

# **DSA -LAB**

# **CONTINUOUS**

# **ASSESSMENT**

*HARSH SHAH JAISWAL  
20BCE10556*

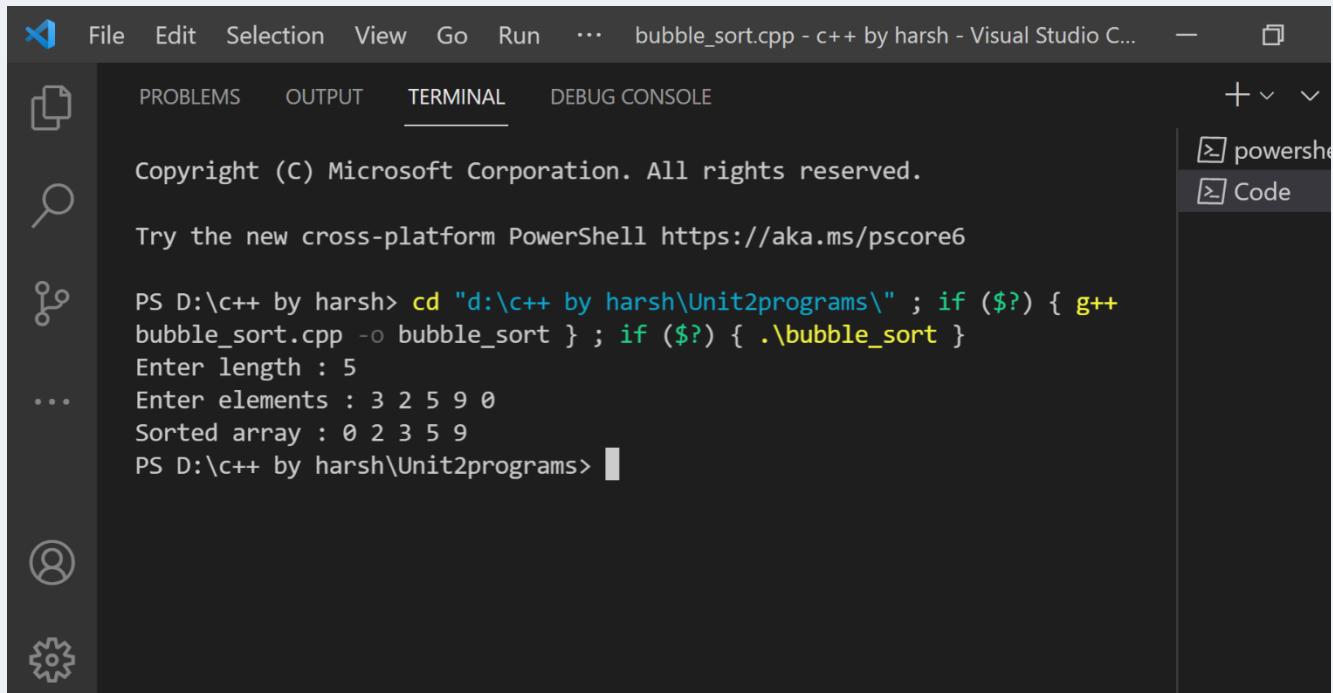
*SLOT – C11+DC1*

# SORTING

## BUBBLE SORT

```
#include <iostream>
using namespace std;
int main() {
    int n;
    cout<<"Enter length : ";
    cin>>n;
    int arr[n];
    cout<<"Enter elements : ";
    for(int i=0;i<n;i++){
        cin>>arr[i];
    }
    for(int i=0;i<n;i++){
        for(int j=0;j<n-i;j++){
            if(arr[j]>arr[j+1]){
                int temp=arr[j];
                arr[j]=arr[j+1];
                arr[j+1]=temp;
            }
        }
    }
    cout<<"Sorted array : ";
    for(int i=0;i<n;i++){
        cout<<arr[i]<<" ";
    }
    return 0;
}
```

## OUTPUT



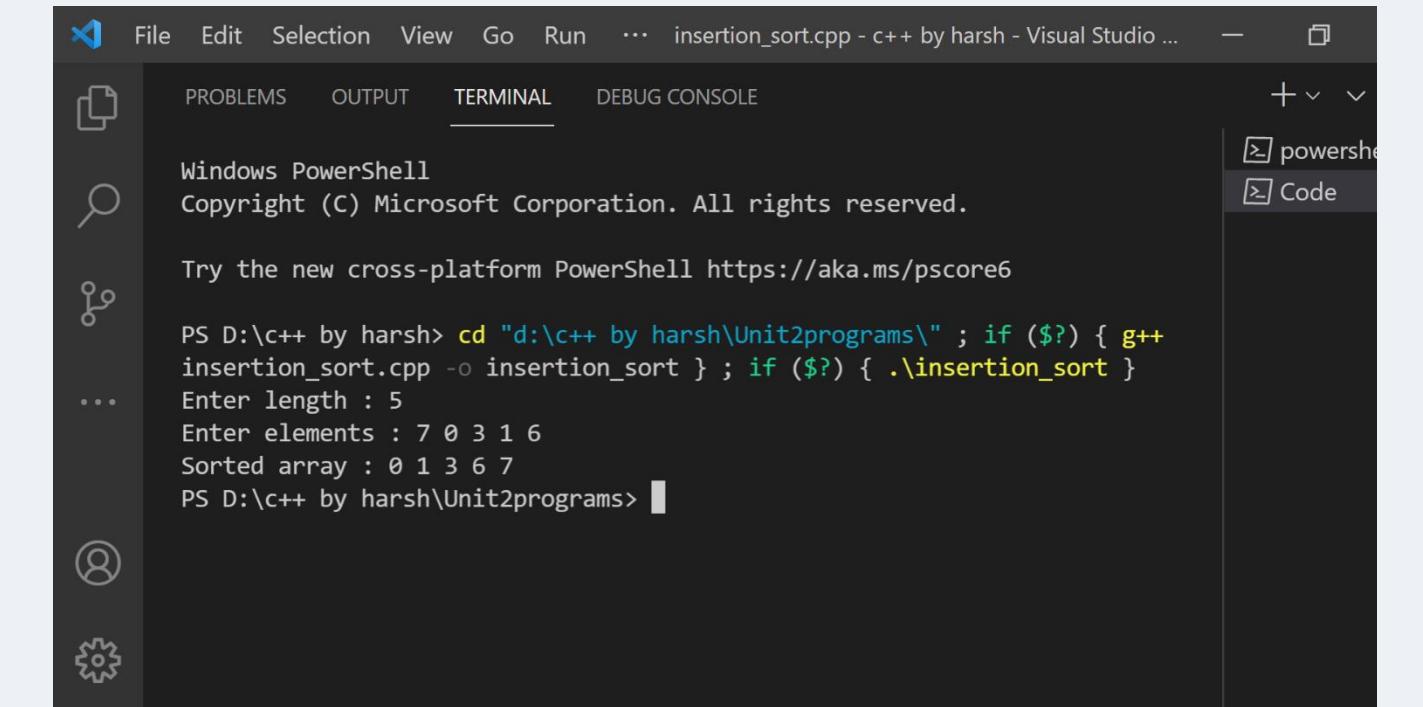
```
File Edit Selection View Go Run ... bubble_sort.cpp - c++ by harsh - Visual Studio C... — □ PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE + v v Copyright (C) Microsoft Corporation. All rights reserved. Try the new cross-platform PowerShell https://aka.ms/pscore6 PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ bubble_sort.cpp -o bubble_sort } ; if ($?) { .\bubble_sort } Enter length : 5 Enter elements : 3 2 5 9 0 Sorted array : 0 2 3 5 9 PS D:\c++ by harsh\Unit2programs>
```

## INSERTION SORT

```
#include<iostream>
using namespace std;
int main()
{
    int n;
    cout<<"Enter length : ";
    cin>>n;
    int arr[n];
    cout<<"Enter elements : ";
    for (int i=0;i<n;i++)
    {
        cin>>arr[i];
    }
    for (int i=1;i<n;i++)
    {
        int current=arr[i];
        int j=i-1;
        while(arr[j]>current && j>=0)
        {
            arr[j+1]=arr[j];
            j--;
        }
        arr[j+1]=current;
    }
}
```

```
        j--;
    }
    arr[j+1]=current;
}
cout<<"Sorted array : ";
for(int i=0;i<n;i++)
{
    cout<<arr[i]<<" ";
}
return 0;
}
```

## OUTPUT



The screenshot shows the Visual Studio Code interface with the 'TERMINAL' tab selected. The terminal window displays the execution of a C++ program named 'insertion\_sort'. The user navigates to the directory 'Unit2programs', compiles the program ('g++ insertion\_sort.cpp -o insertion\_sort'), and runs it ('.\insertion\_sort'). They then enter the length of the array (5) and the elements (7 0 3 1 6), followed by pressing Enter. The program outputs the sorted array (0 1 3 6 7).

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ insertion_sort.cpp -o insertion_sort } ; if ($?) { .\insertion_sort }
Enter length : 5
Enter elements : 7 0 3 1 6
Sorted array : 0 1 3 6 7
PS D:\c++ by harsh\Unit2programs>
```

# SELECTION SORT

```
#include<iostream>
using namespace std;
int main()
{
    int n;
    cout<<"Enter length : ";
    cin>>n;
    int arr[n];
    cout<<"Enter elements : ";
    for (int I = 0; i<n;i++)
    {
        cin>>arr[i];
    }
    for (int i=0;i<n-1;i++)
    {
        for (int j=I+1;j<n;j++)
        {
            if(arr[j]<arr[i])
            {
                int temp=arr[j];
                arr[j]=arr[i];
                arr[i]=temp;
            }
        }
    }
    cout<<"Sorted array : ";
    for (int i=0;i<n;i++)
    {
        cout<<arr[i]<<" ";
    }
    return 0;
}
```

# OUTPUT

```
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE + v v
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ selection_sort.cpp -o selection_sort } ; if ($?) { .\selection_sort }
Enter length : 5
Enter elements : 9 6 8 1 3
Sorted array : 1 3 6 8 9
PS D:\c++ by harsh\Unit2programs> []
```

# QUICK SORT

```
#include<iostream>
using namespace std;
void swap(int *a, int *b) {
    int temp;
    temp = *a;
    *a = *b;
    *b = temp;
}
int Partition(int a[], int l, int h) {
    int pivot, index, i;
    index = l;
    pivot = h;
    for(i = l; i < h; i++) {
        if(a[i] < a[pivot]) {
            swap(&a[i], &a[index]);
            index++;
        }
    }
    swap(&a[pivot], &a[index]);
    return index;
}
int RandomPivotPartition(int a[], int l, int h) {
    int pvt, n, temp;
```

```
n = rand();
pvt = l + n%(h-l+1);
swap(&a[h], &a[pvt]);
return Partition(a, l, h);
}
int QuickSort(int a[], int l, int h) {
    int pindex;
    if(l < h) {
        pindex = RandomPivotPartition(a, l, h);
        QuickSort(a, l, pindex-1);
        QuickSort(a, pindex+1, h);
    }
    return 0;
}
int main() {
    int n, i;
    cout<<"Enter length : ";
    cin>>n;
    int arr[n];
    cout<<"Enter elements : ";
    for(i = 0; i < n; i++) {
        cin>>arr[i];
    }
    QuickSort(arr, 0, n-1);
    cout<<"Sorted array : ";
    for (i = 0; i < n; i++)
        cout<<arr[i]<<" ";
    return 0;
}
```

# OUTPUT

The screenshot shows a Visual Studio Code interface. The title bar reads "File Edit Selection View Go Run ... quick\_sort.cpp - c++ by harsh - Visual Studio Co...". The left sidebar has icons for file, search, and settings. The top menu bar includes "PROBLEMS", "OUTPUT", "TERMINAL", and "DEBUG CONSOLE". The "TERMINAL" tab is selected, displaying a Windows PowerShell session. The output shows the execution of a C++ program named "quick\_sort.cpp". The program asks for an array length (5) and elements (1 4 2 3 8), then prints a sorted array (1 2 3 4 8). The status bar at the bottom shows "Ln 19, Col 33" and other system information like weather and battery level.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ quick_sort.cpp -o quick_sort } ; if ($?) { .\quick_sort }
Enter length : 5
Enter elements : 1 4 2 3 8
Sorted array : 1 2 3 4 8
PS D:\c++ by harsh\Unit2programs>
```

