Data gotten from the Specification Document:

SLL: 300 videos | 20 customers | 100 transactions

SLL took 0.003 seconds

DLL: 300 videos | 20 customers | 100 transactions

DLL took 0.004 seconds

BST: 300 videos | 20 customers | 100 transactions

BST took 0.002 seconds

AVL Tree 300 videos | 20 customers | 100 transactions

AVL Tree took 0.0 seconds

From these 4 data structures the order of slowest to fastest with this testing data is as follows: AVL Tree, Doubly Linked List, Singly Linked List, and finally the Binary Search Tree. The obvious thing about this comparison is that the AVL tree is a lot worse than the other three as AVL tree has many other operations to perform such as the balancing, the rotations and all those things. They all have the same theoretical time complexity of O(logn) but the additional functions of the AVL do impact the overall time complexity a little bit. The other three are relatively the same with only slight variations between them with the BST being the fastest. When they are all searched, they are effectively the same as a linked list however since the DLL has to make more changes overall like the pointers for previous and next compared to just next in an SLL and the header and trailer nodes, this can be the reason it takes longer to search than the others.