Министерство науки и образования РФ

Федеральное государственное автономное образовательное

учреждение высшего профессионального образования

«Санкт-Петербургский государственный электротехнический

университет «ЛЭТИ» им. В. И. Ульянова (Ленина)»

(СПбГЭТУ «ЛЭТИ»)

Факультет компьютерных технологий и информатики

Кафедра вычислительной техники

Отчёт

по лабораторной работе № 4

на тему:

“Наследование и полиморфизм”

по дисциплине “Алгоритмы и структуры данных”

Вариант 19

Выполнил студент гр. 4306: Табаков А.В.

Принял: Колинько П.Г.

Санкт-Петербург  
2016

**Цель**

Получить практические навыки работы с иерархией объектов: наследование и полиморфизм.

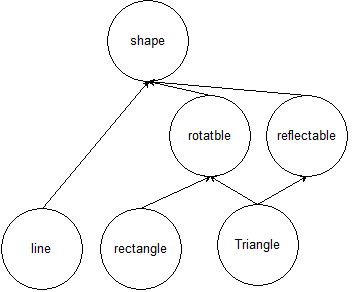
**Задание**

Составить и отладить программу выводящую на экран изображение фигуры в заданных позициях в соответствии с вариантом. Задание указано в таблице 1.

Таблица 1. Задание 19 варианта

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *№ п/п* | *Наименование* | *Вид* | *Отражение* | *Поворот* | *Расположение* |
| *8* | *Треугольник* |  | *Да* | *Да* | *Рога*  *Бакенбарды* |

**Иерархия классов**



**Пояснения**

1. Какие функции-члены пришлось переопределить и почему?

Функции: возврата крайних точек фигуры, рисования, перемещения, поворота вправо\влево, отражения по вертикали\горизонтали. Их пришлось переопределять, по двум причинам, первая: в базовых классах эти функции объявлены чисто виртуальными;

вторая: эти функции имеют специфический принцип работы, конкретно для нашего класса треугольник.

1. Какие функции-члены сделаны недоступными и каким образом это сделано?

Конструкторы копирования и переноса объектов. Закрыты модификатором доступа private.

**Демонстрация программы**

Код программы см. приложение.

Рис. 1. Инициализация фигур



Рис. 2. Подготовка фигур к стыковке



Рис. 3. Демонстрация отражений



Рис. 4. Возврат фигур после демонстрации



Рис. 5. Стыковка фигур

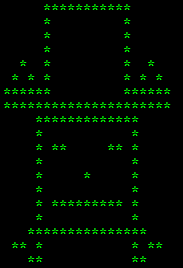


Рис. 6. Попросим помахать шляпой

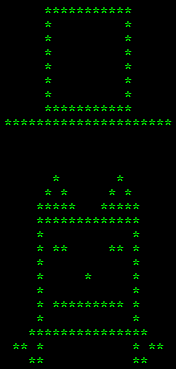
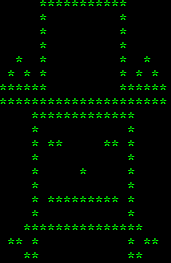


Рис. 7. Возврат шляпы



**Ответы на контрольные вопросы**

1. Так как треугольник может вращаться и отражаться, следует наследовать классы: reflectable и rotatble.
2. Для кружка следует наследовать базовый класс фигуры (shape).
3. Класс фигуры (shape).
4. При выборе типа private все данные наследуемых классов станут private в классе наследника.

Если взять protected, то данные которые были public или protected в наследуемых классах, станут protected.

Тип public не меняет права доступа наследуемых классов.

1. Можно, по умолчанию будет выбран тип private.
2. Указание программисту о том, что для класса наследника функцию стоит переопределить.
3. Требует обязательную реализацию в классе наследнике. Также класс с чисто виртуальными функциями становится абстрактным и не может быть создан объект данного класса.
4. Нет. Функции базового класса могут быть полезны и в производном, их стоит переопределять, если они реализуют не, то что требуется для данного класса.
5. Для вызова функций-членов конкретных классов.
6. Задать конструктору тип доступа private.
7. Аналогично с 10.
8. Никак. Они не подлежат изменению.
9. Через указатель, вызывая динамическое распределение памяти с конструктором производного класса.

**Вывод**

При выполнении лабораторной работы были получены практические навыки работы с иерархией объектов, такие как наследование и полиморфизм на языке программирования «С/C++».

**Список используемых источников**

* Алгоритмы и структуры данных: методические указания к лабораторным работам, практическим занятиям и курсовому проектированию. Ч.2 Вып. 1601 / сост.: П.Г. Колинько. - СПб.: Изд-во СПБГЭТУ "ЛЭТИ", 2016. - 48 с.
* Освой С++ самостоятельно за 21 день. Сиддхартха Рао. 688 стр., с ил.;

ISBN 978-5-8459-1825-3; 7 издание.

* http://stackoverflow.com – Сайт вопросов и ответов по программированию.
* http://cyberforum.ru – Форум программистов и сисадминов.
* http://alenacpp.blogspot.ru/2006/03/blog-post\_11.html - Права доступа при наследовании

**Приложение**

Source.cpp – Код программы

screen.h, shape.h – Заголовочные файлы программы