# MERGE SORT

```
#include <iostream>
using namespace std;
void Merge(int *a, int low, int high, int mid){
    int i, j, k, temp[high-low+1];
    i = low;
    k = 0;
    j = mid + 1;
    while (i <= mid && j <= high)
    {
        if (a[i] < a[j])
        {
            temp[k] = a[i];
            k++;
            i++;
        }
        else
        {
            temp[k] = a[j];
            k++;
            j++;
        }
    }
}
```

```

        while (i <= mid)
    {
        temp[k] = a[i];
        k++;
        i++;
    }
    while (j <= high)
    {
        temp[k] = a[j];
        k++;
        j++;
    }
    for (i = low; i <= high; i++)
    {
        a[i] = temp[i-low];
    }
}
void MergeSort(int *a, int low, int high)
{
    int mid;
    if (low < high)
    {
        mid=(low+high)/2;
        MergeSort(a, low, mid);
        MergeSort(a, mid+1, high);
        Merge(a, low, high, mid);
    }
}
int main()
{
    int n, i;
    cout<<"Enter length : ";
    cin>>n;
    int arr[n];
    cout<<"Enter elements : ";
    for(i = 0; i < n; i++)
    {
        cin>>arr[i];
    }
    MergeSort(arr, 0, n-1);
    cout<<"Sorted array : ";
    for (i = 0; i < n; i++)
        cout<<arr[i]<<" ";
    return 0;
}

```

# OUTPUT

The screenshot shows the Visual Studio Code interface with the 'OUTPUT' tab selected. The terminal window displays the following text:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ merge_sort.cpp -o merge_sort } ; if ($?) { .\merge_sort }
Enter length : 5
Enter elements : 9 7 8 1 4
Sorted array : 1 4 7 8 9
PS D:\c++ by harsh\Unit2programs> []
```

The status bar at the bottom shows: Ln 20, Col 17 Tab Size: 4 UTF-8 CRLF C++ Go Live Win32 203.

## SEARCHING

### BINARY SEARCH

```
#include<iostream>
using namespace std;
int main(){
    int n;
    cout<<"Enter length : ";
    cin>>n;
    int arr[n];
    cout<<"Enter elements : ";
    for (int i = 0; i < n; i++)
    {
        cin>>arr[i];
    }
    int s=0, e=n, key;
    cout<<"Enter value to find : ";
    cin>>key;
    cout<<"Element found at position : ";
    while (s<=e)
    {
        int mid=(s+e)/2;
```

```
if(arr[mid]==key){  
    cout<<mid;  
    break;  
}  
else if(arr[mid]>key){  
    e=mid-1;  
}  
else{  
    s=mid+1;  
}  
}  
return 0;  
}
```

## OUTPUT

The screenshot shows the Visual Studio Code interface with the terminal tab selected. The terminal window displays the execution of a C++ binary search program. The user navigates to the directory, compiles the code, and runs it. The program prompts for the length of the array, elements, and the value to find, then outputs the position of the found element.

```
File Edit Selection View Go Run ... Binary_search.cpp - c++ by harsh - Visual Studio ... — □  
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE + ▾ ▾  
Windows PowerShell  
Copyright (C) Microsoft Corporation. All rights reserved.  
Try the new cross-platform PowerShell https://aka.ms/pscore6  
PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++  
Binary_search.cpp -o Binary_search } ; if ($?) { .\Binary_search }  
Enter length : 5  
Enter elements : 1 3 4 8 9  
Enter value to find : 8  
Element found at position : 3  
PS D:\c++ by harsh\Unit2programs> [ ]  
① 0 △ 0 Ln 18, Col 6 Spaces: 4 UTF-8 CRLF C++ Go Live Win32 2048  
Type here to search 24°C Light rain ENG 30-07-2023
```

# SINGLY LINKED LIST

## INSERTION AT BEGINNING

```
#include <iostream>
using namespace std;
struct node
{
    int data;
    struct node *next;
};
int main()
{
    struct node *head, *newnode, *temp;
    head = 0;
    newnode = (struct node *)malloc(sizeof(struct node));
    int choice=1;
    while (choice==1)
    {
        newnode = (struct node *)malloc(sizeof(struct node));
        cout << "Enter data : ";
        cin >> newnode->data;
        newnode->next=0;
        if(head==0){
            head=temp=newnode;
        }
        else{
            temp->next=newnode;
            temp=newnode;
        }
        cout<<"Do you want to enter more : ";
        cin>>choice;
    }
    int e,b,p;
    cout<<"Enter element do you want to insert at beginning : ";
    newnode=(struct node*)malloc(sizeof(struct node));
    cin>>newnode->data;
    newnode->next=head;
    head=newnode;
    cout<<"Final list : ";
    temp=head;
    while (temp!=0)
    {
        cout<<temp->data<<" ";
        temp=temp->next;
    }
    return 0;
}
```

# OUTPUT

The screenshot shows the Visual Studio Code interface with the terminal tab selected. The terminal window displays the following interaction:

```
Enter data : 3
Do you want to enter more : 0
Enter element do you want to insert at beginning : 9
Final list : 1 2 3
PS D:\c++ by harsh\Unit2programs> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ insertion_linked_list.cpp -o insertion_linked_list } ; if ($?) { .\insertion_linked_list }
Enter data : 1
Do you want to enter more : 1
Enter data : 3
Do you want to enter more : 1
Enter data : 7
Do you want to enter more : 0
Enter element do you want to insert at beginning : 9
Final list : 9 1 3 7
PS D:\c++ by harsh\Unit2programs> []
```

The terminal also shows the command used to run the program: `cd "d:\c++ by harsh\Unit2programs\" ; if (?) { g++ insertion_linked_list.cpp -o insertion_linked_list } ; if (?) { .\insertion_linked_list }`. The status bar at the bottom indicates the current line (Ln 47), column (Col 6), and other system information like weather and battery level.

## INSERTION AT END

```
#include<iostream>
using namespace std;
struct node{
    int data;
    struct node *next;
};

int main()
{
    struct node *head, *newnode, *temp;
    head = 0;
    newnode = (struct node *)malloc(sizeof(struct node));
    int choice=1;
    while (choice==1)
    {
        newnode = (struct node *)malloc(sizeof(struct node));
        cout << "Enter data : ";
        cin >> newnode->data;
        newnode->next=0;
        if(head==0){
            head=temp=newnode;
        }
        else{
            temp->next=newnode;
            temp=newnode;
        }
    }
}
```

```

        else{
            temp->next=newnode;
            temp=newnode;
        }
        cout<<"Do you want to enter more : ";
        cin>>choice;
    }

newnode=(struct node*)malloc(sizeof(struct node));
cout<<"Enter element which you want to insert at end : ";
cin>>newnode->data;
newnode->next=0;
while(temp->next!=0){
    temp=temp->next;
}
temp->next=newnode;
temp=head;
while(temp!=NULL){
    cout<<temp->data<<" ";
    temp=temp->next;
}
return 0;
}

```

## OUTPUT

The screenshot shows the Visual Studio Code interface with the terminal tab active. The terminal window displays the following output:

```

File Edit Selection View Go Run Terminal Help • insert_at_end.cpp - c++ by harsh - Visual Studio Code [Admin...]
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ insert_at_end.cpp -o insert_at_end } ; if ($?) { .\insert_at_end }
Enter data : 1
Do you want to enter more : 1
Enter data : 2
Do you want to enter more : 1
Enter data : 6
Do you want to enter more : 0
Enter element which you want to insert at end : 9
1 2 6 9
PS D:\c++\Unit2programs> []

```

The terminal also shows the file path and command used to run the program.

# INSERTION AT A SPECIFIC POSITION

```
#include<iostream>
using namespace std;
struct node{
    int data;
    struct node *next;
};
void ins(struct node *head){
    int pos;
    struct node *newnode, *temp;
    temp=head;
    newnode=(struct node*)malloc(sizeof(struct node));
    cout<<"Enter element which you want to insert : ";
    cin>>newnode->data;
    newnode->next=0;
    cout<<"Enter the position where you want to insert an element : ";
    cin>>pos;
    pos--;
    while(pos!=1){
        temp=temp->next;
        pos--;
    }
    newnode->next=temp->next;
    temp->next=newnode;
}
int main()
{
    struct node *head, *newnode, *temp;
    head = 0;
    newnode = (struct node *)malloc(sizeof(struct node));
    int choice=1;
    while (choice==1)
    {
        newnode = (struct node *)malloc(sizeof(struct node));
        cout << "Enter data : ";
        cin >> newnode->data;
        newnode->next=0;
        if(head==0){
            head=temp=newnode;
        }
        else{
            temp->next=newnode;
            temp=newnode;
        }
        cout<<"Do you want to enter more : ";
        cin>>choice;
    }
}
```

```

    ins(head);

    temp=head;
    while(temp!=NULL){
        cout<<temp->data<<" ";
        temp=temp->next;
    }
    return 0;
}

```

## OUTPUT

Try the new cross-platform PowerShell <https://aka.ms/pscore6>

```

PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ insert_at_pos.cpp -o insert_at_pos } ; if ($?) { .\insert_at_pos }
Enter data : 1
Do you want to enter more : 1
Enter data : 2
Do you want to enter more : 1
Enter data : 3
Do you want to enter more : 1
Enter data : 4
Do you want to enter more : 0
Enter element which you want to insert : 99
Enter the position where you want to insert an element : 2
1 99 2 3
PS D:\c++ by harsh\Unit2programs> []

```

## DELETION AT BEGINNING

```

#include<iostream>
using namespace std;
struct node{
    int data;
    node *next;
};
int main(){
    struct node *head,*newnode,*temp;
    head=0;
    int choice;
    while(choice){
        newnode=(struct node*)malloc(sizeof(struct node));

```

```

cout<<"Enter data : ";
cin>>newnode->data;
newnode->next=0;
if (head==0)
{
    head=temp=newnode;
}
else{
    temp->next=newnode;
    temp=newnode;
}
cout<<"Do u want to enter more : ";
cin>>choice;
}
temp=head;
head=temp->next;
temp=head;
while(temp!=0){
    cout<<temp->data<<" ";
    temp=temp->next;
}
return 0;
}

```

## OUTPUT

The screenshot shows the Visual Studio Code interface with the terminal tab active. The terminal window displays the following text:

```

PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ delete_node_beg.cpp
Enter data : 1
Do u want to enter more : 1
Enter data : 3
Do u want to enter more : 1
Enter data : 6
Do u want to enter more : 1
Enter data : 5
Do u want to enter more : 0
3 6 5
PS D:\c++ by harsh\Unit2programs> []

```

## DELETION AT END

```
#include <iostream>
using namespace std;
struct node
{
    int data;
    node *next;
};

struct node *del_last(struct node *head)
{
    if (head == 0)
    {
        cout << "List is already empty " << endl;
    }
    else
    {
        struct node *prenode, *temp;
        temp = head;
        prenode=head;
        while (temp->next != 0)
        {
            prenode=temp;
            temp = temp->next;
        }
        prenode->next = 0;
        free(temp);
        temp = 0;
    }
    return head;
}
int main()
{
    struct node *head, *newnode, *temp;
    head = 0;
    int choice;
    while (choice)
    {
        newnode = (struct node *)malloc(sizeof(struct node));
        cout << "Enter data : ";
        cin >> newnode->data;
        newnode->next = 0;
        if (head == 0)
        {
            head = temp = newnode;
        }
        else
```

