1. Complete Implementation of StacK
   1. static and dynamic implementation (use switch case)
2. Complete Implementation Of Queue
   1. Static and Dynamic Implementation (use switch case)
3. Singly Linked list implementation of stack and queue
4. Circular Linked list implementation of stack and queue
5. Implementation of primitive operation of Doubly linked list
6. Implementation of primitive operation of Singly linked list
7. Implementation of primitive operation of Circular linked list
8. Application to develop calculator using postfix evaluation
9. Develop Data Compression Tool Using Huffman Algorithm.
10. Menu driven sorting using different technique
11. Menu driven searching using different technique

# **Purpose**

The purpose of the projects is to help you learn the course material and to help you begin to implement your own personal library of tools. Many of the subsequent projects will rely on previous ones (for example, you may be asked to specifically use your linked list classes to implement more complex data structures).

# **Program Documentation and Style**

The following programming style is required for all projects.

Your name and ID must appear at the top of all files which you have created or modified.

Write clear and understandable code. Improve the clarity of your code by using vertical and horizontal spacing, meaningful variable names, proper indentation and comments.

Precede each function with comments indicating:

* What it does
* What each parameter is used for
* Assumptions that it makes
* How it handles errors