Andreas Wachs Hjalager
Student ID: 19167

Assignment 01

BDSA2021
September 12, 2021

1 C#

1.1 Generics

The type constraints for the first version of `GreaterCount`, it constraints the type `T` to be implementing the `IComparable` interface, ensuring that items of type `T` can be compared between themselves. The generic type `U` is not having any constraints applied to it, but it is neither used in this case.

Additionally, the `item` object type needs to be implementing `IEnumerable` with the generic type `T` provided.

The type constraints for the second version of `GreaterCount`, we achieve the same result as the above answer, but through transitivity where the constraints are that `T` is of same type of `U` and type `U` implements the `IComparable` type with type `U`. This is only accepted if the `IComparable` interface allows for implementations for the type `U`.

2 Software Engineering

2.1 Exercise 1

2.2 Exercise 2

2.3 Exercise 3

2.4 Exercise 4

2.5 Exercise 5

2.6 Exercise 6