LAB 3 – WRAPPER & ITERATOR

Write a fake vector class. The class should use the built-in vector/list/array in your language of choice to effectively "wrap" inside of it an existing vector, while presenting a limited vector-like functionality to the user. Here is a TypeScript interface that represents what your class should look like:

```
interface Vector<T> extends Iterable<T> {
 get(index: number);
 set(index: number, value: T);
 length: number;
 push(value: T);
 pop(): T;
 insert(index: number, value: T);
 // remember to implement the iterable functionality
}
```

If you are working in another language, you may translate this interface into a C# interface, Java interface, VB interface, or C++ pure virtual class, because the class you write must implement/inherit the interface/class that is shown above. If you are programming in raw JavaScript (ES6/ES2015), you cannot implement the interface, but you should carefully program your class to work identically to the presented interface.

Please ensure you implement an iterator (or Enumerable, for C# programmers). You may use generators if your language supports them, or you may implement the iterator classes manually if you choose to do so.