**Sample “Data Structures” Projects**

**Ocrober 2016**

**Reference:**

Robert Sedgewick and Kevin Wayne. 2011. Algorithms (4th ed.). Addison-Wesley Professional. <http://algs4.cs.princeton.edu/home/>

**3. Searching**

**Project # 1 3.2. BST**

**//Creative problems**

*3.2.29 Binary tree check.*

*3.2.30 Order check.*

*3.2.32 Certification.*

**[ETC…]**

//**Experiments**.

**3.3 Balanced Search Trees**

**Project #2. 2-3 search trees** [start from existing version in git or from the scratch]

**//creative problems**

*3.3.23 2-3 trees without balance restriction*

*3.3.32 AVL trees.*

*3.3.33 Certification*

*3.3.39 Delete the minimum*

**//Experiments**

*3.3.43 Cost plots.*

*3.3.44 Average search time.*

**4. Graphs**

**Project #3. Undirected Graphs**

**//creative problems**

*4.1.32 Parallel edge detection.*

**4.1.35** *Biconnectedness.*

**4.1.36** *Edge connectivity*

*// experiments*

**4.1.39** *Random graphs.*

**4.1.40** *Random simple graphs.*

**4.1.41** *Random sparse graphs.*

**4.1.45** *Random interval graphs.*

**4.1.47** *Path lengths in DFS.*

**Project #4. Directed Graphs**

**//creative problems**

**4.2.21** *LCA of a DAG.*

**4.2.22** *Shortest ancestral path.*

**4.2.23** *Strong component.*

**//experiments**

**4.2.32** *Random digraphs.*

**4.2.33** *Random simple digraphs.*

**4.2.34** *Random sparse digraphs.*

**4.2.40** *Path lengths in DFS*

**4.2.42** *Strong components.*

**Project #5. Minnimum Spanning Trees**

**Project #6. Shortest Paths**

**5. Strings**

**Project #7. String Sorts**

**Project #8. TRIEs**

**Project #9. Substring Search**