|  |  |
| --- | --- |
| Вариант 1 | Вариант 2 |
| 1)Реализовать класс делегат с сигнатурой  delegate double MyDel(int[] a);  2)Реализовать анонимный метод, вычисляющий среднее значение массива  3)Написать 10 unit тестов для проверки корректности вычислений. | 1)Реализовать класс делегат с сигнатурой  delegate double MyDel(int[] a);  2)Реализовать анонимный метод, вычисляющий разность между наибольшим и наименьшим элементом массива.  3)Написать 10 unit тестов для проверки корректности вычислений. |
| 2)Реализовать класс делегат с сигнатурой  delegate double MyDel(MyDelegate[] a);  2)по веденному с к клавиатуры знаку + или – реализовать 2 лямбда оператора:   1. (x, y) => { return x + y; } 2. (x, y) => { return x - y; }   Если введен другой оператор выдавать ошибку.  3)Написать 10 unit тестов для проверки корректности вычислений. | 2)Реализовать класс делегат с сигнатурой  delegate double MyDel(MyDelegate[] a);  2)по веденному с к клавиатуры знаку \* или / реализовать 2 лямбда оператора:   1. (x, y) => { return x \* y; } 2. (x, y) => { return x / y; }   Если введен другой оператор выдавать ошибку.  3)Написать 10 unit тестов для проверки корректности вычислений. |
| 3) Для класса Computer из прошлого урока опишите реализацию событий:   1. Добавление нового пользователя 2. Замена процессора 3. Установка нового ПО 4. Замена оперативной памяти   Для каждого события реализовать add, remove. | 3) Для класса Console из прошлого урока опишите реализацию событий:   1. Добавление нового пользователя 2. Подключение внешнего накопителя 3. Установка новой игры 4. Замена геймпада   Для каждого события реализовать add, remove. |
| 4)Запустите Unit тесты из задания 1 и 2 использую утилиту nunit-console.  <https://nunit.org/docs/2.4.6/consoleCommandLine.html>  <https://github.com/nunit/nunit-console/releases/tag/v3.11.1> | 4)Запустите Unit тесты из задания 1 и 2 использую утилиту nunit-console.  <https://nunit.org/docs/2.4.6/consoleCommandLine.html>  <https://github.com/nunit/nunit-console/releases/tag/v3.11.1> |