A03 – DATA STRUCTURES

OVERVIEW

Data structures are an integral part of most computer programs. Many popular structures are already implemented in the .NET Framework.

In this assignment, you will design tests to measure the performance of some data structures.

OBJECTIVES

- Implement various .NET supported Data Structures in desktop applications.

ACADEMIC INTEGRITY AND LATE PENALTIES

- Please refer to the SET Policies document regarding Academic Integrity Information
- Please refer to the SET Policies document regarding Late Policy

EVALUATION

- Please see the Rubric for this assignment in eConestoga

PREPARATION

- Review the lesson content on Data Structures

REQUIREMENTS

Submission: Groups of 2, or individual

1. Examine and explore the usage of these three data structures in .NET: List, Dictionary and HashTable.
2. By determining the time it takes to insert data, retrieve data and retrieve a set of sorted data, draw conclusions as to what data structures might be best in specific situations.
3. Discuss your results in a report. This report should suggest the type of data structure to use under certain conditions. In addition, you should explain the reasons for the differences in performance. If there any anomalies, please explain them as well. (This should be one to two pages in total.)

4. It is most appropriate to use one or more console applications. No user input is required. If you feel you want to implement user input, you may do so.
5. SET programming standards must be adhered to. This includes proper commenting, including header comments and comments for classes and methods.

Things to consider:

- The design of your tests is important. If you are measuring durations that are very small, your results will likely be inaccurate. You may need to run that certain tests multiple times and find an average duration.
- Tests should be run with a variety of sample sizes. There may be different relative results between small data sets and large data sets.
- Eliminate sources that may affect your results in a negative way. For example, screen output is very time consuming.
- You will certainly need to do some research to help explain some of your observations.
- You should use the Stopwatch class in the System.Diagnostics namespace for accurate measurement of code execution.

FILE NAMING REQUIREMENTS

- There are no specific file naming requirements. Please keep the name short, but meaningful.
- The submitted file must be a .zip file (or fully compatible).

SUBMISSION REQUIREMENTS

- The full C# solution (or multiple solutions) and the report should be submitted in one zip folder.
- The report must be either a Word document or a .pdf file.
- Only one submission is required per group. However, in the case of a group submission, please make sure you identify the group members in the description you can provide with your submission in eConestoga.