        }

        try {
            this->ValidateData();
        } catch (runtime_error err) {
            cout << "Введенные данные некорректны, ошибка: \"" << err.what() << "\"\"
<< endl
                << "Попробуйте ввести новые данные." << endl;
            continue;
        }

        break;
    }

    this->ProcessValues(); // count result from input
    cout << endl; // just one empty string between input and output

    // Print fixed float values with 4 numbers after the comma.
    cout << fixed;
    cout.precision(4);
    this->OutputResult(); // print results
};

protected:
    virtual void InputValues() {};
    virtual void ValidateData() const {};
    virtual void ProcessValues() {};
    virtual void OutputResult() const {};
};

class TProgram1: public TProgram {
protected:
    int radius = 0;
    double length = 0.0;
    double square = 0.0;

    void InputValues() {
        cout << "Введите радиус окружности: ";
        radius = inputInt();
    }
}

```

```

void ValidateData() const {
    throwLeqZero(radius, "Радиус");
}
void ProcessValues() {
    auto radius = double(this->radius);

    length = radius * M_PI * 2;
    square = radius * radius * M_PI;
}
void OutputResult() const {
    cout << fixed;
    cout.precision(4);
    cout << "Длина окружности: " << length << endl
        << "Площадь окружности: " << square << endl;
}
};

```

```

class TProgram2: public TProgram {
protected:
    int cubeSide = 0;
    int cubeVolume = 0;

    int cylinderBaseRadius = 0;
    int cylinderHeight = 0;
    double cylinderVolume = 0;

    int coneBaseRadius = 0;
    int coneHeight = 0;
    double coneVolume = 0;

    void InputValues() {
        cout << "Сторона куба: ";
        cubeSide = inputInt();
        cout << "Радиус основания цилиндра: ";
        cylinderBaseRadius = inputInt();
        cout << "Высота цилиндра: ";
        cylinderHeight = inputInt();
        cout << "Радиус основания конуса: ";
        coneBaseRadius = inputInt();
        cout << "Высота конуса: ";
        coneHeight = inputInt();
    }

    void ValidateData() const {
        throwLeqZero(cubeSide, "Длинна стороны куба");
        throwLeqZero(cylinderBaseRadius, "Радиус основания цилиндра");
        throwLeqZero(cylinderHeight, "Высота цилиндра");
    }
};

```

```

        throwLeqZero(coneBaseRadius, "Радиус основания конуса");
        throwLeqZero(coneHeight, "Высота конуса");
    }

    void ProcessValues() {
        cubeVolume = pow(double(cubeSide), 3);
        cylinderVolume = M_PI * pow(double(cylinderBaseRadius), 2) * cylinderHeight;
        coneVolume = M_PI * pow(double(coneBaseRadius), 2) * coneHeight / 3;
    }

    void OutputResult() const {
        cout << "Объем куба: " << cubeVolume << endl
              << "Объем цилиндра: " << cylinderVolume << endl
              << "Объем конуса: " << coneVolume << endl;
    }
};

struct TPoint {
    int x = 0, y = 0;
};

class TProgram3: public TProgram {
protected:
    TPoint one = {};
    TPoint two = {};
    TPoint dot = {};

    bool result = false;

    void InputValues() {
        cout << "Введите координаты верхней левой точки прямоугольника (в формате x y): ";

        one.x = inputInt();
        one.y = inputInt();

        cout << "Введите координаты нижней правой точки прямоугольника (в формате x y): ";

        two.x = inputInt();
        two.y = inputInt();

        cout << "Введите координаты произвольной точки (в формате x y): ";
        dot.x = inputInt();
        dot.y = inputInt();
    }

    void ValidateData() const {
        if (one.x >= two.x) {

```



```

        throwErr("Координата x левой точки прямоугольника должна быть меньше x правой
точки.");
    }
    if (two.y >= one.y) {
        throwErr("Координата y верхней точки должна быть больше y нижней точки.");
    }
};

void ProcessValues() {
    result = dot.x >= one.x and dot.x <= two.x and dot.y <= one.y and dot.y >= two.y;
}

void OutputResult() const {
    if (result) {
        cout << "Точка принадлежит прямоугольнику.";
    } else {
        cout << "Точка не принадлежит прямоугольнику.";
    }
}
};

```

```

class TProgram4: public TProgram {
protected:

```

```

    int height = 0;
    int width = 0;
    int windowHeight = 0;
    int windowWidth = 0;

```

```

    int result = 0;

```

```

void InputValues() {
    cout << "Высота стены дома: ";
    height = inputInt();
    cout << "Ширина стены дома: ";
    width = inputInt();

```

```

    cout << "Высота окна: ";
    windowHeight = inputInt();
    cout << "Ширина окна: ";
    windowWidth = inputInt();
}

```

