Solution

Data Structures and Algorithms

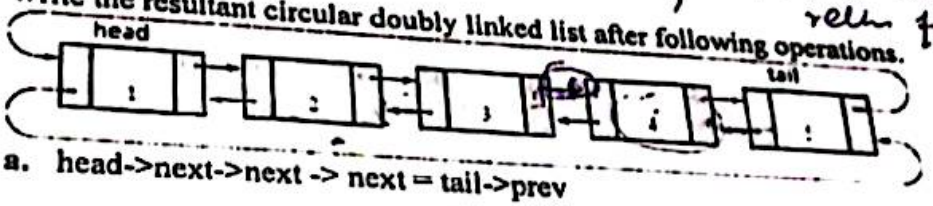
Quiz # 3(CLO3)

Name: **Rollno**

1. Write code for Stack ADT to check bracket balancing.

```
bool bracketBalancing(){
bool check_bracket( string str) {
int i= 0;
while ( i< str.length()){
if ( str[i] == '(' || str[i] == '{' || str[i] == '[' ) {
    s1.push(str[i]);
}
if( str[i]= ')' && s1.top== '(' || '}' || ']' ) {
    s1.pop();
}
```

2. Write the resultant circular doubly linked list after following operations.



a. head->next->next -> next = tail->prev

unchange

b. tail->prev->prev->prev = head ->next

unchange

3. You are given a pointer that points to a node in a linked list. Write a C++ program to find if the linked list is circular or not.

```
void check ( ) {
    node * temp2= he.d;
while (temp->nul!=NULL) {
        temp2=temp2->next;
}
if( temp2->next==NULL && he.d->prev==NULL) {
    "circular";
} else {not} }
```

Data Structures and Algorithms

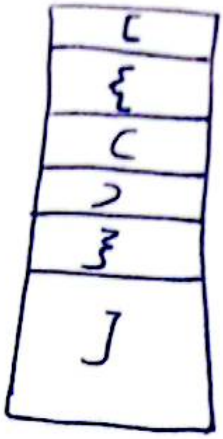
Quiz # 3(CLO3)

Name: Mahrukh | **Roll#** F2023408122

1. Write code for Stack ADT to check bracket balancing.

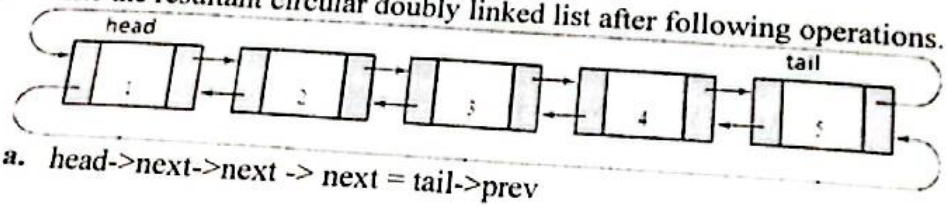
```
bool BracketBalancing(){
```

[{()}]



Stack of Balanced bracket.

2. Write the resultant circular doubly linked list after following operations.



a. head->next->next -> next = tail->prev

4

b. tail->prev->prev->prev = head ->next

2

2

3. You are given a pointer that points to a node in a linked list. Write a C++ program to find if the linked list is circular or not.

```
class Node{
  int data;
  node *ptr;
node (int val){
  data = val;
  ptr = Null;
} };
class Link list{
node *head = new node();
}
```

2

```
void circularcheck(){
node * tail;
tail * &ptr;
if( ~~tail~~ tail* ptr == Head
cout << "Circular"; }
else {
cout<<
```

2

Data Structures and Algorithms

Quiz # 3(CLO3)

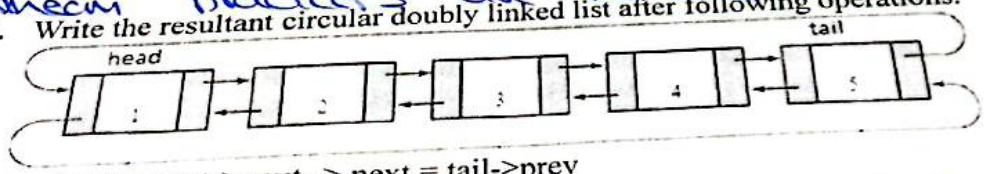
Name: Muhammad Talha Khan	**Roll#** [illegible]

1. Write code for Stack ADT to check bracket balancing

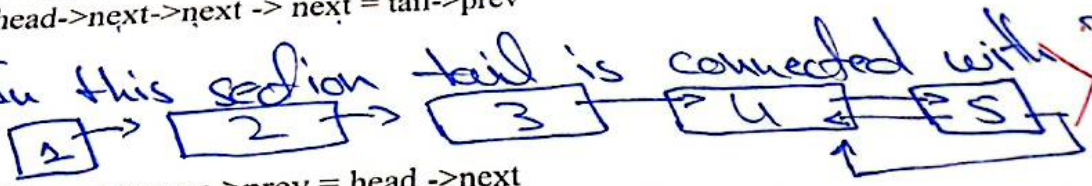
```
bool BracketBalancing(){
```

Bracket balancing is a method when 2 the opening bracket it will be going put in the stack and when the closing bracket came it will pop up. If no bracket found at the end of program in each it means brackets are balance and other side vice.

