|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Week** | **Overall** | **Docs** | **Struct** | **Names** | **Tests** | **Funct** |
| **1 & 2**  **Search timer** | B | C | B | B | C | B |
| **3 & 4**  **Generic swap** | B | B | B | B | C | B |
| **5**  **Sorting** | B+ | A | B | B | A | B |
| **6**  **Linked lists** | B+ | B | B | A | B | B |
| **8**  **Binary trees** | B | B | B | B | C | B |

Evidence/justification:

Week 1 & 2:

The CleverRandomListing performs the task required, the code is commented step by step so another user can see how it works which is shown in the documentation. The structure is easy to read with a simple to understand naming convention, I tested it’s functionality with the timer multiple times and it works as desired. I feel the documentation can look busy and maybe could have done with some of the test results from the test class.

Week 3 & 4:

The Swap class swaps two objects in an array as required, it is generic as it swaps multiple types of objects, I demonstrated ints and strings swapping. I could have done a testing class rather than just a main but it does demonstrate it on a basic level. There are comments in the code to explain what’s happening and I feel the naming was simple.

Week 5:

I have shown many results from the low sizes to high, each sheet having a description of what’s happening. There are formulae to give an indication to how execution times vary to data sizes, however I feel the Quick Sorts algorithm is more original and accurate than the selection sorts.

Week 6:

The names of what the functions do a named clearly with comments, it performs the task which is demonstrated in the test methods, one shows a test with a large array the other shows a test with a small array. These have been shown with the results printed.

Week 8:

It performs the task as needed, could have made a detailed test class, however I have made a mina that shows an example of it working. Could have done a larger test sample. Have included comment, naming is generic and easy to understand.