```

    {
        temp->next = newnode;
        temp = newnode;
    }
    cout << "Do you want to enter more : ";
    cin >> choice;
}
head = del_last(head);
temp=head;
while (temp!=0)
{
    cout<<temp->data<<" ";
    temp=temp->next;
}
return 0;
}

```

## OUTPUT

The screenshot shows the Visual Studio Code interface with the terminal tab active. The terminal output is as follows:

```

Try the new cross-platform PowerShell https://aka.ms/pscore6
PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++
Enter data : 1
Do you want to enter more : 1
Enter data : 2
Do you want to enter more : 1
Enter data : 6
Do you want to enter more : 1
Enter data : 5
Do you want to enter more : 0
1 2 6
PS D:\c++ by harsh\Unit2programs> []

```

The status bar at the bottom provides file navigation, code analysis, and system status information.

# DELETION AT A SPECIFIC POSITION

```
#include<iostream>
using namespace std;
struct node{
    int data;
    node *next;
};

void deleteFromPosition(struct node *head){
    struct node *temp, *nextnode;
    int pos;
    temp=head;
    cout<<"Enter position : ";
    cin>>pos;
    int i=1;
    while(i<pos-1){
        temp=temp->next;
        i++;
    }
    nextnode=temp->next;
    temp->next=nextnode->next;
    free(nextnode);
}
int main(){
    struct node *head,*newnode,*temp;
    head=0;
    int choice;
    while(choice){
        newnode=(struct node*)malloc(sizeof(struct node));
        cout<<"Enter data : ";
        cin>>newnode->data;
        newnode->data=0;
        if(head==0){
            head=temp=newnode;
        }
        else{
            temp->next=newnode;
            temp=newnode;
        }
        cout<<"Enter choice : ";
        cin>>choice;
    }
    deleteFromPosition(head);
    temp=head;
    while(temp!=NULL){
        cout<<temp->data<<" ";
        temp=temp->next;
    }
}
```

```

        temp=temp->next;
    }
    return 0;
}

```

## OUTPUT

```

PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++
Enter data : 1
Enter choice : 1
Enter data : 2
Enter choice : 1
Enter data : 3
...
Enter choice : 1
Enter data : 4
Enter choice : 1
Enter data : 5
Enter choice : 1
Enter data : 0
Enter choice : 0
Enter position : 2
1 3 4 5 0
PS D:\c++ by harsh\Unit2programs> []

```

The screenshot shows a Windows desktop environment with Visual Studio Code open. The terminal window displays the execution of a C++ program. The program performs several insertions at the beginning of a doubly linked list and then prints the list. The terminal window is part of the Visual Studio Code interface, which also includes a sidebar with icons for problems, output, terminal, and debug console.

## DOUBLY LINKED LIST

### INSERTION AT BEGINNING

```

#include<iostream>
using namespace std;
struct node{
    int data;
    node *next, *prev;
};
int main(){

    struct node *head, *tail;
    head=0;
    struct node *newnode;
    int choice;
    while(choice){
        newnode=(struct node*)malloc(sizeof(struct node));
        cout<<"Enter data : ";

```

```
cin>>newnode->data;
newnode->next=0;
newnode->prev=0;
if (head==0)
{
    head=tail=newnode;
}
else{
    tail->next=newnode;
    newnode->prev=tail;
    tail=newnode;
}
cout<<"Enter choice : ";
cin>>choice;
}
newnode=(struct node*)malloc(sizeof(struct node));
newnode->prev=0;
cout<<"Enter element you want to insert at beg : ";
cin>>newnode->data;
newnode->next=0;
head->prev=newnode;
newnode->next=head;
head=newnode;
tail=head;
while(tail!=0){
    cout<<tail->data<<" ";
    tail=tail->next;
}
return 0;
}
```

# OUTPUT

```
File Edit Selection View Go Run ... double_linked_list_create.cpp - c++ by harsh - Visual Studio Code [A...]
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6
PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ double_linked_list_
create.cpp -o double_linked_list_create } ; if ($?) { .\double_linked_list_create }
Enter data : 1
Enter choice : 1
Enter data : 2
Enter choice : 1
Enter data : 5
Enter choice : 0
Enter element you want to insert at beg : 9
9 1 2 5
PS D:\c++ by harsh\Unit2programs>
```

## INSERTION AT END

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

struct node{
    int data;
    struct node *next, *prev;
};

int main(){
    struct node *head, *newnode, *tail;
    head=0;
    int choice;
    while(choice){
        cout<<"Enter data : ";
        newnode=(struct node*)malloc(sizeof(struct node));
        cin>>newnode->data;
        newnode->next=0;
        newnode->prev=0;
        if(head==0){
            head=tail=newnode;
        }
        else{
            tail->next=newnode;
        }
    }
}
```

```
    newnode->prev=tail;
    tail=newnode;
}
cout<<"Enter choice : ";
cin>>choice;
}
struct node *temp;
newnode=(struct node*)malloc(sizeof(struct node));
cout<<"Enter element you want to insert at end : ";
newnode->prev=0;
cin>>newnode->data;
newnode->data=0;
tail=head;
while(tail->next!=0){
    tail=tail->next;
}
tail->next=newnode;
newnode->prev=tail;
tail=head;
while(tail!=0){
    cout<<tail->data<<" ";
    tail=tail->next;
}
return 0;
}
```

# OUTPUT

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++
insert_at_end }
Enter data : 1
Enter choice : 1
Enter data : 2
Enter choice : 1
Enter data : 3
Enter choice : 0
Enter element you want to insert at end : 9
1 2 3 9
PS D:\c++ by harsh\Unit2programs>
```

## INSERTION AT A GIVEN POSITION

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

struct node{
    int data;
    struct node *next, *prev;
};

int main(){
    struct node *head, *newnode, *tail;
    head=0;
    int choice;
    while(choice){
        cout<<"Enter data : ";
        newnode=(struct node*)malloc(sizeof(struct node));
        cin>>newnode->data;
        newnode->next=0;
        newnode->prev=0;
        if(head==0){
            head=tail=newnode;
        }
        else{
            tail->next=newnode;
        }
    }
}
```

```
    newnode->prev=tail;
    tail=newnode;
}
cout<<"Enter choice : ";
cin>>choice;
}
struct node *temp;
int pos, i=1;
cout<<"Enter position : ";
cin>>pos;
newnode=(struct node*)malloc(sizeof(struct node));
cin>>newnode->data;
newnode->next=0;
newnode->prev=0;
temp=head;
while(i<pos-1){
    temp=temp->next;
    i++;
}
newnode->prev=temp;
newnode->next=temp->next;
temp->next=newnode;
temp->next->prev=newnode;
tail=head;
while(tail!=0){
    cout<<tail->data<<" ";
    tail=tail->next;
}
return 0;
}
```

# OUTPUT

The screenshot shows the Visual Studio Code interface with the terminal tab selected. The terminal window displays the following command-line session:

```
PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ insert_at_pos }
Enter data : 1
Enter choice : 1
Enter data : 2
Enter choice : 1
Enter data : 3
...
Enter choice : 1
Enter data : 6
Enter choice : 1
Enter data : 8
Enter choice : 0
Enter position : 2
9
1 9 2 3 6 8
PS D:\c++ by harsh\Unit2programs>
```

The terminal status bar indicates Ln 19, Col 31, Spaces: 4, UTF-8, CRLF, C++, Go Live, Win32, and a battery icon.

## INSERTION AFTER THE POSITION

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

struct node{
    int data;
    struct node *next, *prev;
};

int main(){
    struct node *head, *newnode, *tail;
    head=0;
    int choice;
    while(choice){
        cout<<"Enter data : ";
        newnode=(struct node*)malloc(sizeof(struct node));
        cin>>newnode->data;
        newnode->next=0;
        newnode->prev=0;
        if(head==0){
            head=tail=newnode;
        }
        else{
            tail->next=newnode;
            newnode->prev=tail;
            tail=newnode;
        }
    }
}
```

```
    newnode->prev=tail;
    tail=newnode;
}
cout<<"Enter choice : ";
cin>>choice;
}
struct node *temp;
int pos, i=1;
cout<<"Enter position : ";
cin>>pos;
newnode=(struct node*)malloc(sizeof(struct node));
cin>>newnode->data;
newnode->next=0;
newnode->prev=0;
temp=head;
while(i<pos){
    temp=temp->next;
    i++;
}
newnode->prev=temp;
newnode->next=temp->next;
temp->next=newnode;
temp->next->prev=newnode;
tail=head;
while(tail!=0){
    cout<<tail->data<<" ";
    tail=tail->next;
}
return 0;
}
```

# OUTPUT

```
File Edit Selection View Go Run ... doubly_insert_at_pos.cpp - c++ by harsh - Visual Studio Code [Ad... — □ PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE + ▾ ▾ Windows PowerShell Copyright (C) Microsoft Corporation. All rights reserved. Try the new cross-platform PowerShell https://aka.ms/pscore6 PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ doubly_insert_at_pos insert_at_pos } Enter data : 1 Enter choice : 1 Enter data : 2 Enter choice : 1 Enter data : 3 Enter choice : 1 Enter data : 5 Enter choice : 1 Enter data : 9 Enter choice : 0 Enter position : 3 66 1 2 3 66 5 9 PS D:\c++ by harsh\Unit2programs> 0 0 △ 0 Ln 1, Col 1 (1159 selected) Spaces: 4 UTF-8 CRLF C++ Go Live Win32 18:0 Type here to search ○ ⓘ 🌐 📧 ⚡ 🔍 🎯 🌈 24°C Light rain ⌂ ENG 01-08-2
```

## DELETION FROM BEGINNING

```
#include<iostream>
using namespace std;
struct node{
    int data;
    struct node *next, *prev;
};

int main(){
    struct node *head, *newnode, *tail;
    head=0;
    int choice;
    while(choice){
        newnode=(struct node*)malloc(sizeof(struct node));
        cout<<"Enter data : ";
        cin>>newnode->data;
        newnode->next=0;
        newnode->prev=0;
        if(head==0){
            head=tail=newnode;
        }
        else{
            tail->next=newnode;
        }
    }
}
```

```

        newnode->prev=tail;
        tail=newnode;
    }
    cout<<"Enter choice : ";
    cin>>choice;
}
struct node *temp;
if(tail==0){
    cout<<"List is empty "<<endl;
}
else{
    temp=head;
    head=head->next;
    head->prev=0;
}
tail=head;
while(tail!=0){
    cout<<tail->data<<" ";
    tail=tail->next;
}
return 0;
}

```

## OUTPUT

File Edit Selection View Go Run ... double\_linked\_list\_del\_beg.cpp - c++ by harsh - Visual Studio Code [...]

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell <https://aka.ms/pscore6>

```

PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ double_linked_list_
del_beg.cpp -o double_linked_list_del_beg } ; if ($?) { .\double_linked_list_del_beg }
Enter data : 1
Enter choice : 1
Enter data : 2
Enter choice : 1
Enter data : 4
Enter choice : 1
Enter data : 8
Enter choice : 0
2 4 8
PS D:\c++ by harsh\Unit2programs> []

```

⑧ 0 △ 0 Ln 14, Col 31 Spaces: 4 UTF-8 CRLF C++ ⚡ Go Live Win32 ⚡ 24°C Light rain ⌂ ENG 01-08-21

## DELETION FROM END

