

**Практическое занятие №8**  
**«Ассоциация и зависимость»**  
(Продолжительность работы 2 часа)

**Цели:**

Изучить отношения ассоциации и зависимости, получить навыки реализации ассоциации и зависимости на C++.

**1. Краткие теоретические сведения**

**Типы отношений между классами**

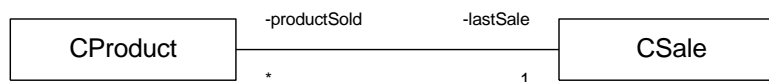
Известны три основных типа отношений между классами. Во-первых, это отношение *"обобщение/специализация"* (*общее и частное*), известное как "is-a". Розы - это цветы, т.е. розы являются специализированным частным случаем, подклассом более общего класса "цветы". Во вторых, это отношение *"целое/часть"*, известное как "part of". Так, лепестки являются частью цветов. В-третьих, это *смысловые отношения* (ассоциации). Например, розы и свечи – и то, и другое можно использовать для украшения стола.

Языки программирования выработали несколько общих подходов к выражению отношений этих трех типов. В частности, большинство объектно-ориентированных языков непосредственно поддерживают разные комбинации следующих видов отношений:

- ассоциация;
- наследование;
- агрегация;
- зависимость (использование);
- параметризация.

**Отношение ассоциации**

В предметной области розничной торговой точки можно выделить две абстракции – товары и продажи. На рис. 1 показана ассоциация, которая связывает соответствующие классы. Класс CProduct – это то, что было продано в некоторой сделке, а класс CSale – сама сделка, в которой продано несколько товаров. Эта ассоциация работает в обе стороны: задавшись товаром, можно выйти на сделку, в которой он был продан, а пойдя от сделки, найти, что было продано.



**Рисунок 1 - Ассоциация**

В Си++ подобную связь можно выразить при помощи "погребенных указателей", например, следующим образом:

```
class CProduct;
class CSale;
class CProduct {
public:
    ...
protected:
    CSale* lastSale;
};
```

```

class CSale {
public:
    ...
protected:
    CProduct** productSold;
};

```

Это ассоциация вида "один-ко-многим": каждый экземпляр товара относится только к одной последней продаже, в то время как каждый объект CSale может указывать на совокупность проданных товаров.

Как показывает этот пример, ассоциация – смысловая (семантическая) связь. По умолчанию она не имеет направления (если не указано иное, ассоциация, как в данном примере, подразумевает двухстороннюю связь) и не объясняет, как классы общаются друг с другом (мы можем только отметить семантическую зависимость, указав, какие роли классы играют друг для друга). Однако именно это требуется на ранней стадии анализа.

В предыдущем примере приведена ассоциация с *мощностью* "один ко многим". На практике можно выделить три наиболее распространенных случая мощности ассоциации:

- "один-к-одному"
- "один-ко-многим"
- "многие-ко-многим".

Мощность "один-к-одному" обозначает очень узкую ассоциацию. Например, в розничной системе продаж примером могла бы быть связь между классом CSale и классом CCreditCardTransaction: каждая продажа соответствует ровно одному снятию денег с данной кредитной карточки.

Мощность "многие-ко-многим" тоже встречается часто. Например, каждый объект класса CCustomer (покупатель) может инициировать транзакцию с несколькими объектами класса CSalePerson (торговый агент), и каждый торговый агент может взаимодействовать с несколькими покупателями.

Однако мощность ассоциации не ограничивается этими тремя случаями.

### **Отношение зависимости (использования)**

Зависимость (использование) - отношение между классами, при котором один класс в своей реализации использует в той или иной форме реализацию объектов другого класса

Во многих промышленных процессах требуется непрерывное изменение температуры. Необходимо поднять температуру до заданного значения, выдержать заданное время и понизить до нормы. Профиль изменения температуры у разных процессов разный; зеркало телескопа надо охлаждать очень медленно, а закаляемую сталь очень быстро.

