In project three, most of the design patterns we used were behavioral. Since our program was mainly function-oriented and had no classes, this eliminated a lot of the other design patterns. The other sections of design patterns involved strategies that require or are mainly used for object and class-oriented programs. However, there were some patterns that we used to implement our design. They helped because of how our program represents different data structures and represents class objects. One main design pattern we used was a chain of responsibility. This is because of the way the program progresses from page to page and how the data structures are displayed. The project is set up in such a way that when one option is selected, it leads to another page that can either lead to another page or a command that changes what is displayed. That chain of commands essentially makes the program run. Another design pattern we used was a memento, which is the pop method. When a value is pushed, it adds a value, which can be reverted with pop so that the rest of the structure before is not affected. This will be more evident once the prototype is completed and the linked list is implemented. An element can be selected and everything ahead of it can be removed without affecting the elements before it. Another pattern we have selected to implement will be an iterator for the binary search tree. This will allow elements to be placed in the tree