Лабораторная работа № 6

**Тема**: Реализация абстрактных типов данных

**Цель работы:** приобрести практические навыки объектного анализа и проектирования, создания классов и использования препроцессорной директив, макросов и макро операторов С ++ для формирования фундаментальных умений модульного (объектного) программирования в кроссплатформенной среде Code :: Blocks.

Вариант № 4

Задача 6.1

Дано следующее сущность предметной области:

Объем конуса равный одной третьей произведения его площади на высоту

V= V- объем конуса;

h –высота;

R- радиус основания;

S- площадь

Объект (экземпляр) класса как абстракция данной сущности предметной области за предоставленным интерфейсом обеспечивает

1. Вывод информации про своего разработчика и университет в котором он учится;
2. Вывод сообщения про значения своих атрибутов;
3. Вывод сообщения про свой объем;
4. Изменения своих атрибутов, на введенные пользователем;

Задача 6.2

Преподаватель проверяет, если файл проекта (main.cpp) во время его компиляции находится не в папке (………), то выдает 100 звуковых сигналов и в текстовый файл записывает «Установленные условия к выполнению лабораторной работы НАРУШЕНЫ», иначе - записывает имя и фамилию разработчика , время и дату компиляции программы, создает объект класса задачи 6.1 и осуществляет его тестирование (UNIT TESTING)

Выполнение лабораторной работы

1. У преподавателя получил условия задачи 6.1-6.2 по варианту№4.

2. В отчете по лабораторной работе (далее - отчет) записал тему и цель работы, номер варианта, и условия задачи.

3. Осуществил анализ задачи 6.1: концептуализацию предметной области, объектный анализ, определение интерфейса сущности предметной области; полученные результаты

(Артефакты) задокументировал и включаю в отчет.

#ifndef PERSONAL\_DATAS\_H\_INCLUDED

#define PERSONAL\_DATAS\_H\_INCLUDED

using namespace std;

class conus {

public:

double rad = 1;

double plos =1;

double vys = 1;

double obm = plos\*vys;

void PrintDatas();

void PrintAtributes();

void ChangeAtributes(double rad, double vys, double plos);

void PrintObm();

double Printobm(){return rad\*plos;};};

void conus::PrintDatas()

{

cout << "Информация про разработчика : \n\n";

cout << "Студент Группы КИ-15\n"

<< "Аннаев Арслан \n"

<< "\*\*\*KNTU\*\*\*\n";}

void conus::PrintAtributes()

{

cout << "Объем конуса :" << (vys\*plos) << endl;

cout << "Радиус основания :" << rad << endl;

cout << "Высота конуса :" << vys << endl;

cout << "Площадь основания:" << plos << endl;

}

void conus::PrintObm()

{

cout << "Объем конуса :" << plos\*vys << endl;

}

void conus::ChangeAtributes(double dob\_rad, double dob\_vys, double dob\_plos)

{

cout << "Старые атрибуты :\n";

PrintAtributes();

rad = dob\_rad;

vys = dob\_vys;

plos = dob\_plos;

}

#endif // PERSONAL\_DATAS\_H\_INCLUDED

#include <iostream>

#include <clocale>

#include <Personal\_Datas.h>

using namespace std;

int main()

{

conus abstract;

setlocale(LC\_ALL, "rus");

cout << "Задача \_\_\_\n" << endl;

double dob\_obm;

double dob\_rad;

double dob\_vys;

double dob\_plos;

char vyh;

int vyb,mat=50,c;

do {

cout << "#1 Вывод информации про разработчика и университет где он учится" << endl;

cout << "#2 Вывод информации про значения атрибутов абстракции " << endl;

cout << "#3 Вывод информации про свой объем " << endl;

cout << "#4 Изменение атрибутов конуса\n\n" << endl;

cout << "Выберите значение для выполнения программы "; cin >> vyb;

switch (vyb)

{

case 1: abstract.PrintDatas(); break;

case 2: abstract.PrintAtributes(); break;

case 3: abstract.PrintObm(); break;

case 4: { cout << "Введите значение для радиуса\n";

do{

while (!(cin>>dob\_rad))

{ cin.clear();

cin.ignore(mat,'\n');

cout << "Некорректное значение\n\n";};

cout << "Введите значение для высоты\n";

while (!(cin>>dob\_vys))

{ cin.clear();

cin.ignore(mat,'\n');

cout << "Некорректное значение\n\n";};

cout << "Введите значение для площади\n";

while (!(cin>>dob\_plos))

{ cin.clear();

cin.ignore(mat,'\n');

cout << "Некорректное значение\n\n";};

if (!(dob\_rad<=0)&&(dob\_vys<=0)&&(dob\_plos<=0))

{ cout << "Значения атрибутов не должны быть отрицательными\n";

c=0;

}

else { abstract.ChangeAtributes( dob\_rad, dob\_vys, dob\_plos);

c=1; }

} while(c!=1);}

break;

default: cout << "Повторите попытку\n"<< "Некорректный выбор\n"; }

cout << "Завершить программу?\n"

<< "(Y) or (N)" << endl;

cin>> vyh;

} while (vyh!='Y');

return 0;}

#include <iostream>

#include <fstream>

#include <windows.h>

#include <cstring>

#include "Personal\_Datas.h"

using namespace std;

class Teacher{

public:

Teacher(){

TestSuite(); }

private:

void TestSuite(){

ofstream output;

output.open("Prov.txt", ios::trunc);

conus abstract;

ifstream prj;

prj.open("C:\\Users\\Арслан\\Desktop\\2 Семестр\\Модульное программирование\\Лабораторная\\Лабораторная работа 6\\task6\_2\\main.cpp", ios::in);

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

prj.close();

cout << "Ошибка файл не открывается!" << endl;

output << "Установленные требования к лабораторной работе НАРУШЕНЫ!" << endl;;

output.close();

for (int i=0; i!=100; i++){

cout << "\a";

Sleep(100);

}

}

else {

output << "Разработчик Аннаев Арслан" << endl;

output << "Время и дата компиляции: " << \_\_TIMESTAMP\_\_ << endl;

output.close();

double dob\_rad, dob\_vys, dob\_plos;

streambuf \*old = cout.rdbuf(0);

cout.rdbuf(old);

for (int i=1; i!=4; i++)

{

if (abstract.rad == dob\_rad)

cout << "TEST\_CASE\_D " << i << ": PASSED" << endl;

else

cout << "TEST\_CASE\_D " << i << ": FAILED" << endl;

dob\_plos+5;

dob\_rad+5;

dob\_vys+5;

}

for (int i=1; i!=4; i++) {

if (abstract.Printobm()==(abstract.plos\*abstract.vys))

cout << "TEST\_CASE\_V " << i << ": PASSED" << endl;

else

cout << "TEST\_CASE\_V " << i << ": FAILED" << endl;

abstract.vys+5; } }

}

protected:};

int main()

{ setlocale(LC\_ALL, "rus");

Teacher TestSuite;

return 0;

}

выводы:

При выполнении лабораторной работы приобрел практические навыки объектного анализа и проектирования, создания классов и использования препроцессорной директив, макросов и макро операторов С ++ для формирования фундаментальных умений модульного (объектного) программирования в кроссплатформенной среде Code :: Blocks.