Абстракция нагрева имеет достаточно четкое поведение, что дает нам право на описание такого класса – CTemperatureRamp, который по смыслу задает функцию времени от температуры:

```

class CTemperatureRamp {
public:
    CTemperatureRamp();
    virtual ~CTemperatureRamp();
    virtual void Clear();
    virtual void Bind( int temperature, int minute );
    int TemperatureAt( int minute );
protected:

```

...  
};

Управления нагревателем, поддерживающего требуемый профиль, мы от этого класса не требуем. Мы предпочитаем разделение понятий, при котором нужное поведение достигается взаимодействием трех объектов: объекта `CTemperatureRamp`, нагревателя `CHeater` и контроллера `CTemperatureController`. Класс `CTemperatureController` можно определить так:

```
class CTemperatureController {  
public:  
    CTemperatureController( int location );  
    ~CTemperatureController();  
    void Process( const CTemperatureRamp& );  
    int Schedule( const CTemperatureRamp& );  
private:  
    CHeater h;  
};
```

Класс `CTemperatureController` и `CHeater` связаны агрегацией. Класс `CTemperatureRamp` упомянут в заголовке двух методов класса `CTemperatureController`. Поэтому можно сказать, что класс `CTemperatureController` пользуется услугами класса `CTemperatureRamp`.

Отношение использования между классами соответствует равноправной связи между их объектами. Это то, во что превращается ассоциация, если оказывается, что одна из ее сторон (клиент) пользуется услугами другой (сервера).

На самом деле, один класс может использовать другой по-разному. В нашем примере это происходит в заголовке метода. Можно представить, что `CTemperatureController` внутри реализации функции `Schedule` использует, например, объект класса `CPredictor` (предсказатель значения температуры). Отношения целого и части тут ни при чем, поскольку этот объект не входит в объект `CTemperatureController`, а только используется. В типичном случае такое отношение использования проявляет себя, если в реализации какого-либо метода происходит объявление локального объекта используемого класса.

## 2. Практическое задание (100%)

Для задачи индивидуального варианта создать программу, реализующую отношение ассоциации и зависимости. Для всех закрытых полей определить метод установки значений (при недопустимых аргументах возвращать «false» и выдавать текст ошибки на экран) и метод чтения. Все классы следует наделить конструкторами и деструктором. Необходимо явно реализовать конструктор копирования и перегрузить оператор присваивания.

### Варианты заданий

1. Создать класс *квадрат*, члены класса – *длина стороны*. Предусмотреть в классе методы *вычисления и вывода сведений о фигуре – диагональ, периметр, площадь*. Создать производный класс – *правильная квадратная призма с высотой H*, добавить в класс метод *определения объема фигуры*, перегрузить методы *расчета площади и вывода сведений о фигуре*. Написать программу, демонстрирующую работу

*с этими классами: дано  $N$  квадратов и  $M$  призм, найти квадрат с максимальной площадью и призму с максимальной диагональю.*

Добавить класс окружность с закрытым полем радиус. Предусмотреть в классе методы вычисления и вывода сведений о фигуре – площади, длины окружности. Создать производный класс – сфера, добавить в класс метод определения объема фигуры, перегрузить методы расчета площади и вывода сведений о фигуре.

Добавить метод в класс квадрат, который изменяет длину стороны в зависимости от переданной вписанной окружности.

Добавить ассоциацию правильная прямоугольная призма вписана в сферу. В выводе сведений о призме выводить информацию о сфере.

*2. Создать класс треугольник, члены класса – длины 3-х сторон. Предусмотреть в классе методы проверки существования треугольника, вычисления и вывода сведений о фигуре – длины сторон, углы, периметр, площадь. Создать производный класс – равносторонний треугольник, перегрузить в классе проверку, является ли треугольник равносторонним и метод вывода сведений о фигуре. Написать программу, демонстрирующую работу с классом: дано  $K$  треугольников и  $L$  равносторонних треугольников, найти среднюю площадь для  $K$  треугольников и наибольший равносторонний треугольник.*

Создать класс окружность с закрытым полем радиус. Предусмотреть в классе методы вычисления и вывода сведений о фигуре – площади, длины окружности.

Добавить метод в класс равносторонний треугольник, который изменяет длину сторон в зависимости от переданной вписанной окружности.

Добавить ассоциацию треугольник вписан в окружность. В выводе сведений о треугольнике выводить информацию об окружности.

*3. Создать класс окружность, член класса – радиус  $R$ . Предусмотреть в классе методы вычисления и вывода сведений о фигуре – площади, длины окружности. Создать производный класс – круглый прямой цилиндр с высотой  $h$ , добавить в класс метод определения объема фигуры, перегрузить методы расчета площади и вывода сведений о фигуре. Написать программу, демонстрирующую работу с классом: дано  $N$  окружностей и  $M$  цилиндров, найти окружность максимальной площади и средний объем цилиндров.*

Создать производный класс – сфера, добавить в класс метод определения объема фигуры, перегрузить методы расчета площади и вывода сведений о фигуре.