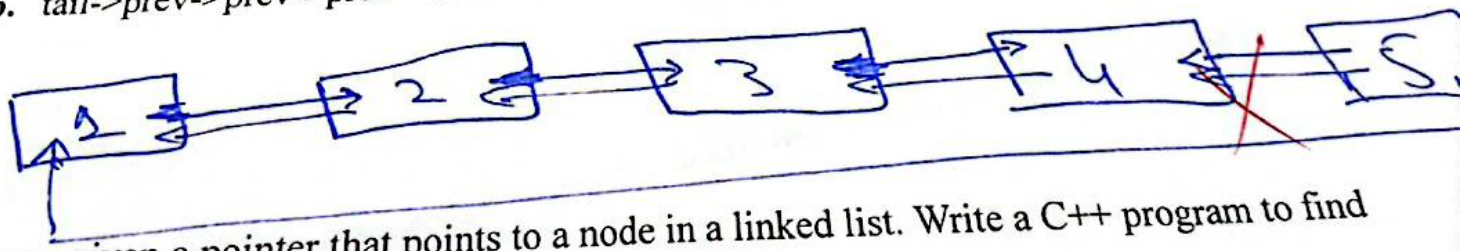
2. Write the resultant circular doubly linked list after following operations.



a. head->next->next -> next = tail->prev



b. tail->prev->prev->prev = head ->next



3. You are given a pointer that points to a node in a linked list. Write a C++ program to find if the linked list is circular or not.

We can check through if the pointer move further to the other indexes.

Data Structures and Algorithms

Quiz # 3(CLO3)

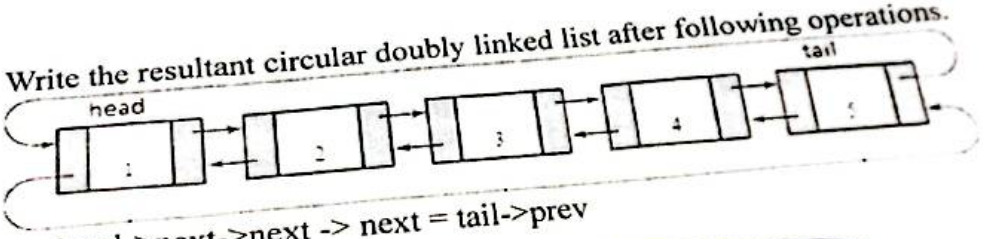
Name: Usman Ali; **Roll#** F2023MCS215

1. Write code for Stack ADT to check bracket balancing.

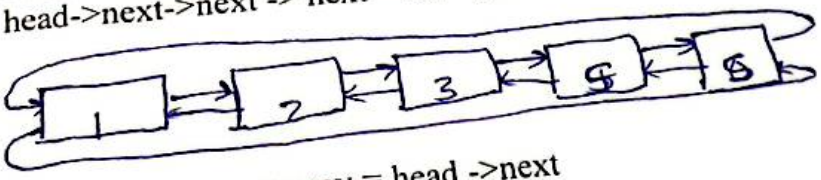
```
bool BracketBalancing(){
```

On stack ADT to check the bracket balancing we can use the bool Bracket balancing which further tells us about the bracket balancing.

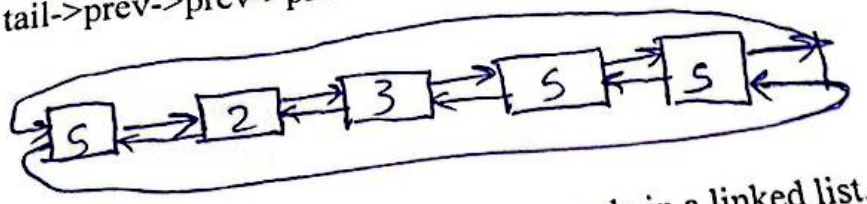
2. Write the resultant circular doubly linked list after following operations.



a. head->next->next -> next = tail->prev



b. tail->prev->prev->prev = head ->next



3. You are given a pointer that points to a node in a linked list. Write a C++ program to find if the linked list is circular or not.

To find the if the linked list is circular or not can use the function circular can tell us if the ~~function~~ linked list is or not.