Algorithms and Data Types

Assignment 2 Logbbook

05/19

Stefan Ahmed

21359035

SW 10 - Sorting

1. Integers:

[8,7,6,5,4,3,2,1]

Bubble sort

Step 1

8 7 6 5 4 3 2 1

7 8 6 5 4 3 2 1

7 6 8 5 4 3 2 1

7 6 5 8 4 3 2 1

7 6 5 4 8 3 2 1

7 6 5 4 3 8 2 1

7 6 5 4 3 2 8 1

7 6 5 4 3 2 1 8

Step 2

7 6 5 4 3 2 1 8

6 7 5 4 3 2 1 8

6 5 7 4 3 2 1 8

6 5 4 7 3 2 1 8

6 5 4 3 7 2 1 8

6 5 4 3 2 7 1 8

6 5 4 3 2 1 7 8

Step 3

6 5 4 3 2 1 7 8

5 6 4 3 2 1 7 8

5 4 6 3 2 1 7 8

5 4 3 6 2 1 7 8

5 4 3 2 6 1 7 8

5 4 3 2 1 6 7 8

Step 4

5 4 3 2 1 6 7 8

4 5 3 2 1 6 7 8

4 3 5 2 1 6 7 8

4 3 2 5 1 6 7 8

4 3 2 1 5 6 7 8

Step 5

4 3 2 1 5 6 7 8

3 4 2 1 5 6 7 8

3 2 4 1 5 6 7 8

3 2 1 4 5 6 7 8

Step 6

3 2 1 4 5 6 7 8

2 3 1 4 5 6 7 8

2 1 3 4 5 6 7 8

Step

2 1 3 4 5 6 7 8

1 2 3 4 5 6 7 8

Selection Sort

[8,7,6,5,4,3,2,1]

8 7 6 5 4 3 2 1

1 7 6 5 4 3 2 8 Pass 1 - 7 comp, 1 swap

1 2 6 5 4 3 7 8 Pass 2 – 6 comp, 1 swap

1 2 3 5 4 6 7 8 Pass 3 – 5 comp, 1 swap

1 2 3 4 5 6 7 8 Pass 4 – 4 comp, 1 swap

Insertion Sort

[8,7,6,5,4,3,2,1] List to be sorted

[7,8,6,5,4,3,2,1] 1 shift

[6,7,8,5,4,3,2,1] 1 shift

[5,6,7,8,4,3,2,1] 1 shift

[4,5,6,7,8,3,2,1] 1 shift

[3,4,5,6,7,8,2,1] 1 shift

[2,3,4,5,6,7,8,1] 1 shift

[1,2,3,4,5,6,7,8] 1 shift

Week 11 – Sorting

1.

[8,7,6,5,4,3,2,1]

Shell sort

8 7 6 5 4 3 2 1

8 5 2 Sublist 1

7 4 1 Sublist 2

6 3 Sublist 3

2 5 8 Sublist 1 sorted

1 4 7 Sublist 2 sorted

3 6 Sublist 3 sorted

Apply insertion sort on the list:

2 1 3 5 4 6 7 8

1 2 3 5 4 6 7 8

1 2 3 4 5 6 7 8 List is sorted

Merge sort

8 7 6 5 4 3 2 1

8 7 6 5 4 3 2 1 List to be sorted

8 7 6 5 4 3 2 1 Split the list into half

8 7 6 5 4 3 2 1 Split the list into half again

8 7 6 5 4 3 2 1 Split the pairs of numbers

7 8 5 6 3 4 1 2 Merge the numbers

5 6 7 8 1 2 3 4 Continue merging

1 2 3 4 5 6 7 8 Merge the two halves of the list back together

1 2 3 4 5 6 7 8 List is sorted

Week 7 – Queues

1. What values are returned during the following sequence of queue operations, if executed on an initially empty queue?

```
Q = Queue()
Q.enqueue(5)
Q.enqueue(3)
Q.dequeue()
Q.enqueue(2)
Q.enqueue(8)
Q.dequeue()
Q.dequeue()
Q.enqueue(9)
Q.enqueue(1)
Q.dequeue()
Q.enqueue(7)
Q.enqueue(6)
Q.dequeue()
Q.dequeue()
Q.enqueue(4)
Q.dequeue()
Q.dequeue()
```

The values returned are:

5
3
2

8
9
1
7
6

2. Given that the Stack and Queue classes have been implemented correctly, what is the output of the following?

```
s1 = Stack()
q = Queue()
s1.push(1)
s1.push(2)
s1.push(3)
while not s1.isEmpty():
    (q.enqueue(s1.pop()))
print(q.dequeue(), end = ' ')
print(q.dequeue(), end = ' ')
print(q.dequeue(), end = ' ')
```

The output is

3
2
1

Q = Queue [ ]

[5,3]
[3] 5

3 2 1

Week 8 – Linked Lists

1. What is the output of the following program?

```
def TestUnorderedList():
    my_list = UnorderedList()
    number_list = [11, 17, 7, 3, 26, 54, 2]
    for num in number_list:
        my_list.add(num)
    print (my_list.size())
    print (my_list.search(17))
    print (my_list.search(1))
    my_list.remove(2)
    my_list.remove(54)
    print (my_list.size())
```