Добавить ассоциацию цилиндр вписан в сферу. В выводе сведений о цилиндре выводить информацию о сфере.

Создать класс прямоугольник, члены класса – длины сторон  $a$  и  $b$ . Предусмотреть в классе методы вычисления и вывода сведений о фигуре – длины сторон, диагоналей, периметр, площадь. Добавить метод, который изменяет длину стороны  $b$  в зависимости от переданной вписанной окружности.

*4. Создать класс квадрат, члены класса – длина стороны. Предусмотреть в классе методы вычисления и вывода сведений о фигуре – диагоналей, периметр, площадь. Создать производный класс – правильная пирамида с апофемой  $h$ , добавить в класс метод определения объема фигуры, перегрузить методы расчета площади и вывода сведений о фигуре. Написать программу, демонстрирующую работу с классом: дано  $N$  квадратов и  $M$  пирамид, найти квадрат с минимальной площадью и количество пирамид с высотой более числа  $a$  ( $a$  вводить).*

Создать класс окружность с закрытым полем радиус. Предусмотреть в классе методы вычисления и вывода сведений о фигуре – площади, длины окружности. Создать производный класс – сфера, добавить в класс метод определения объема фигуры, перегрузить методы расчета площади и вывода сведений о фигуре.

Добавить метод в класс квадрат, который изменяет длину стороны в зависимости от переданной описанной окружности.

Добавить ассоциацию правильная пирамида вписана в сферу. В выводе сведений о призме выводить информацию о сфере.

5. Создать класс четырехугольник, члены класса – координаты 4-х точек. Предусмотреть в классе методы проверки существования четырехугольника вычисления и вывода сведений о фигуре – длины сторон, диагоналей, периметр, площадь. Создать производный класс – параллелограмм, предусмотреть в классе проверку, является ли фигура параллелограммом. Написать программу, демонстрирующую работу с классом: дано  $N$  четырехугольников и  $M$  параллелограммов, найти среднюю площадь  $N$  четырехугольников и параллелограммы наименьшей и наибольшей площади.

Создать класс окружность с закрытым полем радиус. Предусмотреть в классе методы вычисления и вывода сведений о фигуре – площади, длины окружности.

Добавить метод в класс окружность, который изменяет радиус в зависимости от переданного описанного параллелограмма.

Добавить ассоциацию окружность вписана в параллелограмм. В выводе сведений о параллелограмме выводить информацию об окружности.

6. Создать класс треугольник, члены класса – координаты 3-х точек. Предусмотреть в классе методы проверки существования треугольника, вычисления и вывода сведений о фигуре – длины сторон, углы, периметр, площадь. Создать производный класс – равносторонний треугольник, предусмотреть в классе проверку, является ли треугольник равносторонним. Написать программу, демонстрирующую работу с классом: дано  $N$  треугольников и  $M$  равносторонних треугольников, вывести номера одинаковых треугольников и равносторонний треугольник с наименьшей медианой.

Добавить класс окружность с закрытым полем радиус. Предусмотреть в классе методы вычисления и вывода сведений о фигуре – площади, длины окружности.

Добавить ассоциацию треугольник вписан в окружность. В выводе сведений о треугольнике выводить информацию об окружности.

Добавить метод в класс равносторонний треугольник, который изменяет координаты двух точек и описанную вокруг треугольника окружность в зависимости от переданной вписанной окружности.

7. Создать класс прямоугольник, члены класса – длины сторон  $a$  и  $b$ . Предусмотреть в классе методы вычисления и вывода сведений о фигуре – длины сторон, диагоналей, периметр, площадь. Создать производный класс – параллелепипед с высотой  $c$ , добавить в класс метод определения объема фигуры, перегрузить методы расчета площади и вывода сведений о фигуре. Написать программу, демонстрирующую работу с классом: дано  $N$  прямоугольников и  $M$  параллелепипедов, найти количество прямоугольников, у которых площадь больше средней площади прямоугольников и количество кубов (все ребра равны).

