# МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

ГЛАВНОЕ УПРАВЛЕНИЕ ПО ОБРАЗОВАНИЮ

МОГИЛЕВСКОГО ОБЛАСТНОГО ИСПОЛНИТЕЛЬНОГО КОМИТЕТА

# УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

# “МОГИЛЕВСКИЙ ГОСУДАРСТВЕННЫЙ ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ”

Дисциплина Учебная дисциплина Тестирование

программного обеспечения

УТВЕРЖДАЮ

Заместитель директора

по учебной работе

\_\_\_\_\_\_\_\_\_М.М. Федоськова

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ЛАБОРАТОРНАЯ РАБОТА №9**

ПРОВЕДЕНИЕ МОДУЛЬНОГО ТЕСТИРОВАНИЯ И АНАЛИЗ РЕЗУЛЬТАТА

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ

Разработал преподаватель

Карманов А.В.

2021

Обсуждено и одобрено

на заседании цикловой комиссии спецдисциплин

специальности «Программное обеспечение

информационных технологий»

Протокол № \_\_ от \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**1 Цель работы**

* 1. Формирование умения проводить модульное тестирование и выполнять анализ результата

**2 Методическое и материальное обеспечение**

2.1 Методические рекомендации по выполнению лабораторной работы

2.2 Персональный компьютер IBM PC

**3 Последовательность выполнения работы**

3.1 Ознакомиться с теоретическим материалом.

3.2 Выполнить индивидуальное задание.

3.3 Оформить отчет.

**4 Теоретическое обоснование**

Рассмотрим разработанный класс «Цилиндр» из лабораторной работы 5.1

using System;

namespace ConsoleApplication7

{

class Cyl

{

private double r, h;

public double H

{

set

{

if (value < 1) { h = 0; } else { h = value; }

}

get { return h; }

}

public double R

{

set

{

if (value < 1) {r = 0; } else

r = value;

}

get { return r; }

}

public double S\_ocn() => (Math.PI \* R \* R);

public double Ob()

{

return (500);

}

public double Bp() => (2 \* Math.PI \* r \* h);

public double Pp() => (Bp() + 2 \* S\_ocn());

}

class CNS:Cyl

{

public double S\_ocn() => (Math.PI \* R \* R);

public double Ob() => (S\_ocn() \* H/3);

public double Bp() => (2 \* Math.PI \* R \* H);

public double Pp() => (Bp() + 2 \* S\_ocn());

}

class Program

{

static void Main()

{

Cyl cl = new Cyl();

cl.H = 8;

cl.R = 5;

Console.WriteLine(cl.Ob());

Console.ReadLine();

}

}

}

Создадим для него тесты

Для этого выбираем пункты меню Файл → Создать → Проект. В левой части открывшегося окна выбираем пункт «Тест», в правой – «Проект модульного теста»

using System;

using Microsoft.VisualStudio.TestTools.UnitTesting;

namespace UnitTestProject1

{

[TestClass]

public class UnitTest1

{

[TestMethod]

public void TestMethod1()

{

}

}

}

Добавим решение в проект.

Для этого нажмем правой кнопкой мыши по слову «Решение», выберем пункт меню «Существующий проект» и выберем нужный проект.

Далее добавляем нужный класс выбрав «Добавить» → «Существующий элемент»

В public void TestMethod1() добавим код

public void TestMethod1()

{

Cyl cl = new Cyl();

cl.R = 20;

cl.H = 15;

double res = Math.PI \* 20 \* 20 \* 15;

Assert.AreEqual(cl.Ob(), res);

}

Для проверки теста выбираем «Тест» → «Выполнить»→ «Все тесты»

В том случае, если результат совпал с ожидаемым, тест считается пройденным успешно.

Для оценки покрытия кода тестами необходимо выбрать пункты меню «Тест» → «Анализ покрытия кода тестами» → «Все тесты».

Покрытие тестами должно составлять не менее 90%

**5 Индивидуальное задание**

Разработать тесты для автоматизированного модульного тестирования, провести тестирование, выполнить оценку покрытия программного кода тестами.

1 Разработать класс «Цилиндр», реализовать нахождение объема и площадей поверхности. Составить план тестирования (предусмотреть все возможные варианты)

2 Разработать класс «Конус», реализовать нахождение объема и площадей поверхности. Составить план тестирования (предусмотреть все возможные варианты)

3 Разработать класс «Шар», реализовать нахождение объема и площадей поверхности. Составить план тестирования (предусмотреть все возможные варианты)

4 Разработать класс «Призма», реализовать нахождение объема и площадей поверхности. Составить план тестирования (предусмотреть все возможные варианты)

5 Разработать класс «Пирамида», реализовать нахождение объема и площадей поверхности. Составить план тестирования (предусмотреть все возможные варианты)

**6 Содержание отчета** (в электронном виде, в рабочей папке учащегося)

6.1 Разработанный класс

6.2 Разработанные тесты

6.3 Результат покрытия кода тестами

**Список используемых источников**

1 Синицын, С.В. Тестирование программного обеспечения. / С. В. Синицын. – М.: Интернет-университет информационных технологий, 2006 . – 367 с.