The outputs are

7

True

False

5

2. Give an algorithm for finding the second-to-last node in a non-empty singly linked list in which the last node is indicated by a next reference of None.

```
curr = self.head()
next = curr.get_next()

While curr.get_next != None
curr = curr.get_next()
next = curr.get_next()

Return curr
```

3. Describe a recursive algorithm that counts the number of nodes in a singly linked list

```
  def Size()
  curr == self.head()
  count = 0
 While curr != none
   count = count+1
  curr = curr.get_next
Return count
```

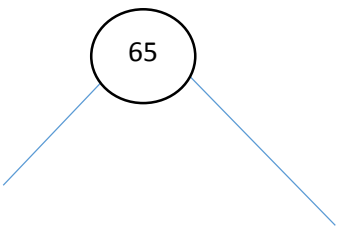
Week 9 – Linked Lists

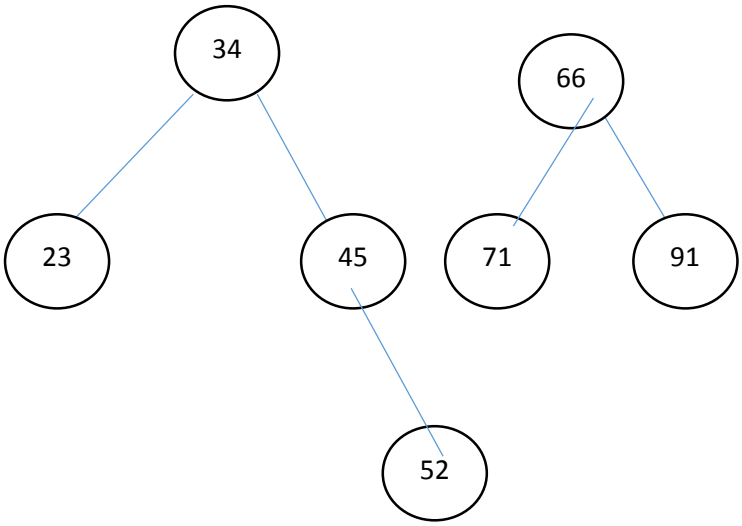
1.

def

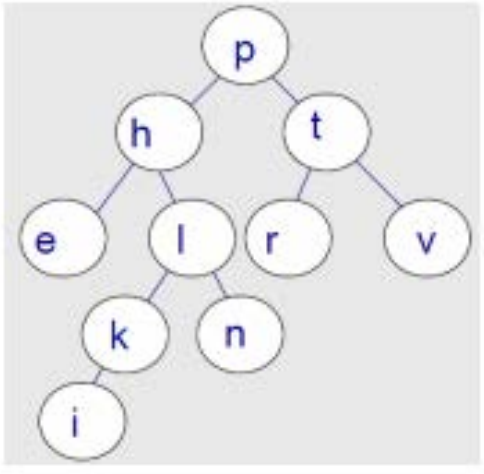
1. Create a binary search tree by adding the following values in the order given:

65 34 66 91 23 45 71 52





2. The following diagram shows a binary tree with the root node containing the value, p. Write the pre-order, in-order and post-order traversals of the following binary tree.



Pre-order

1

2 8

3 4 9 10

5 7

6

Post-order

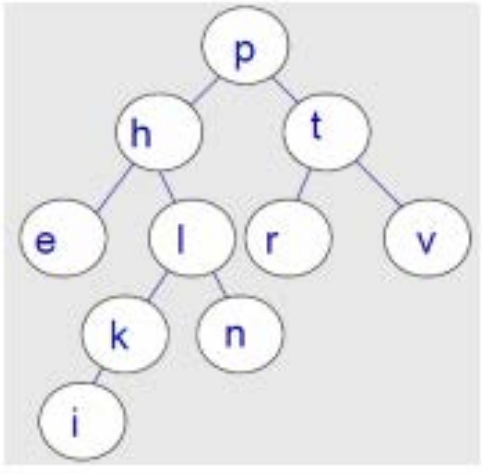
10

6 9

5 4 7 8

2 3

1



Inorder

7

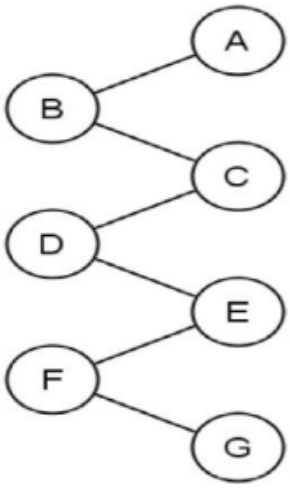
2 9

1 5 8 10

4 6

3

3. The following diagram shows a binary tree with the root node containing the value, A. Write the pre-order, in-order and post-order traversals of the following binary tree.



Preorder traversal

1

2

3

4

5

6

7

Postorder Traversal

7

6

5

4

3

2

1

5.Draw the binary search tree structure after inserting the following integer search key values into an empty binary search tree in the order given: 7, 3, 1, 6, 5, 10, 8, 9