Tut - 8

U20CS110
Krishna Pandey

Sol-1) // Traversing a doubly linked list

```
Start hold the address of 1st node.
(i) Set node = start (intialize pointers variable)
(ii) Repeat step 3 and 4 while ≠ NULL
(iii) Process info [node] (scanning the values)
(iv) Set node = next [node]
(v) Exist.
```

// Insertion at the begining

```
Start ~~the~~ ~~node~~ holds the address of 1st Node.
(i) Create a node named as node
(ii) If node == NULL then "out of memory space" Exist.
(iii) Set info [NODE) = X (new data item)
(iv) Set info (node) = start (new node points to original 1st node)
(v) set Prev [node ] = NULL
(vi) Set Prev [start] = node
(vii) Set start = node
(viii) Exist.
```

// Insertion at the end

```
Start holds the address of 1st node
(i) Create a node named as node.
(ii) If node == NULL then "Out of memory space" Exist.
(iii) Set info[node] = X
(iv) Set next [node] = NULL.
(v) set curr = start
vi) repeat step (vii) and (viii) while curr ≠ NULL
vii) Set pre = curr
viii) Set curr = next (curr) (end of loop)
X) set next [Pre] = node
x) set Prev (node) = Pre.
xi) Exit
```

// Insertion at specific location

```
Start hold the address of 1st node
1) Create a new node named as node
2) If node == NULL then write "out of space" Exit.
3) set info [node] = X
4) set next [node] = NULL
5) set curr = start
6) Read LOC (3)
7) set i = 1
8) Repeat 9 to 10 while curr ≠ NULL and i < loc
9) Set pre = curr
10) Set curr = next [curr]
11) set i = i+1
12) If curr == NULL "Position not found"
13) Set next [Pre] = node
14) set prev [node] = pre
15) set next [node] = curr
16) set prev [curr] = node.
17) Exit
```

// Deletion from begining

```
Start holds the address of 1st node
1) Set temp = start
2) If start == NULL 'Underflow'
3) start = next [start]
4) set prev [start] = NULL
5) temp space free
6) Exit.
```

```
// Deletion at the end
start = address of the Ist node
1) set node = start
2) set temp = start
3) If node == NULL , 'underflow' Exit
4) Repeat (5) and (6) while next [node] ≠ NULL
5) Set temp = node
6) Set node = next [node]
7) Set next [temp] = NULL
8) Free 'node')
9) Exit
```

```
// Deletion of a node after specified location.
  start = address of the Ist node.

1) Set node = start
2) If node == NULL 'Underflow' and exist.
3) Read item.
4) If info [node] = item and next [node] = NULL
     then 'invalid Deletion' and exit.
5) Repeat step 6 and 7 while node ≠ NULL and
     item ≠ info [node]
6) Set temp = node.
7) Set node = next [node)
8) If node == NULL or next [node) == NULL then
     'Invalid deletion'.
9) Set next [temp] = next [node]
10) set prev [next [node] ] = temp.
11) Free the space associated with node.
12) Exist.
```

```
// Search for an element
(i) If  head == NULL
        write 'underflow' and exist.
2) Set  ptr = head
3) set  i = 0
4) repeat step 5 to 7  while  ptr != NULL
5) If  Ptr → data = item
      return i
      (End of if)
6) i = i+1
7) ptr = ptr → next
8) Exit.
```