```
#include<iostream>
using namespace std;
struct node{
    int data;
    struct node *next, *prev;
};

int main(){
    struct node *head, *newnode, *tail;
    head=0;
    int choice;
    while(choice){
        newnode=(struct node*)malloc(sizeof(struct node));
        cout<<"Enter data : ";
        cin>>newnode->data;
        newnode->next=0;
        newnode->prev=0;
        if(head==0){
            head=tail=newnode;
        }
        else{
            tail->next=newnode;
            newnode->prev=tail;
            tail=newnode;
        }
        cout<<"Enter choice : ";
        cin>>choice;
    }
    struct node *temp;
    if(tail==0){
        cout<<"List is empty "<<endl;
    }
    else{
        temp=tail;
        tail->prev->next=0;
        tail=tail->prev;
    }
    tail=head;
    while(tail!=0){
        cout<<tail->data<<" ";
        tail=tail->next;
    }
    return 0;
}
```

# OUTPUT

The screenshot shows a Visual Studio Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, ...
- Title Bar:** double\_linked\_list\_del\_end.cpp - c++ by harsh - Visual Studio Code [...]
- Left Sidebar:** Includes icons for PROBLEMS, OUTPUT, TERMINAL, DEBUG CONSOLE, and other development tools.
- Terminal Tab:** Shows the command-line output of a C++ program. The program prompts for data and choice, then prints the result.
- Output:**

```
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ double_linked_list_
del_end.cpp -o double_linked_list_del_end } ; if ($?) { .\double_linked_list_del_end }
Enter data : 1
Enter choice : 1
Enter data : 2
Enter choice : 1
Enter data : 3
Enter choice : 1
Enter data : 5
Enter choice : 0
1 2 3
PS D:\c++ by harsh\Unit2programs>
```
- Right Sidebar:** Shows sections for powers and Code.
- Bottom Status Bar:** ShowsLn 1, Col 1 (931 selected), Spaces: 4, UTF-8, CRLF, C++, Go Live, Win32, and system status (24°C, Light rain, ENG, 01-08-2023).

## DELETION FROM A SPECIFIC POSITION

```
#include<iostream>
using namespace std;
struct node{
    int data;
    struct node *next, *prev;
};

int main(){
    struct node *head, *newnode, *tail;
    head=0;
    int choice, count;
    while(choice){
        newnode=(struct node*)malloc(sizeof(struct node));
        cout<<"Enter data : ";
        cin>>newnode->data;
        newnode->next=0;
        newnode->prev=0;
        if(head==0){
            head=tail=newnode;
        }
        else{
            tail->next=newnode;
        }
    }
}
```

```
    newnode->prev=tail;
    tail=newnode;
}
cout<<"Enter choice : ";
cin>>choice;
count++;
}
int pos, i=1;
struct node *temp;
cout<<"Enter position : ";
cin>>pos;
if(pos<0||pos>count){
    cout<<"Invalid position : ";
}
else{
    temp=head;
    while(i<pos){
        temp=temp->next;
        i++;
    }
    temp->prev->next=temp->next;
    temp->next->prev=temp->prev;
}

tail=head;
while(tail!=0){
    cout<<tail->data<<" ";
    tail=tail->next;
}
return 0;
}
```

# OUTPUT

The screenshot shows the Visual Studio Code interface with the following details:

- Top Bar:** File, Edit, Selection, View, Go, Run, ...
- Title Bar:** double\_linked\_list\_del\_pos.cpp - c++ by harsh - Visual Studio Code [...]
- Left Sidebar:** Icons for PROBLEMS, OUTPUT, TERMINAL, and DEBUG CONSOLE.
- Terminal Content:**

```
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ double_linked_list_
{ .\double_linked_list_del_pos }

Enter data : 1
Enter choice : 1
Enter data : 2
Enter choice : 1
Enter data : 3
Enter choice : 1
Enter data : 4
Enter choice : 1
Enter data : 8
Enter choice : 1
Enter data : 7
Enter choice : 0
Enter position : 3
1 2 4 8 7
PS D:\c++ by harsh\Unit2programs>
```
- Right Sidebar:** Icons for powers and Code.
- Bottom Status Bar:** Ln 19, Col 31, Spaces: 4, UTF-8, CRLF, C++, Go Live, Win32, 18:30, 01-08-2024.
- Taskbar:** Shows icons for File Explorer, Task View, Taskbar settings, Edge, Mail, File Explorer, Word, and Google Chrome.
- System Tray:** Shows weather (24°C Light rain), battery, signal, and ENG.

# CIRCULAR LINKED LIST

## INSERTION AT BEGINNING

```
#include<iostream>
using namespace std;
struct node{
    int data;
    struct node *next;
};
int main(){
    struct node *head,*newnode,*temp;
    int choice;
    head=0;
    while(choice){
        newnode=(struct node*)malloc(sizeof(struct node));
        cout<<"Enter data : ";
        cin>>newnode->data;
        newnode->next=0;
        if (head==0)
        {
            head=temp=newnode;
        }
        else{
            temp->next=newnode;
            temp=newnode;
        }
        temp->next=head;
        cout<<"Enter choice : ";
        cin>>choice;
    }

    //ADDING ELEMENT AT BEGINNING
    newnode=(struct node*)malloc(sizeof(struct node));
    cin>>newnode->data;
    temp = head;
    while(temp->next != head)
        temp = temp->next;
    newnode->next = head;
    temp -> next = newnode;
    head = newnode;
    temp=head;
    while(temp->next!=head){
        cout<<temp->data<<" ";
        temp=temp->next;
    }
}
```

```
    cout<<temp->data;
    return 0;
}
```

## OUTPUT

The screenshot shows the Visual Studio Code interface with the 'TERMINAL' tab selected. The terminal window displays the following output:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ cll_insertion_beg.c _beg }
Enter data : 1
Enter choice : 1
Enter data : 2
Enter choice : 1
Enter data : 3
Enter choice : 1
Enter data : 4
Enter choice : 0
9
9 1 2 3 4
PS D:\c++ by harsh\Unit2programs>
```

The terminal also shows the status bar with 'Ln 41, Col 25' and other settings like 'UTF-8', 'CRLF', and 'C++'. Below the terminal is a taskbar with various icons and a system tray showing weather information.

## INSERTION AT END

```
#include<iostream>
using namespace std;
struct node{
    int data;
    struct node *next;
};
int main(){
    struct node *head, *newnode, *temp;
    head=0;
    int choice;
    while(choice){
        newnode=(struct node*)malloc(sizeof(struct node));
        cout<<"Enter data : ";
        cin>>newnode->data;
        newnode->next=0;
        if (head==0)
        {
            head=temp=newnode;
```

```
    }
    else{
        temp->next=newnode;
        temp=newnode;
    }
    temp->next=head;
    cout<<"Enter choice : ";
    cin>>choice;
}

//INSERTING AT END
newnode=(struct node*)malloc(sizeof(struct node));
cin>>newnode->data;
newnode->next=0;
temp->next=newnode;
temp=newnode;
temp->next=head;
//DISPLAYING
temp=head;
while(temp->next!=head){
    cout<<temp->data<<" ";
    temp=temp->next;
}
cout<<temp->data;
return 0;
}
```

# OUTPUT

The screenshot shows the Visual Studio Code interface with the 'TERMINAL' tab selected. The terminal window displays the execution of a C++ program named 'cll\_insertion\_end'. The program prompts the user for data and choice values, and then prints them out. The terminal also shows the current working directory as 'D:\c++ by harsh\Unit2programs'.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ cll_insertion_end.c
pp -o cll_insertion_end } ; if ($?) { .\cll_insertion_end }
Enter data : 1
Enter choice : 1
Enter data : 2
Enter choice : 1
Enter data : 3
Enter choice : 1
Enter data : 4
Enter choice : 0
8
1 2 3 4 8
PS D:\c++ by harsh\Unit2programs>
```

The status bar at the bottom indicates the following information: Ln 36, Col 17, Spaces: 4, UTF-8, CRLF, C++, Go Live, Win32, and system icons for search, file, browser, mail, and others. The taskbar at the bottom of the screen shows the Windows Start button, a search bar, and pinned application icons for File Explorer, Mail, and others. The system tray shows the date (07-08-22), time (11:24), weather (25°C Rain showers), battery level, signal strength, and network connection.

# INSERTION AT A SPECIFIC POSITION

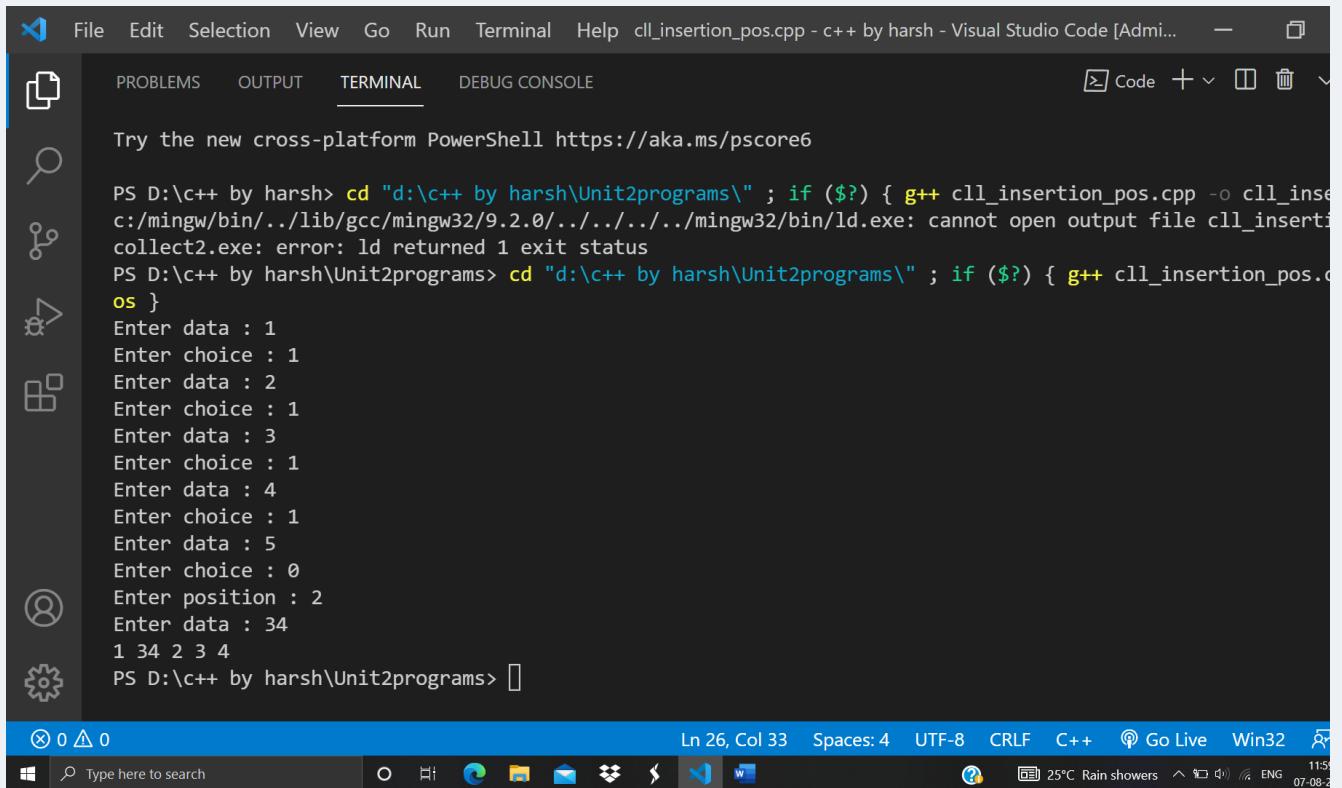
```
#include<iostream>
using namespace std;
struct node{
    int data;
    struct node *next;
};
int main(){
    struct node *head, *newnode, *temp;
    head=0;
    int choice, l=0;
    while(choice){
        newnode=(struct node*)malloc(sizeof(struct node));
        cout<<"Enter data : ";
        cin>>newnode->data;
        newnode->next=0;
        l++;
        if (head==0)
        {
            head=temp=newnode;
        }
        else{
            temp->next=newnode;
            temp=newnode;
        }
        temp->next=head;
        cout<<"Enter choice : ";
        cin>>choice;
    }
    struct node *tail;
    int pos, i=1;
    cout<<"Enter position : ";
    cin>>pos;
    if(pos<1 || pos>l){
        cout<<"Invalid position "<<endl;
    }
    else{
        newnode=(struct node*)malloc(sizeof(struct node));
        struct node *p=temp->next;
        cout<<"Enter data : ";
        cin>>newnode->data;
        newnode->next=0;
        while(i<pos-1){
            p=p->next;
            i++;
        }
```

