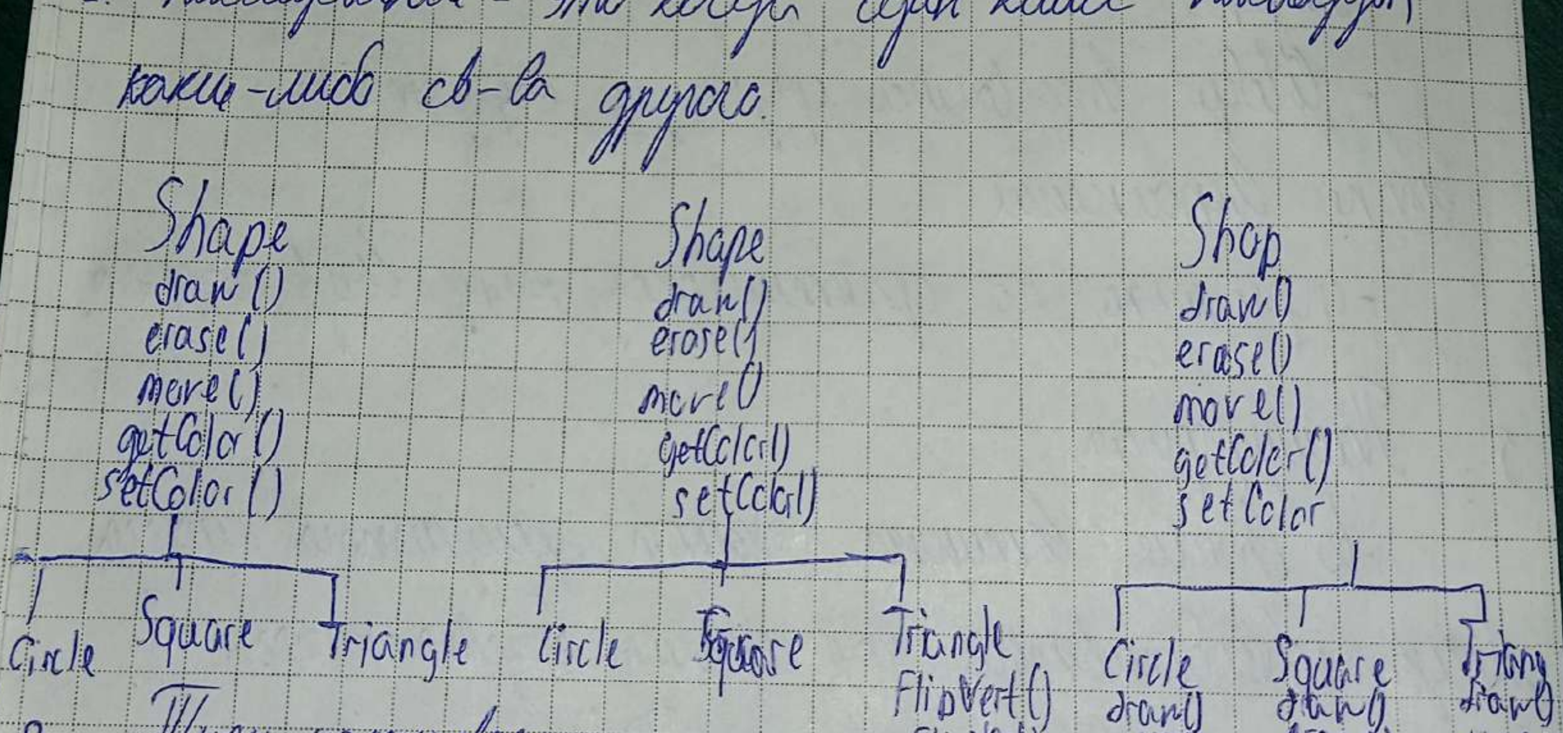
Лабораторная работа №3

Контрольные вопросы

1. Существуют два механизма повторного использования кода – наследование и композиция.

Композиция - простейший механизм для создания нового кода путем объединения нескольких объектов существующих классов в одном классе.

2. Наследование - это когда один класс наследует какие-либо св-ва другого.


Shape
draw()
erase()
move()
getColor()
setColor()
Circle
Square
Triangle
Shape
draw()
erase()
move()
getColor()
setColor()
Circle
Square
Triangle
FlipVert()
FlipHor()
Shop
draw()
erase()
move()
getColor()
setColor
Circle
draw()
erase()
Square
draw()
erase()
Triang
draw()
erase

3. Типы наследования

-публичный - публичные и защищенные данные наследуются без изменения уровня доступа к ним

-защищенный - все унаследованные данные стали защищ.

-приватный - все унаследованные данные стали приватны

Множественное наследование происходит когда подкласс имеет два или более суперкласса

4. Раннее

- выбор вызываемой функции осуществляется на этапе компиляции
- определяется по объявленному типу вызывающего объекта

Позднее

- выбор вызываемой функции осуществляется на этапе выполнения
- определяется по фактическому типу вызыв. объекта

5. Полиморфизм

- во время выполнения объекты производного класса могут обрабатываться как объекты базового класса в тех местах ~~[illegible]~~, как параметры метода а коллекции или массивы
- Базовые классы могут определять и реализовывать виртуальные методы, а производные классы - переопределять их

Виртуальные методы - позволяют работать с группами связанных объектов универсальным образом.

С их помощью можно создать иерархию классов, в которой каждый отдельный класс ~~фигур~~ является производным из общего базового.

Применять виртуальный метод для вызова соответствующего метода на любой производный класс через единый вызов в методе базового класса

6. Абстрактный класс - это класс, у которого не реализован один или более методов

Интерфейс - это абстрактный класс, у которого ни один метод не реализован, все они публичные и нет переменных класса.

Абстрактный класс нужен ~~только~~ тогда, когда нужно семейство классов, у которых есть много общего.

В C++ специального слова для обозначения интерфейсов нет.

Чисто виртуальный метод - метод класса, реализация которого отсутствует.   virtual void F()=0;

Класс, содержащий абстрактные методы принято называть абстрактным.

7. Интерфейс - это абстрактный класс с чисто виртуальными, публичными методами, задающими структуру отношение с объектами

8. Очередность вызова деструктора

· Сначала отрабатывает код деструктора самого класса (D)

- Должен быть именно подобъектом, указатели на производ.

· Затем вызываются деструкторы невиртуальных базовых классов.

· Деструкторы всех виртуальных базовых классов.

Вызов конструкторов зависит от порядка описания.

Виртуальный деструктор нужен для очистки памяти и ресурсов. Деструктор объявляют виртуальным для гарантии того, что будет вызван деструктор для производного класса

9. RTTI - динамическая идентификация типа данных)
еханизм в некоторых языках программирования
оторый позволяет определить тип данных переменной
ли объекта во время выполнения программ.

В C++ он позволяет определять тип объекта во
ремя выполнения программы
Приведение во время выполнения, которое проверяет
правильность приведения, является самым простым
одходом для определения типа объекта во
время выполнения с использованием указателя или
ссылка

Два типа приведения:
Апкастинг: когда указатель или ссылка на
объект производного класса обрабатывается как
указатель базового класса
Понижение: когда указатель базового класса или
ссылка преобразуется в указатель производного
класса
Используйте == и != в классе type_info чтобы
сравнить равны ли типы двух объектов.