1. (This complete question is worth 30 marks)

(i) Explain the implementation of a *Doubly-Linked List* using Python classes and objects, giving a clear specification for each class, object and variable. You do not need to specify any methods. Include in your explanation an illustration of how the sequence of objects <x,y,z> would be represented.

(7 ***marks)***

(ii) Give clear pseudocode for the operation of swapping two neighbouring nodes in a doubly linked list, using the implementation you specified in 1(i) above. (Note: your pseudocode must swap the nodes, and not simply swap the data)

(4 marks)

(iii) State the defining characteristics of a *Stack,* and state the standard operations offered in the Stack ADT. For each operation, give a brief comment explaining what it does.

(5 marks)

(iv) Explain, using text or sketches of the layout in memory or pseudocode or Python code, how we can use a doubly-linked list to give an efficient implementation of a Stack. State the complexity of each operation.

(6 marks)

(v) A *MinStack* is an ADT which extends the Stack ADT with one additional method, *min(),* which simply reports the minimum valued element in the stack. Give a design for an efficient implementation of MinStack — ideally, this implementation should allow constant time operation (i.e. 0(1)) for min(), and should not change the complexity of any of the other methods compared to original Stack.

(8 marks)