**Задание 1 лаб." Объектно-ориентированный анализ, проектирование и использование полиморфизма при позднем связывании в иерархии классов "**

состоящую из 2 задач (схем) " **Объектно-ориентированный анализ и моделирование, разработка статических моделей** " и использование

4-х этапов (а,б,в,г) для каждая из задач:

**а). Проектирование моделей в стандарте UML.**

**б). Программная реализация на JAVA в отдельном java-пакете.**

**в). Использование и полученного пакета для разработки программной системы.**

**г). Придумать для данных схем (моделей) описание (легенду) задачи реального мира. (это в конце, для обеих моделей).**

***Примечание*. сначала выполнить проектирование, пункт а.**

Задача 1:

**Этап а:**

а) Для схемы 1:

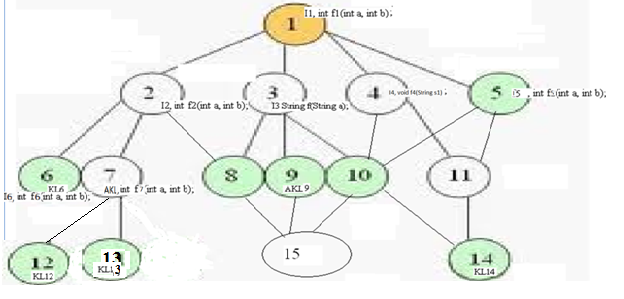
1.1. **Доопределить** схему (идентифицировать тип сущности при этом, предложить наиболее обоснованное возможное решение, по минимизации уровня абстракции каждого не определенного узла, сохраняя уже имеющиеся определенные ) и дописать до, как минимум, по одной собственной функции для каждой сущности.

**1.2. Для схемы Разработать Модель** классов в стандарте UML (Enterprise Architect), касается отношений (зависимостей между сущностями), используя имена, предложенные на схеме, которую Вы доопределили и дописать, как минимум по одной функции для каждой сущности.

**Этап б.**

**Реализовать** **модель** в программном коде JAVA):

СХЕМА 1, подробное описание:



, где I – это интерфейс; f() – это функция принадлежащая данной абстракции. Причем в каждой из абстракций (I,AKL,KL) своя дополнительная функция (f5, f6, f7, f8,…);

AKL - абстрактный класс (создать его как таковой, часть функций в нем должна быть реализована)? ;

KL настоящий (реальный) класс.

**2.1. Доопределить** листья дерева как таковые (их тип (I/AKL/KL))и предполагаемую функцию Вы должны сами, так, чтобы это было правильно с точки зрения формальной логики построения иерархии сущностей и фундаментальных свойств ООП (инкапсуляция, наследование и полиморфизм), и с максимально возможным уровнем реализации (минимизировать абстракции, где это можно в данной схеме, не меняя типа указанных сущностей (интерфейсы должны оставаться интерфейсами, абстрактные классы – абстрактными классами, классы – классами).

Необходимо:

2.2.1. Разработать правильный проект этой структуры, проставить тип сущностей (интерфейс, абстрактный класс, класс), объявить или реализовать, где это необходимо! простенькие методы, типа: сложение двух чисел, вывод чего-то и т.п.

Для этого:

* Проанализировать модель и расписать иерархическую структуру программных абстракций (расписать иерархию абстракций с методами (функциями), где какие можно поставить программные сущности (KL, AKL,I), исходя из логики схемы и правил определения производных сущностей (нижние элементы – все должны быть реальными классами, чтобы их можно было использовать для создания объектов), т.е. определить (снизу вверх) сначала KL, везде, где это допустимо, затем AKL везде, где это допустимо, затем

I-интерфейсы), где не может быть чего-то менее абстрактного.

* Ввести методы, как минимум по одному в интерфейсах и других программных сущностях. Реализовать эти методы, в местах, где это возможно, но так, чтобы абстрактные классы в действительности соответствовали своему типу (оставались абстрактными).

**2.2 Выполнить** программную реализацию (это второй этап по каждой задаче) в java-коде этой структуры. И далее считать, что Вы получаете такую иерархию «интерфейсов-абстрактных классов-классов» в качестве отдельного пакета, от «субподрядчика» (Аутсорсинговой компании) который Вам можно использовать для решений своих профессиональных задач.