```

    }
    newnode->next=p->next;
    p->next=newnode;
}
temp=head;
while(temp->next!=head){
    cout<<temp->data<<" ";
    temp=temp->next;
}
return 0;
}

```

## OUTPUT



```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ cll_insertion_pos.cpp -o cll_insertion_pos ; .\cll_insertion_pos } collect2.exe: error: ld returned 1 exit status
PS D:\c++ by harsh\Unit2programs> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ cll_insertion_pos.cpp -o cll_insertion_pos ; .\cll_insertion_pos }
Enter data : 1
Enter choice : 1
Enter data : 2
Enter choice : 1
Enter data : 3
Enter choice : 1
Enter data : 4
Enter choice : 1
Enter data : 5
Enter choice : 0
Enter position : 2
Enter data : 34
1 34 2 3 4
PS D:\c++ by harsh\Unit2programs> []

```

## DELETION AT BEGINNING

```

#include<iostream>
using namespace std;
struct node{
    int data;
    struct node *next;
};
int main(){
    struct node *newnode, *head, *tail;
    head=0;
    int choice;
}

```

```

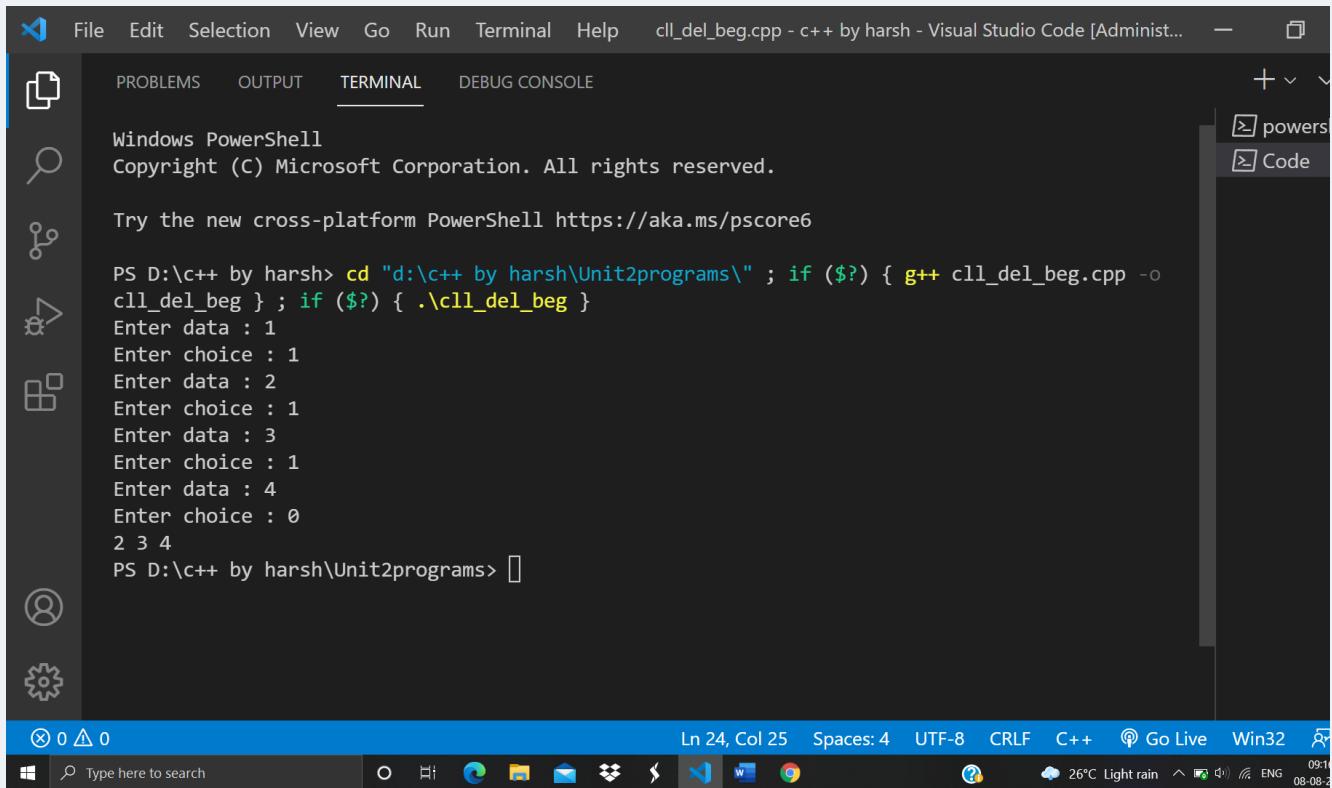
while(choice){
    newnode=(struct node*)malloc(sizeof(struct node));
    cout<<"Enter data : ";
    cin>>newnode->data;
    newnode->next=0;
    if(head==0){
        head=tail=newnode;
    }
    else
    {
        tail->next=newnode;
        tail=newnode;
    }
    tail->next=head;
    cout<<"Enter choice : ";
    cin>>choice;
}

//DELETE FROM BEGINNING
struct node *ptr = head;
while(ptr -> next != head)
{
    ptr = ptr -> next;
    ptr->next = head->next;
    free(head);
    head = ptr->next;

    tail=head;
    while(tail->next!=head){
        cout<<tail->data<<" ";
        tail=tail->next;
    }
    cout<<tail->data;
    return 0;
}

```

# OUTPUT



```
File Edit Selection View Go Run Terminal Help cll_del_beg.cpp - c++ by harsh - Visual Studio Code [Administrator] — □  
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE  
Windows PowerShell  
Copyright (C) Microsoft Corporation. All rights reserved.  
Try the new cross-platform PowerShell https://aka.ms/pscore6  
PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ cll_del_beg.cpp -o cll_del_beg } ; if ($?) { .\cll_del_beg }  
Enter data : 1  
Enter choice : 1  
Enter data : 2  
Enter choice : 1  
Enter data : 3  
Enter choice : 1  
Enter data : 4  
Enter choice : 0  
2 3 4  
PS D:\c++ by harsh\Unit2programs> []  
Ln 24, Col 25 Spaces: 4 UTF-8 CRLF C++ Go Live Win32 ⚙  
Type here to search ○ ⓘ 🌐 📎 📧 ⚡ 🖼 📁 🌐 ENG 08:08:24
```

## DELETION FROM END

```
#include<iostream>
using namespace std;
struct node{
    int data;
    struct node *next;
};
int main(){
    struct node *newnode, *head, *tail;
    head=0;
    int choice;
    while(choice){
        newnode=(struct node*)malloc(sizeof(struct node));
        cout<<"Enter data : ";
        cin>>newnode->data;
        newnode->next=0;
        if(head==0){
            head=tail=newnode;
        }
        else
        {
            tail->next=newnode;
            tail=newnode;
        }
    }
}
```

```
        }
        tail->next=head;
        cout<<"Enter choice : ";
        cin>>choice;
    }
    struct node *current, *prenode;
    if(head->next==head){
        head=0;
        free(head);
    }
    else{
        current=head;
        while(current->next!=head){
            prenode=current;
            current=current->next;
        }
        prenode->next=current->next;
        free(current);
    }
    tail=head;
    while(tail->next!=head){
        cout<<tail->data<<" ";
        tail=tail->next;
    }
    cout<<tail->data;
    return 0;
}
```

# OUTPUT

The screenshot shows a Windows desktop environment with Visual Studio Code open. The code editor has a dark theme. On the left is a sidebar with icons for file operations, search, and other development tools. The main area has tabs for 'PROBLEMS', 'OUTPUT' (which is selected), 'TERMINAL', and 'DEBUG CONSOLE'. The terminal window contains the following text:

```
PS D:\c++ by harsh\Unit2programs> cd "d:\c++ by harsh\Unit2programs\" ;  
if ($?) { g++ cll_del_end.cpp -o cll_del_end } ; if ($?) { .\cll_del_end  
}  
Enter data : 1  
Enter choice : 1  
Enter data : 2  
Enter choice : 1  
Enter data : 3  
Enter choice : 1  
Enter data : 4  
Enter choice : 0  
1 2 3  
PS D:\c++ by harsh\Unit2programs>
```

Below the terminal, the status bar shows: Ln 18, Col 10, Spaces: 4, UTF-8, CRLF, C++, Go Live, Win32, and a battery icon. The taskbar at the bottom includes icons for File Explorer, Task View, Edge browser, Mail, File Explorer, Task View, and a pinned Visual Studio Code icon. Weather and system status are also visible on the taskbar.

## DELETION AT A SPECIFIC POSITION

```
#include<iostream>
using namespace std;
struct node{
    int data;
    struct node *next;
};
int main(){
    struct node *newnode, *head, *tail;
    head=0;
    int choice;
    while(choice){
        newnode=(struct node*)malloc(sizeof(struct node));
        cout<<"Enter data : ";
        cin>>newnode->data;
        newnode->next=0;
        if(head==0){
            head=tail=newnode;
        }
        else{
            tail->next=newnode;
            tail=newnode;
        }
        tail->next=head;
        cout<<"Enter choice : ";
        cin>>choice;
    }

    //DELETE FROM POS
    struct node *curr, *nextnode;
    int pos, i=1;
    curr=head;
    cout<<"Enter pos : ";
    cin>>pos;
    while(i<pos){
        newnode=curr;
        curr=curr->next;
        i++;
    }
    newnode->next=curr->next;
    free(curr);
    tail=head;
    while(tail->next!=head){
        cout<<tail->data<<" ";
        tail=tail->next;
    }
}
```

```
    }
    cout<<tail->data;
    return 0;
}
```

## OUTPUT

```
PS D:\c++ by harsh\Unit2programs> cd "d:\c++ by harsh\Unit2programs\" ;
}
Enter data : 1
Enter choice : 1
Enter data : 2
Enter choice : 1
Enter data : 3
Enter choice : 1
Enter data : 4
Enter choice : 1
Enter data : 5
Enter choice : 0
Enter pos : 3
1 2 4 5
PS D:\c++ by harsh\Unit2programs> 
```

## BALANCED PARENTHESIS

```
#include<bits/stdc++.h>
using namespace std;
bool isValid(string s){
    int n=s.size();
    stack<char> st;
    bool ans = true;
    for (int i=0;i<n;i++){
        if(s[i]=='{' || s[i]=='(' || s[i]== '['){
            st.push(s[i]);
        }
        else if(s[i]== '}'){
            if(!st.empty() && st.top()=='('){
                st.pop();
            }
            else{
                ans = false;
            }
        }
    }
    if(st.empty())
        ans = true;
    else
        ans = false;
    return ans;
}
```

```

        break;
    }
}
else if(s[i]==']'){
    if(!st.empty() && st.top()=='['){
        st.pop();
    }
    else{
        ans = false;
        break;
    }
}
else if(s[i]=='}'){
    if(!st.empty() && st.top()=='{'){
        st.pop();
    }
    else{
        ans = false;
        break;
    }
}
if(!st.empty()){
    return false;
}
return ans;
}
int main(){
    string s;
    cout<<"Enter string : ";
    getline(cin,s);
    if(isValid(s)){
        cout<<"Valid string "<<endl;
    }
    else{
        cout<<"Invalid string "<<endl;
    }
    return 0;
}

```

## OUTPUT 1

The screenshot shows the Visual Studio Code interface with the 'TERMINAL' tab selected. The terminal window displays the following output:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ balanced_parenthesis.cpp -o balanced_parenthesis } ; if ($?) { .\balance d_parenthesis }
Enter string : [2*3+{4+8}*(5-1)]
Valid string
PS D:\c++ by harsh\Unit2programs> []
```

The terminal status bar indicates: Ln 14, Col 14 | Spaces: 4 | UTF-8 | CRLF | C++ | Go Live | Win32.

