

Задание 1.

1. Разработать тип, реализующий алгоритм Евклида для вычисления НОД двух целых чисел (http://en.wikipedia.org/wiki/Euclidean_algorithm).
2. Добавить к разработанному типу дополнительную функциональность в виде перегруженных методов
3. вычисления НОД для трех, четырех или пяти целых чисел.
4. Добавить к разработанному типу метод, реализующий алгоритм Стейна (бинарный алгоритм Эвклида)
5. для расчета НОД двух целых чисел (http://en.wikipedia.org/wiki/Binary_GCD_algorithm). Метод должен
6. принимать выходной параметр, содержащий значение времени, затраченное для выполнения расчетов.
7. Изменить метод, реализующий алгоритм Евклида вычисления НОД двух параметров, добавив
8. аналогичную функциональность.
9. Дополнить методом, подготавливающим данные для построения гистограммы, сравнивающей время нахождения решения каждым из методов.
10. Создать unit-тесты для тестирования разработанных методов.
11. Весь код должен быть снабжён комментариями (автокомментариями)
12. Проект должен быть размещён в репозитории

Задание 2

1. Использовать возможности ООП: классы, наследование, полиморфизм, инкапсуляция.
2. При кодировании должны быть использованы соглашения об оформлении кода (C# code convention).
3. Классы должны быть грамотно разложены по библиотекам.
4. В классах должны быть методы ToString(), GetHashCode() и Equals().
5. Создать класс, позволяющий загружать данные из текстового файла в **массив**.
6. Необходимо создать классы, для работы с фигурами, заданными на плоскости
7. Фигуры могут задаваться размерами своих сторон, или координатами вершин
8. Необходимо предусмотреть у каждой фигуры методы для вычисления площади и периметра
9. Найти в массиве все фигуры, равные данной
10. Создать unit-тесты для тестирования разработанных методов.
11. Весь код должен быть снабжён комментариями (автокомментариями)
12. Проект должен быть размещён в репозитории