Добавить класс окружность с закрытым полем радиус. Предусмотреть в классе методы вычисления и вывода сведений о фигуре – площади, длины окружности. Добавить

метод в класс прямоугольник, который изменяет длину стороны  $b$  в зависимости от переданной вписанной окружности.

Добавить ассоциацию параллелепипед вписан в сферу. В выводе сведений о параллелепипеде выводить информацию о сфере.

8. Создать класс окружность, член класса – радиус  $R$ . Предусмотреть в классе методы вычисления и вывода сведений о фигуре – площади, длины окружности. Создать производный класс – конус с высотой  $h$ , добавить в класс метод определения объема фигуры, перегрузить методы расчета площади и вывода сведений о фигуре. Написать программу, демонстрирующую работу с классом: дано  $N$  окружностей и  $M$  конусов, найти количество окружностей, у которых площадь меньше средней площади всех окружностей, и наибольший по объему конус.

Создать производный класс – сфера, добавить в класс метод определения объема фигуры, перегрузить методы расчета площади и вывода сведений о фигуре.

Добавить ассоциацию конус вписан в сферу. В выводе сведений о цилиндре выводить информацию о сфере.

Создать класс прямоугольник, члены класса – длины сторон  $a$  и  $b$ . Предусмотреть в классе методы вычисления и вывода сведений о фигуре – длины сторон, диагоналей, периметр, площадь. Добавить метод, который изменяет длину стороны  $b$  в зависимости от переданной описанной окружности.

9. Создать класс четырехугольник, члены класса – координаты 4-х точек. Предусмотреть в классе методы вычисления и вывода сведений о фигуре – длины сторон, диагоналей, периметр, площадь. Создать производный класс – равнобокая трапеция, предусмотреть в классе проверку, является ли фигура равнобокой трапецией. Написать программу, демонстрирующую работу с классом: дано  $N$  четырехугольников и  $M$  трапеций, найти максимальную площадь четырехугольников и количество четырехугольников, имеющих максимальную площадь, и трапецию с наименьшей диагональю.

Создать класс окружность с закрытым полем радиус. Предусмотреть в классе методы вычисления и вывода сведений о фигуре – площади, длины окружности.

Добавить метод в класс окружность, который изменяет радиус в зависимости от переданной описанной трапеции.

Добавить ассоциацию окружность вписана в трапецию. В выводе сведений о трапеции выводить информацию о сфере.

10. Создать класс равносторонний треугольник, член класса – длина стороны. Предусмотреть в классе методы вычисления и вывода сведений о фигуре – периметр, площадь. Создать производный класс – правильная треугольная призма с высотой  $H$ , добавить в класс метод определения объема фигуры, перегрузить методы расчета площади и вывода сведений о фигуре. Написать программу, демонстрирующую работу с классом: дано  $N$  треугольников и  $M$  призм. Найти количество треугольников, у которых площадь меньше средней площади треугольников, и призму с наибольшим объемом.

Добавить класс окружность с закрытым полем радиус. Предусмотреть в классе методы вычисления и вывода сведений о фигуре – площади, длины окружности. Создать производный класс – сфера, добавить в класс метод определения объема фигуры, перегрузить методы расчета площади и вывода сведений о фигуре.

Добавить метод в класс равносторонний треугольник, который изменяет длину сторон в зависимости от переданной вписанной окружности.

Добавить ассоциацию правильная треугольная призма вписана в сферу. В выводе сведений о призме выводить информацию о сфере.

11. Создать класс треугольник, члены класса – длины 3-х сторон. Предусмотреть в классе методы проверки существования треугольника, вычисления и вывода сведений о фигуре – длины сторон, углы, периметр, площадь. Создать производный класс – прямоугольный треугольник, предусмотреть в классе проверку, является ли треугольник прямоугольным. Написать программу, демонстрирующую работу с классом: дано  $N$  треугольников и  $M$  прямоугольных треугольников, найти треугольник с максимальной площадью и прямоугольный треугольник с наименьшей гипотенузой.

Создать класс окружность с закрытым полем радиус. Предусмотреть в классе методы вычисления и вывода сведений о фигуре – площади, длины окружности.

Добавить ассоциацию прямоугольный треугольник вписан в окружность. В выводе сведений о прямоугольном треугольнике выводить информацию об окружности.

Добавить метод в класс прямоугольный треугольник, который изменяет длину одного катета и гипотенузы и радиус описанной вокруг треугольника окружности в зависимости от переданной вписанной окружности.