## OUTPUT 2

The screenshot shows the Visual Studio Code interface with the 'TERMINAL' tab selected. The terminal window displays the following output:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ balanced_parenthesis.cpp -o balanced_parenthesis } ; if ($?) { .\balance d_parenthesis }
Enter string : [2*5+(3+9)*3]
Invalid string
PS D:\c++ by harsh\Unit2programs> []
```

The terminal status bar indicates: Ln 1, Col 24 | Spaces: 4 | UTF-8 | CRLF | C++ | Go Live | Win32.

# BINARY SEARCH TREE

## CREATION OF BST

```
#include <stdio.h>
#include <iostream>
using namespace std;
struct node
{
    int data;
    struct node *left, *right;
};

struct node *create()
{
    struct node *temp;
    int data, choice;
    temp = (struct node *)malloc(sizeof(struct node));
    cout << "Enter 0 to exit" << endl;
    cout << "Enter 1 for newnode entry" << endl;
    cout << "Enter your choice : ";
    cin >> choice;
    if (choice == 0)
    {
        return 0;
    }
    else
    {
        cout << "Enter the data : ";
        cin >> data;
        temp->data = data;
        cout << "Enter the left child of " << data << " : ";
        temp->left = create();
        cout << "Enter the right child of " << data << " : ";
        temp->right = create();
        return temp;
    }
}
void display(struct node *root)
{
    if (root != NULL)
    {
        display(root->left);
        cout << root->data << " ";
        display(root->right);
    }
}
int main()
```

```
{  
    struct node *root;  
    root = create();  
    display(root);  
    return 0;  
}
```

## OUTPUT

The screenshot shows the Visual Studio Code interface with the terminal tab active. The terminal window displays the following command-line interaction:

```
PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ binary_tree.cpp -o binary_tree } ; if ($?) { ./binary_tree }  
Enter 0 to exit  
Enter 1 for newnode entry  
Enter your choice : 1  
Enter the data : 3  
Enter the left child of 3 :  
Enter 0 to exit  
Enter 1 for newnode entry  
Enter your choice : 1  
Enter the data : 2  
Enter the left child of 2 :  
Enter 0 to exit  
Enter 1 for newnode entry  
Enter your choice : 0  
Enter the right child of 2 :  
Enter 0 to exit  
Enter 1 for newnode entry  
Enter your choice : 0  
Enter the right child of 3 :  
Enter 0 to exit  
Enter 1 for newnode entry
```

The terminal also shows status indicators at the bottom: Ln 5, Col 2, Spaces: 3, UTF-8, CRLF, C++, Go Live, Win32, and a system tray with icons for battery, signal, and volume.

The screenshot shows a terminal window in Visual Studio Code with the following interaction:

```
Enter 1 for newnode entry
Enter your choice : 1
Enter the data : 4
Enter the left child of 4 :
Enter 0 to exit
Enter 1 for newnode entry
Enter your choice : 0
Enter the right child of 4 :
Enter 0 to exit
Enter 1 for newnode entry
Enter your choice : 1
Enter the data : 5
Enter the left child of 5 :
Enter 0 to exit
Enter 1 for newnode entry
Enter your choice : 0
Enter the right child of 5 :
Enter 0 to exit
Enter 1 for newnode entry
Enter your choice : 0
2 3 4 5
PS D:\c++ by harsh\Unit2programs>
```

The terminal interface includes tabs for PROBLEMS, OUTPUT, TERMINAL (which is selected), and DEBUG CONSOLE. A sidebar on the right shows a tree structure with nodes labeled "powers" and "Code". The status bar at the bottom shows file statistics (0△0), file encoding (Ln 5, Col 2), and system information (Spaces: 3, UTF-8, CRLF, C++, Go Live, Win32).

## INSERTION IN BST

```
#include <stdio.h>
#include <iostream>
using namespace std;
struct node
{
    int data;
    struct node *left, *right;
};
struct node *root=0;
struct node *create()
{
    struct node *temp;
    int data, choice;
    temp = (struct node *)malloc(sizeof(struct node));
    cout << "Enter 0 to exit" << endl;
    cout << "Enter 1 for newnode entry" << endl;
    cout << "Enter your choice : ";
    cin >> choice;
    if (choice == 0)
    {
        return 0;
    }
    else
    {
```

```

        cout << "Enter the data : ";
        cin >> data;
        temp->data = data;
        cout << "Enter the left child of " << data << " : " << endl;
        temp->left = create();
        cout << "Enter the right child of " << data << " : " << endl;
        temp->right = create();
        return temp;
    }
}

struct node *insert(struct node *root,int d){
    struct node *t,*parent;
    t=(struct node*)malloc(sizeof(struct node));
    t->data=d;
    t->left=0;
    t->right=0;
    parent=root;
    if(root==0){
        root=t;
    }
    if(root->data<d){
        root->right = insert(root->right,d);
    }
    else if(root->data>d){
        root->left=insert(root->left,d);
    }
    return root;
}
void display(struct node *root)
{
    if (root != NULL)
    {
        display(root->left);
        cout << root->data << " ";
        display(root->right);
    }
}
int main()
{
    struct node *root;
    root = create();
    int n;
    cout<<"Enter element do you want insert : ";
    cin>>n;
    insert(root,n);
    display(root);
    return 0;
}

```

# OUTPUT

The image shows two separate instances of the Visual Studio Code interface, each displaying a terminal window. Both terminals are running a C++ program named bst\_insertion.cpp, which implements a binary search tree insertion algorithm.

**Terminal 1 (Top):**

```
PS D:\c++ by harsh\Unit2programs> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ bst_insertion.cpp -o bst_insertion } ; if ($?) { .\bst_insertion }
Enter 0 to exit
Enter 1 for newnode entry
Enter your choice : 1
Enter the data : 33
Enter the left child of 33 :
Enter 0 to exit
Enter 1 for newnode entry
Enter your choice : 1
Enter the data : 14
Enter the left child of 14 :
Enter 0 to exit
Enter 1 for newnode entry
Enter your choice : 0
Enter the right child of 14 :
Enter 0 to exit
Enter 1 for newnode entry
Enter your choice : 0
Enter the right child of 33 :
Enter 0 to exit
Enter 1 for newnode entry
```

**Terminal 2 (Bottom):**

```
Enter 0 to exit
Enter 1 for newnode entry
Enter your choice : 1
Enter the data : 35
Enter the left child of 35 :
Enter 0 to exit
Enter 1 for newnode entry
Enter your choice : 0
Enter the right child of 35 :
Enter 0 to exit
Enter 1 for newnode entry
Enter your choice : 1
Enter the data : 50
Enter the left child of 50 :
Enter 1 for newnode entry
Enter your choice : 0
Enter the right child of 50 :
Enter 0 to exit
Enter 1 for newnode entry
Enter your choice : 0
Enter element do you want insert : 31
14 31 33 35 50
```

# SEARCHING IN BST

```
#include<bits/stdc++.h>
using namespace std;
struct node{
    int data;
    struct node *left,*right;
};

struct node *create(){
    struct node *temp;
    int choice,data;
    temp=(struct node*)malloc(sizeof(struct node));
    cout<<"Enter 0 to exit"<<endl;
    cout<<"Enter 1 to enter data"<<endl;
    cout<<"Enter your choice : ";
    cin>>choice;
    if(choice==0){
        return 0;
    }
    else{
        cout<<"Enter data : ";
        cin>>data;
        temp->data=data;
        cout<<"Enter the left child of "<<data<<endl;
        temp->left=create();
        cout<<"Enter the right child of "<<data<<endl;
        temp->right=create();
        return temp;
    }
}

void search(struct node *root,int item){
    if(root==0||root->data==item){
        cout<<"Element found at address "<<root;
    }
    else if(root->data>item){
        return search(root->left,item);
    }
    else{
        return search(root->right,item);
    }
}
int main(){
    struct node *root;
    int key;
    root=create();
    cout<<"Enter node you want to search : ";
```

```
    cin>>key;
    search(root,key);
    return 0;
}
```

## OUTPUT

The screenshot shows the Visual Studio Code interface with the terminal tab selected. The terminal window displays the following command-line interaction:

```
PS D:\c++ by harsh\Unit2programs> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ search_bst.cpp -o search_bst } ; if ($?) { .\search_bst }
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 1
Enter data : 33
Enter the left child of 33
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 1
Enter data : 21
Enter the left child of 21
Enter 0 to exit
Enter your choice : 0
Enter the right child of 34
Enter 0 to exit
Enter 1 to enter data
```

The status bar at the bottom shows the following information: 0△0, Ln 12, Col 35, Spaces: 4, UTF-8, CRLF, C++, Go Live, Win32, ENG IN, and 21:0.

The screenshot shows the Visual Studio Code interface. The terminal window displays the following output:

```
Enter 0 to exit
Enter your choice : 0
Enter the right child of 34
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 0
Enter the right child of 35
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 0
Enter node you want to search : 35
Element found at address 0x706f50
PS D:\c++ by harsh\Unit2programs> []
```

The status bar at the bottom shows the current file is "search\_bst.cpp - c++ by harsh - Visual Studio C...". Other status indicators include Ln 12, Col 35, Spaces: 4, UTF-8, CRLF, C++, Go Live, Win32, and a battery icon.

## TRAVERSAL IN BST

### IN-ORDER TRAVERSAL

```
#include<iostream>
using namespace std;
struct node{
    int data;
    struct node *left,*right;
};
struct node *create(){
    struct node *temp;
    int choice;
    temp=(struct node*)malloc(sizeof(struct node));
    cout<<"Enter 0 to exit"<<endl;
    cout<<"Enter 1 to enter data"<<endl;
    cout<<"Enter your choice : ";
    cin>>choice;
    if(choice==0){
        return 0;
    }
    else{
        int data;
        cout<<"Enter data : ";
        cin>>data;
```

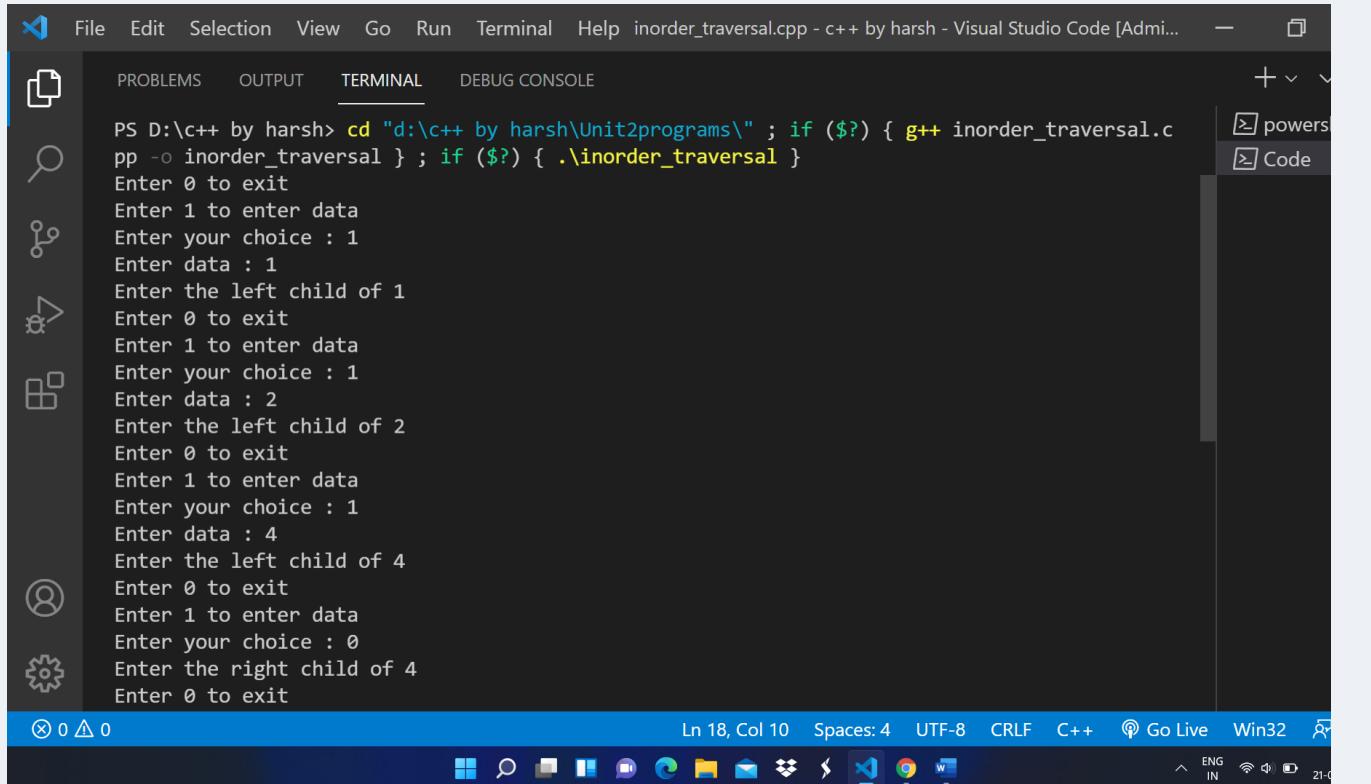
```

