# Exercise: B-Trees 2-3-Trees and Red-Black Trees

This document defines the lab for ["Data Structures – Advanced (C#)" course @ Software University](https://softuni.bg/trainings/4273/data-structures-advanced-with-csharp-november-2023).   
Please submit your solutions (**source code**) of all below described problems in [Judge](https://judge.softuni.org/Contests/2603/02-B-Trees-2-3-Trees-and-Red-Black-Trees-Exercise).

## Red-Black Tree

You are given class **RedBlackTree<T>** your task is to implement all the methods with missing implementation:

* void **Insert**(T element) – adds **new** **entry** to the tree
* int **Count()** – returns the **size** of the tree
* **RedBlackTree<T> Search(T element)** - search for an element and return the entire subtree
* void **DeleteMin**() – removes the **min** element by key in the tree. Throws **InvalidOperationException** if the tree is empty
* void **DeleteMax**() – removes the **max** element by key in the tree. Throws **InvalidOperationException** if the tree is empty
* void **Delete**(T element) – removes the specified element. Throws **InvalidOperationException** if the tree is empty
* void **EachInOrder**(Action<T> action) – Iterates the tree and executes the action in-order