

FAST NUCES – IPT – Assignment 1

Due: Sunday 24th September 2017

If Slate access is available to TA, the assignments will be submitted via SLATE. Alternatively, please email to rhasnani@yahoo.com AND marjan.uzair@gmail.com

Please put 'Assignment #1 <Reg #>' in the subject line.

PLEASE DO NOT COPY

Part A

1. Create a custom class called DynamicIntArray that implements a dynamic array of integers.
 - a. Keep a data member to store the current size of the array
 - b. Keep a data member to store the current capacity of the array
 - c. Keep a data member to store the actual array.
 - d. Provide a default (no argument) constructor that sets the current capacity to 10 and allocates memory for a ten element array of integers.
 - e. Provide another constructor that takes one argument to specify the initial capacity of array.
 - f. Provide a method "void Add(int)" that appends an element to the end of the array. If the current size of the array reaches the capacity of the array, more memory is allocated to extend the capacity of the array. [HINT: Use Array.Resize() method]
 - g. Provide a method "int Get(int)" that returns the element value of the index specified by the argument.
 - h. Provide a method "int IndexOf(int)" that finds the element value specified by the argument and returns the index where it is found, or -1 if it is not found.
2. Write a program that compares the performance of the above Dynamic Array with C# array, ArrayList and List<int>.
 - a. Populate the collections by generating 1M (one million) random values
 - b. Carry out traversal and find the sum of the element values. Print the sum and the time required to carry out the traversal in each case.
 - c. Search five randomly chosen values from each of the collection by calling the IndexOf() function and compare response times.

Part B:

1. Convert your DynamicIntArray class to a template/generic DynamicArray<T> class that can be an array of any type T.
2. Implement the IList<T> interface to the class.
3. Compare performance of DynamicArray<T> with List<T> and C# array for data types:
 - a. Double
 - b. Decimal

c. Boolean

Grading Criteria:

TBA