

11/10/2021

**Department Of Computer Science**

**Subject** **Instructor:** Ma’am Zainab Malik

**Assignment:** 2 **Date:** 10-11-2021

**Class:** BSCS-3B

**Submitted by:**

Madina Javed Iqbal Khokhar 2426

**Task**

**Assigned Task:**

Implement the following functions of two-way linked list, on the basis of discussed algorithm, using

1. Write a function to remove element from head
2. Write a function to remove element from tail
3. Write a function to remove all element from the list
4. Write a function to remove specific element from the list
5. Write a function to replace specific element with the provided element

**Algorithms:**

* **Write a function to remove element from head:**

**Case 1: List is empty**if(head==0) //list is empty

{

cerr<<"List is empty therefore, so, nothing to remove"<<endl;

}

**Case 2: List have Only One element**

if(Info = head->getInfo())   
 delete head  
 head = 0  
 tail = 0

Return info

**Case 2: List have more than One element**

set\*Temp = head

Head = head -> getNext()

If(Info = temp->getInfo())   
 delete temp

Return info

* **Write a function to remove element from tail**

**Case 1: List is empty**if(head==0)//list is empty

{

cerr<<"List is empty therefore nothing to remove"<<endl;

}

**Case 2: List have Only One element**

If(Info = tail->getInfo())   
 delete tail  
 head = 0  
 tail = 0

Return info

**Case 2: List have more than One element**

set\*Temp =tail

tail = tail -> getprev()

if(Info = temp->getInfo())   
 delete temp

Return info

* **Write a function to remove all element from the list:**

**Case 1: List is empty**if(head==0) //list is empty

{

cerr<<"List is empty therefore, so, nothing to remove"<<endl;

}  
**Case 2: List have One or more than one elements**

set \*temp=head

While( temp!= 0 )  
  
{

Temp=temp->next

Removefromhead()

}

* **Write a function to remove specific element from the list**

**Case 1: List is empty**if(head==0) //list is empty

{

cerr<<"List is empty therefore, so, nothing to remove"<<endl;

}

**Case 2: When Head = tail and head = element**

Delete head  
 head =0

Tail=0

**Case 3: When head = element** removeFromHead () **Case 4: When tail= element** removeFromTail ()

**Case 5: When element is in between head and tail** set \*temp=head

While (temp! = tail && temp->getNext ()->getInfo ()! = element)

{

temp=temp->getNext()

}

if(temp==tail)

{

cerr<<"Element not found"

}

else

{

set\*ptr=temp->getNext()

temp->setNext(ptr->getNext())

ptr->getnext(setprev->temp)

delete ptr

}

* **Write a function to replace specific element with the provided element**

**Case 1: List is empty**if(head==0) //list is empty

{

cerr<<"List is empty therefore, so, nothing to replace"<<endl;

}

**Case 2: When Head = tail and head = element**

Head->setinfo(element)

**Case 3: When head == element**  Head->setinfo(element) **Case 4: When tail== element** tail->setinfo(element)

**Case 5: When element is in between head and tail** set \*temp=head

Repeat step 3 While (temp != tail && temp->getNext()->getInfo() != element)

{

temp=temp->getNext()

}

if(temp==tail)

{

cerr<<"Element not found"

}

else

{

set \*ptr=temp->getNext()

Ptr->setinfo(element)

}