

# EPAM Systems, RD Dep.

## Homework Task

# 09 – Extended C#

---

REVISION HISTORY					
Ver.	Description of Change	Author	Date	Approved	
				Name	Effective Date
1.0	Initial version	Alexander_Kuznetsov1	02.03.2012		
2.0		Nikolay Piskarev	29.05.2017		

## Задание 1

Напишите расширяющий метод, который определяет сумму элементов массива.

## Задание 2

Напишите расширяющий метод, который определяет, является ли строка положительным целым числом. Методы Parse и TryParse не использовать.

## Задание 3

Написать методы поиска элемента в массиве (например, поиск всех положительных элементов в массиве) в виде:

1. Метода, реализующего поиск напрямую;
2. Метода, которому условие поиска передаётся через делегат;
3. Метода, которому условие поиска передаётся через делегат в виде анонимного метода;
4. Метода, которому условие поиска передаётся через делегат в виде лямбда-выражения;
5. LINQ-выражения

Сравнить скорость выполнения вычислений.

### **Примечания:**

1. Не забывайте, что LINQ-выражения выполняются в отложенном режиме, так что если вы вдруг получили 0, не удивляйтесь – вы не то измеряете.
2. Для выполнения временных замеров стоит использовать класс *Stopwatch*
3. Выберите размер поискового массива с запасом. Лучше поставить размер не менее нескольких сотен элементов.
4. Выполните каждое измерение в цикле не менее нескольких десятков раз. Определите медиану измерений (значение, расположенное посередине, если результаты измерений отсортировать по возрастанию, или убыванию).



## Требования к оформлению

1. Для каждой домашней работы необходимо создать собственную папку в корне репозитория.
2. Каждое задание должно представлять собой отдельный проект (project) в рамках общего решения (solution).
3. Правила именования:  
Все названия и имена должны быть переведены на английский язык. Именованье транслитерацией не допускается.
  - a. Маска именования репозитория: EPAM\_Ext\_Lab\_<Квартал старта группы>\_<Год>\_<Фамилия\_Имя>.  
*Например, EPAM\_Ext\_Lab\_Q2\_2017\_Ivanov\_Ivan*
  - b. Маска именования решения: HMT\_<Порядковый номер>.  
*Например, HMT\_01, HWT\_11.*
  - c. Маска именования проектов: <Задача><Номер задачи>.  
*Например, Task01, Task12.*
  - d. Каждое задание должно быть оформлено следующим образом:
    1. Формулировка задания в виде комментария (просто копировать-вставить из этого файла).
    2. Код, решающий задачу.
4. Написать письмо преподавателю, информирующее о том, что домашняя работа выполнена. Тема письма должна выглядеть так: [STYYYY\_N]Фамилия\_Имя\_TaskNN. Здесь ST – сокращённое наименование программы (Students Training), YYYY – номер года, N – номер группы, NN – порядковый номер задания в виде двух цифр, напр. 01, 05, 12 и т.д.  
*Например, ST2017\_1 Иванов Иван Task01.*

## Рекомендации к оформлению кода

1. При добавлении нового файла желательно вверху файла добавить блок с комментарием, для чего предназначен этот файл.

2. При добавлении метода в блоке `<summary></summary>` описать предназначение метода, его входных/выходных параметров и возвращаемых значений.

3. Использовать утилиту StyleCop для проверки правил оформления кода. Перед отправкой задания на проверку запускать StyleCop и исправлять все предупреждения, возникающие в секции Warning Visual Studio. Предупреждения об отсутствии документации можно игнорировать (либо настроить StyleCop таким образом, чтобы он не проверял документирование кода).