LAB-05 Singly Linked List

Name: **Kashif Ali**
Roll NO: **20P-0648**
Section: **3D**

Lab-05 code

Task # 1: Implement a SinglyLinkedList class

linkedList.cpp

home > kashiii > Downloads > 20k1887 Muzamil Lab 3 > linkedList.cpp > Node > Node()

```cpp
#include<iostream>
using namespace std;

class Node {
    public:
        int data;
        Node *next;

    Node(const int data)
    {
        this->data = data;
        this->next = NULL;
    }

    Node()
    {
        this->data = 0;
        this->next = NULL;
    }
};

class linkedList {

    public:
        Node *head;
        Node *tail;


    linkedList() {
        this->head = NULL;
        // this->tail = NULL;
    }
```

Task # 2: Add a node at the end of a Singly Linked List.

```
74  void linkedList::pussh(int n) {
75      Node *temp = new Node;
76      temp->data = n;
77      temp->next = NULL;
78
79      if(head == NULL)
80      {
81          head = temp;
82          // tail = temp;/
83      }
84      else
85      {
86          Node *t = head;
87          while(t->next!=NULL)
88          {
89              t = t->next;
90          }
91          t->next = temp;
92
93      }
94  }
95
```

Task # 3: Add a node at the front of a Singly Linked List

```
66  void linkedList::front(int n) {
67      Node *temp = new Node;
68      temp->data = n;
69      temp->next = head;
70      // tail = temp;
71      head = temp;
72  }
```

Task # 4: Add a node after a given node in a Singly Linked List

```
 96  void linkedList::insertIndex(int in, int v)
 97  {
 98      Node *temp = new Node;
 99      temp->data = v;
100
101      if(head == NULL)
102      {
103          head = temp;
104      }
105      else
106      {
107          Node *p = head;
108          for (int i = 1; i <in-1 ; i++)
109          {
110              p = p->next;
111          }
112          temp->next = p->next;
113          p->next = temp;
114          // cout<<p->data<<endl;
115
116      }
```

Task # 5Delete a node from a Singly Linked List
- **Delete Last node** - **Delete any other node**

```
170  void linkedList::deleteEnd()
171  {
172      Node *temp = head;
173      Node *pre = head;
174      while(temp->next!=NULL)
175      {
176          pre = temp;
177          temp = temp->next;
178      }
179      delete temp;
180      pre->next = NULL;
181  }
182
```

Delete last Node

Task # 5Delete a node from a Singly Linked List
- Delete Last node - Delete any other node

```
183  void linkedList::deleteAtIndex(int in)
184  {
185      Node *p = head, *n = head;
186      for (int i = 1; i < in; i++)
187      {
188          p = n;
189          n = n->next;
190      }
191
192      p->next = n->next;
193      delete n;
194
195  }
196
```

Delete any Node

Task # 6
Update a node in a Singly Linked List

```
 96  void linkedList::updateList(int in, int v)
 97  {
 98      Node *temp = new Node;
 99      temp->data = v;
100
101      if(head == NULL)
102      {
103          head = temp;
104      }
105      else
106      {
107          Node *p = head;
108          for (int i = 1; i <in-1 ; i++)
109          {
110              p = p->next;
111          }
112          temp->next = p->next;
113          p->next = temp;
114          // cout<<p->data<<endl;
115
116      }
117  }
118
```

Task # 7

linkedList.cpp

home > kashiii > Downloads > 20k1887 Muzamil Lab 3 > linkedList.cpp > EvenFront(Node **)

```
299  using namespace std;
300
301  /* a node of the singly linked list */
302  class Node
303  {
304      public:
305      int data;
306      Node *next;
307  };
308
309  void EvenFront(Node **head_ref)
310  {
311      Node *end = *head_ref;
312      Node *prev = NULL;
313      Node *curr = *head_ref;
314
315      while (end->next != NULL)
316          end = end->next;
317
318      Node *new_end = end;
319
320      while (curr->data % 2 != 0 && curr != end)
321      {
322          new_end->next = curr;
323          curr = curr->next;
324          new_end->next->next = NULL;
325          new_end = new_end->next;
326      }
327
328      if (curr->data%2 == 0)
329      {
330
```

Code Part-1

Task # 7

linkedList.cpp

home > kashiii > Downloads > 20k1887 Muzamil Lab 3 > linkedList.cpp > EvenFront(Node **)

```
327
328     if (curr->data%2 == 0)
329     {
330
331         *head_ref = curr;
332
333
334         while (curr != end)
335         {
336             if ( (curr->data) % 2 == 0 )
337             {
338                 prev = curr;
339                 curr = curr->next;
340             }
341             else
342             {
343                 prev->next = curr->next;
344
345                 curr->next = NULL;
346
347                 new_end->next = curr;
348                 new_end = curr;
349
350                 curr = prev->next;
351             }
352         }
353     }
354
355     else prev = curr;
356
357     if (new_end != end && (end->data) % 2 != 0)
358     {
359         prev->next = end->next;
360         end->next = NULL;
361         new_end->next = end;
362     }
363     return;
364 }
```

Code Part-2

The End
Thank You
...