Generics

1) Build a generic class that represents a stack data structure (i.e., GenericStack<T>).
   a) ***void push(T element)***
   b) ***T pop()***
   c) ***boolean isEmpty()***
   d) ***T peek()***

   Use an array list to save values(save your time as well)

2) Generic method demo
   a) Bild a static method that prints a generic array of any type
   (static<T> void print (T[] array))
   b) Build a static method that prints only generic array of Number
   c) Build a sort method that sorts an array of generic type elements

3) Wildcard Demo
   a) Write a method that uses bounded wildcard
   b) Write a method that uses lower bound wildcard

4) Create a Pair class that encapsulates two objects of the same data type in an instance of Pair.
   Write a generic static method that returns the smallest value in an instance of Pair.

5) Create a Triplet class that encapsulates three objects of the same data type in a given instance of Triplet

6) Create an Association class that encapsulates two objects of different types

7) BONUS QUESTION
   Build a generic Binary Search Tree (insert, search and inOrderTraverse methods, delete on your own)