        temp->data=data;
        cout<<"Enter the left child of "<<data<<endl;
        temp->left=create();
        cout<<"Enter the right child of "<<data<<endl;
        temp->right=create();
        return temp;
    }
}

void traverse(struct node *root){
    if(root!=0){
        traverse(root->left);
        cout<<root->data<<" ";
        traverse(root->right);
    }
}
int main(){
    struct node *root;
    root=create();
    traverse(root);
    return 0;
}

```

## OUTPUT



The screenshot shows the Visual Studio Code interface with the terminal tab active. The terminal window displays the following command-line interaction:

```

PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ inorder_traversal.c
pp -o inorder_traversal } ; if ($?) { .\inorder_traversal }

Enter 0 to exit
Enter 1 to enter data
Enter your choice : 1
Enter data : 1
Enter the left child of 1
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 1
Enter data : 2
Enter the left child of 2
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 1
Enter data : 4
Enter the left child of 4
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 0
Enter the right child of 4
Enter 0 to exit

```

The screenshot shows a Visual Studio Code interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a file path indicating an open C++ file named 'inorder\_traversal.cpp'. The left sidebar contains icons for Problems, Output, Terminal, and Debug Console. The main area displays a terminal session with the following output:

```
Enter 1 to enter data
Enter your choice : 0
Enter the right child of 4
Enter 0 to exit
Enter the left child of 3
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 0
Enter the right child of 3
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 0
4 2 5 1 3
PS D:\c++ by harsh\Unit2programs> []
```

The status bar at the bottom showsLn 18, Col 10, Spaces: 4, UTF-8, CRLF, C++, Go Live, Win32, and a battery icon.

## PRE-ORDER TRAVERSAL

```
#include<iostream>
using namespace std;
struct node{
    int data;
    struct node *left, *right;
};

struct node *create(){
    struct node *temp;
    int choice;
    temp=(struct node*)malloc(sizeof(struct node));
    cout<<"Enter 0 to exit"<<endl;
    cout<<"Enter 1 to enter data"<<endl;
    cout<<"Enter your choice : ";
    cin>>choice;
    if(choice==0){
        return 0;
    }
    else{
        int data;
        cout<<"Enter data : ";
        cin>>data;
        temp->data=data;
        cout<<"Enter left child of "<<data<<endl;
```

```

        temp->left=create();
        cout<<"Enter right child of "<<data<<endl;
        temp->right=create();
        return temp;
    }
}

void traversal(struct node *root){
    if(root!=0){
        cout<<root->data<<" ";
        traversal(root->left);
        traversal(root->right);
    }
}
int main(){
    struct node *root;
    root=create();
    traversal(root);
    return 0;
}

```

## OUTPUT

The screenshot shows the Visual Studio Code interface with the terminal tab active. The terminal window displays the execution of a C++ program named 'pre\_order\_traversal'. The program creates a binary tree and performs a pre-order traversal on it. The user is prompted to enter data for each node, and the program prints the resulting traversal sequence.

```

PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ pre_order_traversal .cpp -o pre_order_traversal } ; if ($?) { .\pre_order_traversal }
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 1
Enter data : 1
Enter left child of 1
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 1
Enter data : 2
Enter left child of 2
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 1
Enter data : 4
Enter left child of 4
Enter 1 to enter data
Enter your choice : 0
Enter right child of 3
Enter 0 to exit

```

```
File Edit Selection View Go Run Terminal ... pre_order_traversal.cpp - c++ by harsh - Visual Studio Code [Adm...]
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE + ∨
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 1
Enter data : 4
Enter left child of 4
Enter 1 to enter data
Enter your choice : 0
Enter right child of 3
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 0
1 2 4 5 3
PS D:\c++ by harsh\Unit2programs> 
```

## POST-ORDER TRAVERSAL

```
#include<iostream>
using namespace std;
struct node{
    int data;
    struct node *left, *right;
};

struct node *create(){
    struct node *temp;
    int choice;
    temp=(struct node*)malloc(sizeof(struct node));
    cout<<"Enter 0 to exit"<<endl;
    cout<<"Enter 1 to enter data"<<endl;
    cout<<"Enter your choice : ";
    cin>>choice;
    if(choice==0){
        return 0;
    }
    else{
        int data;
        cout<<"Enter data : ";
        cin>>data;
        temp->data=data;
        cout<<"Enter left child of "<<data<<endl;
        cout<<"Enter right child of "<<data<<endl;
    }
}
```

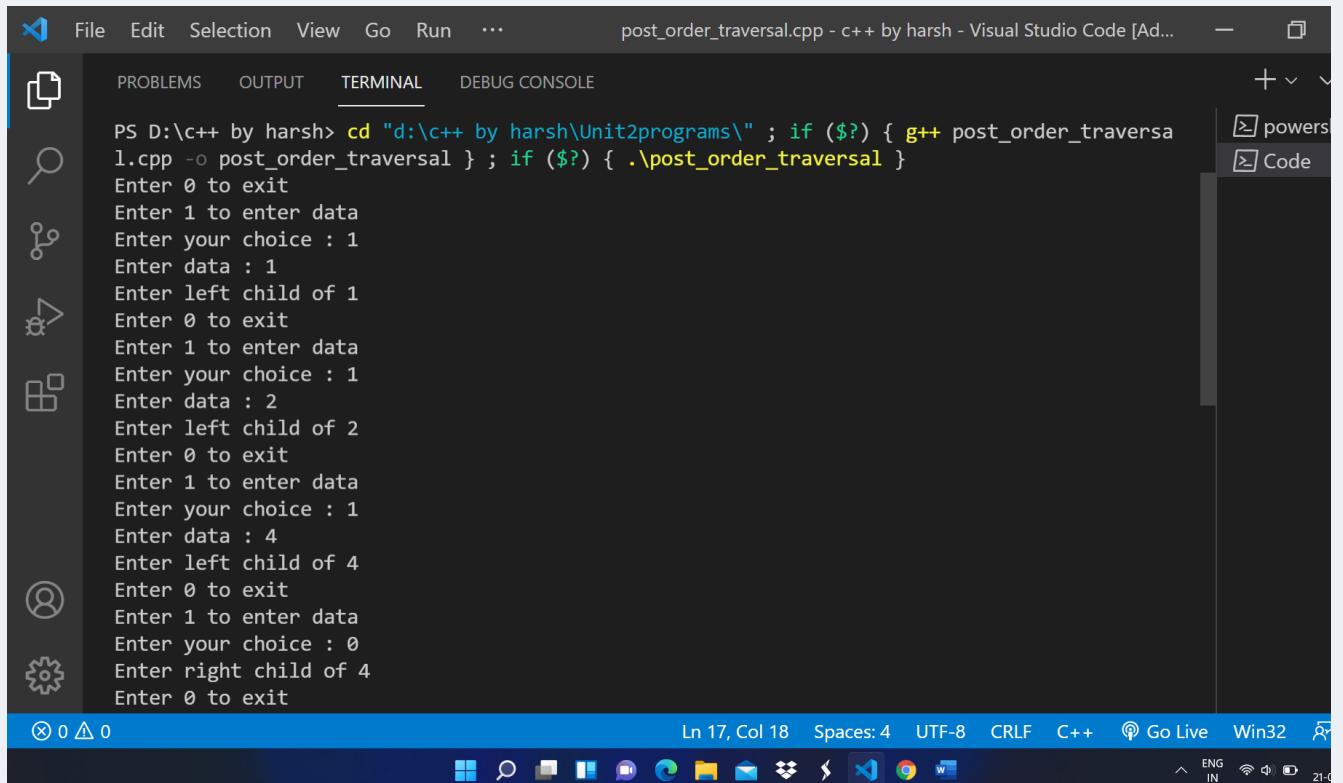
```

        temp->left=create();
        cout<<"Enter right child of "<<data<<endl;
        temp->right=create();
        return temp;
    }
}

void traversal(struct node *root){
    if(root!=0){
        traversal(root->left);
        traversal(root->right);
        cout<<root->data<<" ";
    }
}
int main(){
    struct node *root;
    root=create();
    traversal(root);
    return 0;
}

```

## OUTPUT



The screenshot shows the Visual Studio Code interface with the terminal tab active. The terminal window displays the execution of a C++ program named 'post\_order\_traversal'. The program creates a binary tree and performs a post-order traversal on it. The terminal output is as follows:

```

PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ post_order_traversal.cpp -o post_order_traversal } ; if ($?) { .\post_order_traversal }
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 1
Enter data : 1
Enter left child of 1
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 1
Enter data : 2
Enter left child of 2
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 1
Enter data : 4
Enter left child of 4
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 0
Enter right child of 4
Enter 0 to exit

```

The status bar at the bottom indicates the current line (Ln 17), column (Col 18), and encoding (UTF-8). It also shows icons for file operations, search, and other development tools.

The screenshot shows a Visual Studio Code interface. The top bar includes 'File', 'Edit', 'Selection', 'View', 'Go', 'Run', and '...'. The title bar reads 'post\_order\_traversal.cpp - c++ by harsh - Visual Studio Code [Ad...]' with icons for minimize, maximize, and close. The left sidebar has icons for file, search, file tree, and user. The main area has tabs for 'PROBLEMS', 'OUTPUT', 'TERMINAL' (which is selected), and 'DEBUG CONSOLE'. The terminal output is as follows:

```
Enter right child of 2
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 1
Enter data : 5
Enter data : 3
Enter left child of 3
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 0
Enter right child of 3
Enter 0 to exit
Enter 1 to enter data
Enter your choice : 0
4 5 2 3 1
PS D:\c++ by harsh\Unit2programs> []
```

The status bar at the bottom shows 'Ln 17, Col 18' and other settings like 'Spaces: 4', 'UTF-8', 'CRLF', 'C++', 'Go Live', 'Win32', and a date/time '21-0'. Below the status bar is a taskbar with icons for various applications.

## GROUP ACTIVITY WORK

**1. You are making an IPOD playlist to hear the songs. Assuming that shuffle functions are not applicable, choose an appropriate data structure that will add and delete songs onto you're your IPOD in such a way that recently inserted songs will always be first song currently on the IPOD.**

```
#include <bits/stdc++.h>
using namespace std;

//REPRESENTING NODES
struct node
{
    char data[50];
    struct node *next;
};

//DECLARING HEAD, NEWNODE AND TEMP POINTER
```

```

struct node *head, *newnode, *temp;

//THIS FUNCTION WILL ADD NEW SONG AT BEGINNING
void insertAtBeg(int len)
{
    cin.ignore();
    newnode = (struct node *)malloc(sizeof(struct node));
    cout << "Enter song name which you want to insert in IPOD : ";
    cin.getline(newnode->data, 50); //TAKING SONG NAME AS AN INPUT
    temp = head;

