Институт информационных технологий и управления в технических системах

Кафедра информационных технологий и компьютерных систем

ОТЧЁТ

по лабораторной работе №3

«ИССЛЕДОВАНИЕ МЕХАНИЗМА ВИРТУАЛЬНЫХ ФУНКЦИЙ»

по дисциплине «Объектно-ориентированное программирование»

Выполнил студент группы ПИН/б-19-1-о

Мельник А.С.

Проверил ассистент

Тимофеев И.С.

Севастополь

2020

**1. ЦЕЛЬ РАБОТЫ**

Исследование основных средств описания виртуальных функций и использование их при написании объектно-ориентированных программ.

**2. ОПИСАНИЕ ПРОЕКТА**

Разработать программу на языке С++, в которой необходимо переопределить базовый класс как абстрактный. Один из методов базового класса необходимо переопределить в порожденном классе. Создать дополнительный порожденный класс со своими уникальными полями.

Создать указатель на базовый класс. Под указатель поместить объект производного класса. Проиллюстрировать работу программы с объектом через указатель — ввод/вывод данных, вызов функций класса.

Разработать тестовые примеры.

Выполнить отладку программы.

Сформулировать выводы.

Оформить отчет по проделанной работе.

**3. АНАЛИЗ ЗАДАЧИ**

Для варианта задания, полученного в лабораторной работе №2, необходимо выполнить следующее:

3.4.1 Переопределить базовый класс как абстрактный — для этого определить заданную вариантом функцию как чисто виртуальную. В обязательном порядке переопределить эту функцию в производном классе.

3.4.2 Дополнительно определить еще один производный класс со своими уникальными полями, переопределить в нем чисто виртуальный метод.

3.4.3 Создать указатель на базовый класс. Под указатель поместить объект производного класса. Проиллюстрировать работу программы с объектом через указатель — ввод/вывод данных, вызов функций класса.

3.4.4 Проиллюстрировать корректную работу механизма виртуальных функций — вызвать виртуальную функцию и убедиться, что тип выполняемой функции соответствует типу объекта под указателем. Поменять объект под указателем на объект второго производного класса. Проверить корректную работу механизма.

3.4.5 Описать деструкторы классов, содержащие тестовые фразы для вывода на экран. Отследить во время выполнения программы работу деструкторов по тестовым фразам — место вызова, порядок выполнения. Определить работу деструкторов с использованием виртуальных методов. Отследить изменения в работе программы

**4. ТЕКСТ С++ ПРОГРАММЫ, ЗАДАННОЙ ВАРИАНТОМ ЗАДАНИЯ**

#include <iostream>

using std::cout;

using std::endl;

class Array

{

public:

virtual void Output() = 0;

virtual void Check() = 0;

virtual ~Array() { cout << "Array - destruction" << endl; };

};

class Vector : public Array

{

public:

int Num;

int ArrNum[5];

void Input(int num, int arrNum[5]) { Num = num; for (int i = 0; i < 5; i++) ArrNum[i] = arrNum[i]; }

void Output() override { cout << "Num:" << Num << endl << "ArrNum:" << ArrNum << endl; }

void Check() override

{

for (int i = 0; i < 5; i++)

{

if (ArrNum[i] < 0)

{

cout << "Есть отрицательные элементы" << endl;

return;

}

}

}

void Minimal()

{

int min = ArrNum[0];

for (int i = 0; i < 5; i++)

{

if (min > ArrNum[i]) min = ArrNum[i];

}

cout << "Минимальный:" << min << endl;

}

Vector() {};

Vector(int num, int arrNum[5]) { Num = num; for (int i = 0; i < 5; i++) ArrNum[i] = arrNum[i]; }

~Vector() { cout << "Vector - destruction" << endl; };

};

class AdditionalVector : public Array

{

public:

int ArrNum[5];

void Input(int arrNum[5]) { for (int i = 0; i < 5; i++) ArrNum[i] = arrNum[i]; }

void Output() override { cout << "ArrNum:" << ArrNum << endl; }

void Check() override

{

for (int i = 0; i < 5; i++)

{

if (ArrNum[i] > 0)

{

cout << "Есть положительные элементы" << endl;

return;

}

}

}

AdditionalVector() {};

AdditionalVector(int arrNum[5]) { for (int i = 0; i < 5; i++) ArrNum[i] = arrNum[i]; }

~AdditionalVector() { cout << "AddVector - destruction" << endl; };

};

int main()

{

setlocale(LC\_ALL, "rus");

Array\* vector = new Vector();

int arr[5] = { -5, 17, 55, 0, -17 };

((Vector\*)vector)->Input(6, arr);

cout << "Производный:";

vector->Check();

vector->Output();

int arr1[5] = { -7, 10, 43, 4, -107 };

vector = new AdditionalVector(arr1);

cout << "Второй производный:";

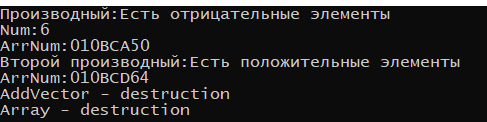
vector->Check();

vector->Output();

delete vector;

}

**5. СВЕДЕНИЯ ОБ ОТЛАДКЕ ПРОГРАММЫ**

****

**6. ВЫВОД**

В ходе лабораторной работы исследованы основные средства создания виртуальных функций и их особенности.