**ЭТАП в.**

ИСПОЛЬЗОВАТЬ пакет в разработке системы.

**3.1. В отдельном (своем) пакете** вне реализованной иерархии, подключите пакет с реализацией иерархии, объявите класс My\_class, реализуйте в метод main() и необходимые объекты, чтобы продемонстрировать вызовы, любого из методов данной иерархии через имя объекта класса (правильно расставляйте ограничения доступа, учитывайте , что вызов методов осуществляется из другого пакета.

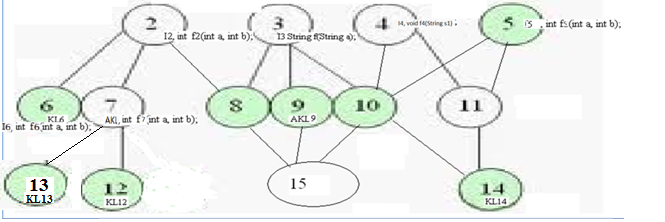
**3.2. Показать, также (написать в коде java),** как и какие методы можно вызвать через любую из интерфейсных ссылок (например I1 i2; I3 k2; …,и т.д.) и через любую из ссылок на абстрактный класс (например, AKL2 f2; …, и т.д.) (…это - приведение объекта к типу интерфейса (полиморфизм при позднем связывании)).

**3.3. Показать** (написать в коде Java), как реализуется приведение интерфейса к типу объекта (полиморфизм при позднем связывании), для вызова через ссылки на абстрактные сущности (интерфейс и абстрактный класс) методов, которые не являются собственностью абстрактных сущностей.

**Задача 2.**

**То же, только структура, такая:**

**СХЕМА 2**

****

**Здесь также необходимо доопределить иерархию (расставить тип и идентификаторы сущностей, определить методы и далее согласно формулировкам задания , приведенного выше описания выполнить пункты этапов 1,2,3.**

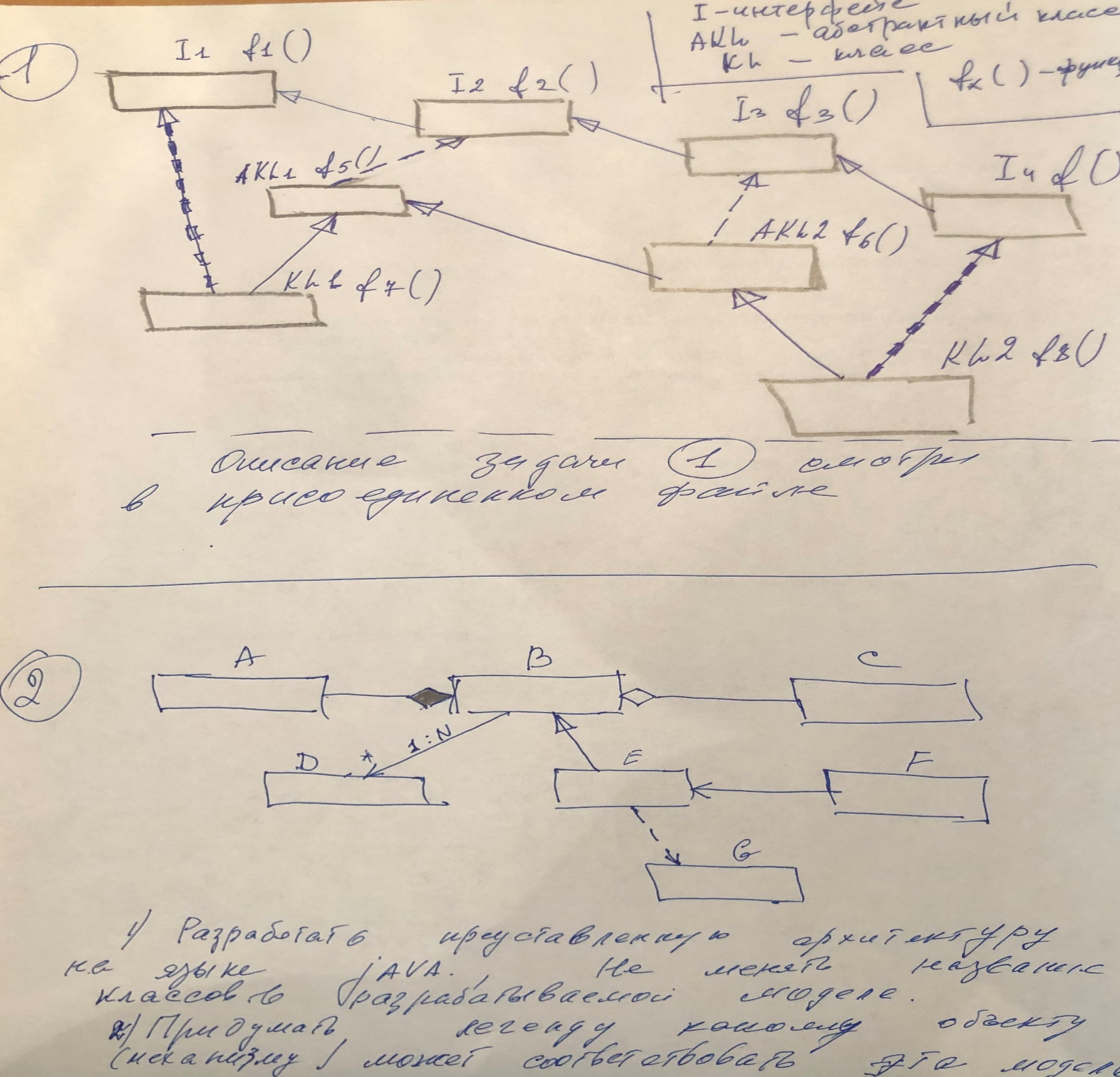
**Этап г (не обязательны, но очень желательный на +).**