12. Создать класс четырехугольник, члены класса – координаты 4-х точек. Предусмотреть в классе методы вычисления и вывода сведений о фигуре – длины сторон, диагоналей, периметр, площадь. Создать производный класс – квадрат, предусмотреть в классе проверку, является ли фигура квадратом. Написать программу, демонстрирующую работу с классом: дано  $N$  четырехугольников и  $M$  квадратов, найти четырехугольники с минимальной и максимальной площадью и номера одинаковых квадратов.

Создать класс окружность с закрытым полем радиус. Предусмотреть в классе методы вычисления и вывода сведений о фигуре – площади, длины окружности.

Добавить ассоциацию окружность описана вокруг квадрата. В выводе сведений о квадрате выводить информацию об окружности

Добавить метод в класс квадрат, который изменяет координаты трех точек квадрата и описанную окружность от переданной вписанной окружности.

13. Создать класс треугольник, члены класса – длины 3-х сторон. Предусмотреть в классе методы проверки существования треугольника, вычисления и вывода сведений о фигуре – длины сторон, углы, периметр, площадь. Создать производный класс – равнобедренный треугольник, предусмотреть в классе проверку, является ли треугольник равнобедренным. Написать программу, демонстрирующую работу с классом: дано  $N$  треугольников и  $M$  равнобедренных треугольников, найти среднюю площадь для  $N$  треугольников и равнобедренный треугольник с наименьшей площадью.

Создать класс окружность с закрытым полем радиус. Предусмотреть в классе методы вычисления и вывода сведений о фигуре – площади, длины окружности.

Добавить ассоциацию равнобедренный треугольник вписан в окружность. В выводе сведений о равнобедренном треугольнике выводить информацию об окружности.

Добавить метод в класс равнобедренный треугольник, который изменяет длину основания и радиус окружности описанной вокруг равнобедренного треугольника в зависимости от переданной вписанной окружности.

14. Создать класс квадрат, член класса – длина стороны. Предусмотреть в классе методы вычисления и вывода сведений о фигуре – периметр, площадь, диагональ.

*Создать производный класс – куб, добавить в класс метод определения объема фигуры, перегрузить методы расчета площади и вывода сведений о фигуре. Написать программу, демонстрирующую работу с классом: дано N1 квадратов и N2 кубов. Найти среднюю площадь квадратов и количество кубов с наибольшей площадью.*

Создать класс окружность с закрытым полем радиус. Предусмотреть в классе методы вычисления и вывода сведений о фигуре – площади, длины окружности. Создать производный класс – сфера, добавить в класс метод определения объема фигуры, перегрузить методы расчета площади и вывода сведений о фигуре.

Добавить метод в класс квадрат, который изменяет длину стороны в зависимости от переданной описанной окружности.

Добавить ассоциацию куб вписан в сферу. В выводе сведений о кубе выводить информацию о сфере

*15. Создать класс четырехугольник, члены класса – координаты 4-х точек. Предусмотреть в классе методы вычисления и вывода сведений о фигуре – длины сторон, диагоналей, периметр, площадь. Создать производный класс – ромб, предусмотреть в классе проверку, является ли фигура ромбом. Написать программу, демонстрирующую работу с этими классами: дано N четырехугольников и M ромбов, найти четырехугольник с минимальным периметром и среднюю площадь ромбов.*

Создать класс окружность с закрытым полем радиус. Предусмотреть в классе методы вычисления и вывода сведений о фигуре – площади, длины окружности.

Добавить метод в класс окружность, который изменяет радиус в зависимости от переданного описанного ромба.

Добавить ассоциацию окружность вписана в ромб. В выводе сведений о ромбе выводить информацию о сфере.

### **3. Список рекомендуемой литературы**

1. Павловская Т. А. C/C++. Программирование на языке высокого уровня : для магистров и бакалавров : учеб. для вузов / Т. А. Павловская. - Гриф МО. - Санкт-Петербург: Питер, 2013. - 460 с. : ил.
2. Professional C++, 3rd Edition. Marc Gregoire. ISBN: 978-1-118-85805-9. Paperback 984 pages. September 2014

### **4. Контрольные вопросы**

1. В случае реализации отношения использования как может быть передана информация об используемом объекте?
2. В случае реализации отношения ассоциации как может храниться информация о связанных объектах?
3. Привести 3 примера отношения ассоциации.