```

void ValidateData() const {
    throwLeqZero(height, "Высота стены дома");
    throwLeqZero(width, "Ширина стены дома");
    throwLeqZero(windowHeight, "Высота окна");
    throwLeqZero(windowWidth, "Ширина окна");

```

```

    if (windowHeight >= height) {
        throwErr("Высота окна не может превышать или быть равной высоте дома.");
    }
}

```

```

    }
    if (windowWidth * 2 >= width) {
        throwErr("Ширина окна не может превышать или быть равной половине ширины
дома.");
    }
}
void ProcessValues() {
    result = 4 * width * height - 8 * windowWidth * windowHeight + 2 * width * width;
}
void OutputResult() const {
    cout << "Площадь поверхности дома: " << result;
}
};

```

```

class TProgram5: public TProgram {
protected:
    int a = 0;
    int b = 0;
    int c = 0;
    int result = 0;

    void InputValues() {
        cout << "Введите три числа через пробел: ";
        a = inputInt();
        b = inputInt();
        c = inputInt();
    }
    void ProcessValues() {
        result = a + b + c;
    }
    void OutputResult() const {
        cout << "Сумма чисел: " << result;
    }
};

```

```

class TProgram6: public TProgram5 {
protected:
    void ProcessValues() {
        result = min(a, b);
        result = min(result, c);
    }
    void OutputResult() const override {
        cout << "Минимальное число: " << result;
    }
};

```

```

TProgram* ChooseProgram(int num) {
    switch (num) {
        case 1: return new TProgram1();
        case 2: return new TProgram2();
        case 3: return new TProgram3();
        case 4: return new TProgram4();
        case 5: return new TProgram5();
        case 6: return new TProgram6();
        default: return nullptr;
    }
}

int main() {

    // Для начала выбираем программу, которую мы хотим запустить
    cout << "Выбери программу:" << endl
        << "1 - Длина и площадь окружности." << endl
        << "2 - Объем куба, цилиндра, конуса." << endl
        << "3 - Принадлежность точки прямоугольнику." << endl
        << "4 - Площадь поверхности дома." << endl
        << "5 - Сумма трех чисел." << endl
        << "6 - Минимальное из трех чисел." << endl;

    //
    int progNum = 0;
    while (true) {
        cout << "Ввод: ";

        try {
            progNum = inputInt("Номер программы");
        } catch (...) {
            progNum = 0;
        }

        if (1 <= progNum && progNum <= 6) {
            break;
        }

        cout << "Ввод должен содержать целое число от 1 до 6. Попробуйте еще раз." <<
endl;
    }

    cout << endl;
    TProgram* program = ChooseProgram(progNum);

    program->run();
}

```

```

delete program;
return 0;
}

```

Тестирование

Ниже приведены таблицы с результатами тестирования.

Таблица 1 - площадь и длина окружности

Введенные данные	Ожидаемый результат	Полученный результат
10	$L = 62.8319$; $R = 314.1593$	Длина окружности: 62.8319 Площадь окружности: 314.1593
-10	Ошибка	Ошибка
0	Ошибка	Ошибка
Some text	Ошибка	Ошибка

Таблица 2 - объем куба, цилиндра, конуса

Введенные данные	Ожидаемый результат	Полученный результат
10 10 10 10 10	$V1 = 1000.0$; $V2 = 3141.5927$; $V3 = 1047.1976$	Объем куба: 1000 Объем цилиндра: 3141.5927 Объем конуса: 1047.1976
-10 0 -50 12 20	Ошибка	Ошибка
0 0 0 0 0	Ошибка	Ошибка
Some text	Ошибка	Ошибка

Таблица 3 - принадлежность точки прямоугольнику

Введенные данные	Ожидаемый результат	Полученный результат
0 30 30 0 0 0	Да	Точка принадлежит прямоугольнику.
-10 0 10 -20 40 40	Нет	Точка не принадлежит прямоугольнику.
30 40 -10 -10 0 0	Ошибка	Ошибка
Some text	Ошибка	Ошибка

Таблица 4 - площадь поверхности дома

Введенные данные	Ожидаемый результат	Полученный результат
100 100 20 20	56800	Площадь поверхности дома: 56800

Введенные данные	Ожидаемый результат	Полученный результат
20 20 100 100	Ошибка	Ошибка
Some text	Ошибка	Ошибка

Таблица 5 - сумма трех чисел

Введенные данные	Ожидаемый результат	Полученный результат
1 2 3	6	Сумма чисел: 6
0 0 0	0	Сумма чисел: 0
Some text	Ошибка	Ошибка

Таблица 6 - минимальное число

Введенные данные	Ожидаемый результат	Полученный результат
1 2 3	1	Минимальное число: 1
0 0 0	0	Минимальное число: 0
-10 10 0	-10	Минимальное число: -10
Some text	Ошибка	Ошибка