**Придумать для данных схем (моделей) описание (легенду) задачи реального мира, для которой бы подошло описание подобными структурами**

**Задача 2 " Объектно-ориентированный анализ, проектирование и использование полиморфизма в иерархии классов программного кода, при позднем связывании»**

На модели Рис. 1. представлена иерархия из интерфейсов I1,I2,I3,I4, двух абстрактных классов AKL1 и AKL2 и 2-х классов KL1 и KL2. (КЛАССЫ, ИНТЕРФЕЙСЫ, АБСТРАКТНЫЕ КЛАССЫ И ФУНКЦИИ ИМЕНОВАТЬ ТАК, КАК ОНИ НАЗВАНЫ НА ДИАГРАММАХ)

* 1. **Выполнить** программную реализацию  в java-коде этой структуры. И далее считать, что Вы получаете такую иерархию «интерфейсов-абстрактных классов-классов» в качестве отдельного пакета (API), от субподрядчика (*Аутсорсинговой компании*)  и хотите использовать данный пакет для решения ваших профессиональных задач.
  2. **В отдельном пакете** вне этой иерархии подключите пакет, содержащий код иерархической модели, объявите класс My\_class, в нем  реализуйте статический метод main(…) {…} и необходимые объекты, чтобы продемонстрировать вызовы, любого из методов иерархии подключаемого пакета, через ссылку на объекта класса (правильно расставляйте ограничения доступа, учитывайте , что вызов осуществляется из другого пакета.
  3. **Показать, также (написать в коде java),** как и какие методы можно вызвать через любую из интерфейсных ссылок (например I1, i2,   I3 , I4.) и через любую из ссылок на абстрактный класс ( AKL1 и AKL2 ) (…это – метод приведение объекта к типу интерфейса абстрактного класса)).
  4. Написать программный код на листе бумаги, название классов, абстрактных классов и интерфейсов не менять, быть готовым объяснить код и реализацию любых изменений в коде.

(Более подробное описание требований к задачам такого типа смотри ПРИЛОЖЕНИЕ 1.)

**Придумать для данных схем (моделей) описание (легенду) задачи реального мира, для которой бы подошло описание подобными структурами**