It is extremely important that you understand the programming assignments; please participate in the classroom discussion. You are expected to solve the programming assignments individually. You are encouraged to discuss with your friends. However, **do not copy**. Follow the submission and grading policies for full points.

Linked List, Queue, and BST in C

The primary focus is to implementation *Linked List, Queue, and BST* in C.

Q1. (13+6+6 = 25 points) Write code in C for inserting a key in a sorted singly linked list (SSLL) with adequate comments. Write a driver/main function to insert at least any 100000 positive numbers. Also, write a print function for the SSLL. Use the *struct* and function prototype. You may make any reasonable assumptions.

```
struct linked_list
{
  int key;
  struct linked_list *next;
};
typedef struct linked_list node;
node *insertSSLL(node *head, int x);
node *printSSLL(node *head);
```

Q2. (15+15+10+10 = 50 points) Deque (double ended queue) is a representation for a list where insertions and deletions can be made at either end. Write code in C for inserting and deleting at either end. Write driver/main to insert: 55 (head), 99 (tail), 77 (tail), and 33 (head) and to delete: tail and head. Also, write a print function for the Deque. Use the *struct* and function prototypes. You may make any reasonable assumptions.

```
struct Deque
{
  int key;
  struct Deque *leftt;
  struct Deque *right;
}*head, *tail;
void AddDQ(struct Deque *p, int x);
void DeleteDQ(struct Deque *p);
void PrintDQ(struct Deque *p);
```

Q3. (15+10 = 25 points) Write code in C to create a BST for any collection of seven positive numbers (example: 25, 16, 56, 3, 69, 85, and 68). Print the BST in Inorder, Preorder, and Postorder. Make all reasonable assumptions.

(Optional) Q4. (15 points) Re-do Q3 to create an AVL tree. Re-use Q3 Print functions.