    //USING WHILE LOOP FOR EVERY TEMP->NEXT!=HEAD
    while (temp->next != head)
        temp = temp->next;

    newnode->next = head;
    temp->next = newnode;
    head = newnode;
    len++; //INCREMENTING VALUE OF LEN BY ONE
    cout << newnode->data << " inserted successfully" << endl;
    cout << "-----x-----" << endl;
}

void del(int len)
{
    //DECLARING CURR
    struct node *curr;
    int pos, i = 1;
    curr = head;
    cout << "Enter song number which you want to delete from IPOD : ";
    cin >> pos;

    //CHECKING IF POSITION OF A GIVEN SONG IS 1
    if (pos == 1)
    {
        //DECLARING PTR=HEAD
        struct node *ptr = head;

        //USING WHILE LOOP FOR EVERY PTR->NEXT!=HEAD
        while (ptr->next != head)
            ptr = ptr->next;
        ptr->next = head->next;
        free(head);
        head = ptr->next;
        len--; //DECREMENTING LENGTH BY ONE
        cout << "\tSong deleted successfully" << endl;
    }

    //CHECKING IF POSITION IS LESS THAN GIVEN LENGTH
}

```

```

else if (pos < len)
{
    //USING WHILE LOOP FOR EVERY I LESS THAN POSITION
    while (i < pos)
    {
        newnode = curr;
        curr = curr->next;
        i++;
    }
    newnode->next = curr->next;
    free(curr);
    len--; //DECREMENTING LENGTH BY ONE
    cout << "\tSong deleted successfully" << endl;
}

//CHECKING IF POSITION EQUAL TO LENGTH
else if (pos == len)
{
    //DECLARING CURRENT AND PRENODE
    struct node *current, *prenode;

    //CHECKING IF HEAD->NEXT EQUAL TO HEAD
    if (head->next == head)
    {
        head = 0;
        free(head);
        cout << "\tSong deleted successfully" << endl;
    }
    else
    {
        current = head;

        //USING WHILE LOOP FOR EVERY CURRENT->NEXT!=HEAD
        while (current->next != head)
        {
            prenode = current;
            current = current->next;
        }
        prenode->next = current->next;
        free(current);
        len--; // DECREMENTING LENGTH BY ONE
    }
    cout << "\tSong deleted successfully" << endl;
}

//IF ALL THE CONDITIONS FAILS
else
{
    cout << "Song number is out of range" << endl;
}

```

```

    }
    cout << "-----x-----" << endl;
}

void play()
{
    cout << "\tPLAYING SONGS" << endl;
    cout << "-----x-----" << endl;
    temp = head;

    // USING WHILE LOOP FOR EVERY TEM->NEXT!=HEAD
    while (temp->next != head)
    {
        cout << temp->data << endl;
        temp = temp->next;
    }
    cout << temp->data << endl;
    cout << "-----x-----" << endl;
}

int main()
{
    cout << "\t\tGROUP 8 PROJECT" << endl;
    cout << "-----x-----" << endl;
    cout << "\t\tGROUP MEMBERS" << endl;
    cout << "\t      1. HARSH SHAH JAISWAL - 20BCE10556" << endl;
    cout << "\t      2. ANKIT JINDAL - 20BCE10507" << endl;
    cout << "\t      3. SIDDHIBALA MANGLAM - 20BCE10550" << endl;
    cout << "\t      4. SHANTANU KASANA - 20BCE10557" << endl;
    cout << "\t      5. NIKHITA GARG - 20BCE10553" << endl;
    cout << "-----x-----" << endl;
    int choice = 0, num, l = 0;
    head = 0;
    cout << "Enter number of songs you want to insert into the new IPOD : ";
    cin >> num;
    cin.ignore();

    //USING WHILE LOOP FOR EVERY CHOICE IS LESS THAN NUM
    while (choice < num)
    {
        //TAKING INPUT OF SONG NAME FOR NEW IPOD
        newnode = (struct node *)malloc(sizeof(struct node));
        cout << "Enter song name : ";
        cin.getline(newnode->data, 50);
        newnode->next = 0; //IMPLEMENTING NEWNODE->NEXT=0

        //CHECKING IF HEAD EQUAL TO NULL
        if (head == 0)
        {

```

```

        head = temp = newnode;
    }

    //IF ABOVE CONDITIONS FAILS
    else
    {
        temp->next = newnode;
        temp = newnode;
    }
    temp->next = head;
    choice++; //INCREMENTING CHOICE BY ONE
    l++;      //INCREMENTING LENGTH BY ONE
}
int ch;
cout << "Enter 1 to insert the song\nEnter 2 to delete the song\nEnter 3 to play
all songs\nEnter -1 to exit code" << endl;
cin >> ch;

//USING INFINITE WHILE LOOP
while (1)
{
    //CHECKING IF CH IS EQUAL TO ONE
    if (ch == 1)
    {
        //CALLING INSERT AT BEGINNING FUNCTION TO ADD GIVEN SONG AT FIRST
        insertAtBeg(l);
        l++; //INCREMENTING LENGTH BY ONE
    }

    //CHECKING IF CH IS EQUAL TO TWO
    else if (ch == 2)
    {
        //CALLING DELETE AT GIVEN POSITION FUNCTION TO DELETE THE SPECIFIED POSITION SONG
        del(l);
        l--; // DECREMENTING LENGTH BY ONE
    }

    //CHECKING IF CH IS EQUAL TO THREE
    else if (ch == 3)
    {
        //CALLING PLAY FUNCTION TO PPLAY SONGS
        play();
        cout << endl;
    }

    //CHECKING IF CH IS EQUAL TO -1
    else if (ch == -1)
    {

```

```

        cout << "Thank you for playing songs";
        break; // BREAKING THE WHILE LOOP AND EXITING THE CODE
    }

    //IF ALL OF THE ABOVE CONDITIONS FAILS
    else
    {
        cout << "Enter 1 or 2 or 3 or -1" << endl;
    }
    cout << "Enter 1 to insert the song\nEnter 2 to delete the song\nEnter 3 to p
lay all songs\nEnter -1 to exit code" << endl;
    cin >> ch; // TAKING CH AS INPUT
}

return 0;
}

```

## OUTPUT

```

File Edit Selection View Go Run Terminal Help DSA_ipod.cpp - c++ by harsh - Visual Studio Code [Administrator]
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE + ∨
powers
Code
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows
PS D:\c++ by harsh> cd "d:\c++ by harsh\Unit2programs\" ; if ($?) { g++ DSA_ipod.cpp -o DSA_ipod } ; if ($?) { .\DSA_ipod }
GROUP 8 PROJECT
-----
X
GROUP MEMBERS
1. HARSH SHAH JAISWAL - 20BCE10556
2. ANKIT JINDAL - 20BCE10507
3. SIDDHI MANGLAM - 20BCE10550
4. SHANTANU KASANA - 20BCE10557
5. NIKHITA GARG - 20BCE10553
-----
Enter number of songs you want to insert into the new IPOD : 5
Enter song name : song a
Enter song name : song b
Enter song name : song d
Enter song name : song c
Enter song name : song e
Enter 1 to insert the song
Enter 2 to delete the song
Ln 17, Col 18 Spaces: 4 UTF-8 CRLF C++ Go Live Win32 ⚙
ENG IN 21-0

```

```
File Edit Selection View Go Run Terminal Help DSA_ipod.cpp - c++ by harsh - Visual Studio Code [Administr...]
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE + ▾
Enter 1 to insert the song
Enter 2 to delete the song
Enter 3 to play all songs
Enter -1 to exit code
3
PLAYING SONGS
-----
song a
song b
song d
song c
song e
-----
Enter 1 to insert the song
Enter 2 to delete the song
Enter 3 to play all songs
Enter -1 to exit code
1
Enter song name which you want to insert in IPOD : song 1
song 1 inserted successfully
-----x-----
```

```
Ln 0 Col 0 Spaces: 4 UTF-8 CRLF C++ Go Live Win32 ⚙ ENG IN 21-0
```

```
File Edit Selection View Go Run Terminal Help DSA_ipod.cpp - c++ by harsh - Visual Studio Code [Administr...]
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE + ▾
Enter song name which you want to insert in IPOD : song 1
song 1 inserted successfully
-----x-----
```

```
Enter 1 to insert the song
Enter 2 to delete the song
Enter 3 to play all songs
Enter -1 to exit code
1
Enter song name which you want to insert in IPOD : song 2
song 2 inserted successfully
-----x-----
```

```
Enter 1 to insert the song
Enter 2 to delete the song
Enter 3 to play all songs
Enter -1 to exit code
3
PLAYING SONGS
-----x-----
```

```
song 2
song 1
song a
song b
-----x-----
```

```
Ln 0 Col 0 Spaces: 4 UTF-8 CRLF C++ Go Live Win32 ⚙ ENG IN 21-0
```

song 1  
song a  
song b  
song d  
song c  
song e

-----x-----

Enter 1 to insert the song  
Enter 2 to delete the song  
Enter 3 to play all songs  
Enter -1 to exit code  
2  
Enter song number which you want to delete from IPOD : 1  
Song deleted successfully

-----x-----

Enter 1 to insert the song  
Enter 2 to delete the song  
Enter 3 to play all songs  
Enter -1 to exit code  
3  
PLAYING SONGS

Ln 17, Col 18 Spaces: 4 UTF-8 CRLF C++ ⚡ Go Live Win32 ⚡

File Edit Selection View Go Run Terminal Help DSA\_ipod.cpp - c++ by harsh - Visual Studio Code [Administr... — □

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE + ▾

powers Code

3  
PLAYING SONGS

-----x-----

song 1  
song a  
song b  
song d  
song c  
song e

-----x-----

Enter 1 to insert the song  
Enter 2 to delete the song  
Enter 3 to play all songs  
Enter -1 to exit code  
2  
Enter song number which you want to delete from IPOD : 6  
Song deleted successfully

-----x-----

Enter 1 to insert the song  
Enter 2 to delete the song  
Enter 3 to play all songs

Ln 17, Col 18 Spaces: 4 UTF-8 CRLF C++ ⚡ Go Live Win32 ⚡

File Edit Selection View Go Run Terminal Help DSA\_ipod.cpp - c++ by harsh - Visual Studio Code [Administr... — □

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE + ▾

powers Code

Song deleted successfully

Enter 1 to insert the song  
Enter 2 to delete the song  
Enter 3 to play all songs  
Enter -1 to exit code  
3

PLAYING SONGS

song 1  
song a  
song b  
song d  
song c

Enter 1 to insert the song  
Enter 2 to delete the song  
Enter 3 to play all songs  
Enter -1 to exit code  
4

Enter 1 or 2 or 3 or -1

Enter 1 to insert the song  
Enter 2 to delete the song  
Enter 3 to play all songs  
Enter -1 to exit code  
2

Enter song number which you want to delete from IPOD : 7

Song number is out of range

Enter 1 to insert the song  
Enter 2 to delete the song  
Enter 3 to play all songs  
Enter -1 to exit code  
3

PLAYING SONGS

song 1  
song a  
song b  
song d  
song c

File Edit Selection View Go Run Terminal Help DSA\_ipod.cpp - c++ by harsh - Visual Studio Code [Administr...]

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

Enter 1 to insert the song  
Enter 2 to delete the song  
Enter 3 to play all songs  
Enter -1 to exit code  
3  
PLAYING SONGS

song 1  
song a  
song b  
song d  
song c

Enter 1 to insert the song  
Enter 2 to delete the song  
Enter 3 to play all songs  
Enter -1 to exit code  
-1  
Thank you for playing songs  
PS D:\c++\Unit2programs> □

Ln 17, Col 18 Spaces: 4 UTF-8 CRLF C++ ⚙ Go Live Win32 ⚙

⊗ 0 Δ 0

# **THANK YOU**