OS Assignment #1 Documentation

The general structure of the program is, as the assignment requires, in the form of a B+ Tree. The keys in the B+ Tree are student IDs, ordered alphanumerically. The B+ tree is right biased, with an order k, which is decided up by the user when they run the program. As each key represents a single student, at the leaf node level, each student has a corresponding singly linked list. Like the content of the B+ tree, the content of the singly linked list is dynamically allocated during runtime, and can be expanded or redacted during runtime.

The keys, pointers, and singly linked lists are all represented as members of the Node class, furthermore, they are all in the form of dynamic arrays. The keys are dynamic arrays of type int; the pointers are dynamic arrays of pointers to Nodes; and the singly linked lists are dynamic arrays of pointers to CourseNodes, the structures which make up the singly linked lists.

Since the structure of B+ trees allow for quick look-ups, the functions in this program which search for single targets utilize this characteristic by searching from the root, and traversing downwards using the pointers which help lead to the target faster. For example, the function which displays information of a single student, and the function which calculates the GPA of a single student both apply this approach. Functions which search for multiple targets, on the other hand, utilize the characteristic of the linked leaf nodes, and traverse through the leaf nodes to obtain targeted information.

Use of external resources: the functions “insertKey”, “insertInternal” and “findParent” of class “BPTree” contain code from geeksforgeeks.org (https://www.geeksforgeeks.org/insertion-in-a-b-tree/).