**Семинар 2. Класс как тип. Методы**

**Задача 1.** Территория Рубинового Королевства имеет форму правильного n-угольника со стороной некоторого вещественного размера. Количество смотровых башен на границах королевства определяется как . Определить класс Рубиновое Королевство с необходимыми полями. Для подсчета количества башен в королевстве и площади территории использовать свойства. В основной программе запросить у пользователя необходимые для создания экземпляра королевства данные и вывести всю информацию о нем.

**Задача 2.** Определить статический класс **SimpleFunctionsIntegrals**, содержащий методы для вычисления интегралов функцийна заданном промежутке. В основной программе:

1. Ввести границы интервала и вывести значения интегралов всех четырех функций на этом промежутке (если какой-то из интегралов нельзя посчитать на заданном промежутке вывести соответствующее сообщение об этом).

2. Посчитать сумму результатов вычисления интегралов (если какой-то из интегралов нельзя вычислить, не учитывать его).

3. Определить метод **GenerateSpell,** получающий на вход сумму результатов вычисления интегралов, в котором случайным образом генерировать два целых числа – левую и правую границы интервала. Если сумма интегралов входит в этот промежуток, выводить на консоль заклинание, случайно выбираемое из заранее заданного массива, в противном случае выводить: *“You are too tired to generate spell :(”*

**Задача 3.** Определить класс волшебных палочек **MagicWand**. Поля класса – материал древесины **wood**, материал сердцевины core, длина **length** (вещественное число). Свойство с возвращаемым значением строкового типа – гибкость **Flexibility**, зависит от коэффициента **:**

|  |  |
| --- | --- |
|  | *“low”* |
|  | *“middle”* |
|  | *“high”* |

Определить общедоступный метод **()**, выбирающий волшебника с вероятностью 30%, а также метод (), формирующий и возвращающий строку с информацией о волшебной палочке.

В основной программе создать экземпляр волшебной палочки, значения и выбрать из заранее заданных массивов случайным образом, – сгенерировать случайно в диапазоне [10, 15). Вывести сведения о палочке и информацию о том, выбрала ли она волшебника.

**Задача 4.** Определите класс артефактов с инкапсулированным свойством и статическим полем – счетчик количества артефактов.

Определите класс с инкапсулированными свойствами и . Если в свойства пытаются записать некорректные значения, заменяйте их на умалчиваемые.

Определите класс **Goblin** c инкапсулированным свойством **Surname**.

Определите статический метод string CreateNewArtifact(Wizard wizard, Artifact prefab), инициализирующий поля prefab в зависимости от волшебника. Если его опыт больше 100, то артефакт успешно создается, и его легендарность равна 100. Иначе с вероятностью 50% артефакт не создается совсем, с вероятностью 50% -- создается, и его легендарность – это случайное число в диапазоне от опыта волшебника до 100. Метод возвращает сообщение об успешности создания артефакта: *“{Wizard} creates artifact with legendary {Legendary}." Или "{wizard.Surname} could not create artifact."* Соответственно.

Определите метод **UseArtifact**как с параметром типа , так и с параметром типа . В случае, если параметр типа , метод возвращает сообщение об успешности применения артефакта: *"{person} uses artifact successfully!"* или *"{person} tried to use artifact!.. But he is goblin so he will be cursed!..*" соответственно.

В основной программе смоделируйте будни волшебного мира. Для этого создайте 5 экземпляров класса **Wizard** и 5 экземпляров класса **Goblin** со случайно инициализированными полями.

Создайте 10 «пустых» экземпляров класса артефактов. Пусть каждый из волшебников попробует создать артефакт, а затем каждый из волшебников и гоблинов попробует применить артефакт. Выводите на экран сообщения об успешности выполнения всех действий. Выведите количество созданных артефактов, используя статическое поле. Запишите в файл, находящийся в папке решения, информацию об артефактах.