

// Online C++ compiler to run C++ program online

```
#include <iostream>
```

```
using namespace std;
```

```
class Node{
```

```
public:
```

```
int data;
```

```
Node* next;
```

```
Node(int data){
```

```
    this->data=data;
```

```
    this->next=nullptr;
```

```
}
```

```
};
```

```
class Stack{
```

```
public:
```

```
Node* top;
```

```
Stack(){
```

```
    top=nullptr;
```

```
}
```

```
bool isEmpty(){
```

```
    return top==nullptr;
```

```
}
```

```
void Push(int data){
```

```
    Node* newnode=new Node(data);
```

```
    newnode->next=top;
```

```
    top=newnode;
```

```
    cout<<data<<"pushed on the stack"<<endl;
```

```
}
```

```
void Pop(){
```

```
    if(isEmpty()){
```

```
        cout<<"stack is empty"<<endl;
```

```
        return;
```

```
}
```

```
Node* temp=top;
```

```

        cout<<"deleted data"<<temp->data<<endl;
        top=top->next;
        delete temp;
    }

    int peek(){
        if(isEmpty()){
            cout<<"stack is empty"<<endl;
            return -1;
        }
        return top->data;
    }

    void printstack(){
        if(isEmpty()){
            cout<<"stack is empty"<<endl;
        }
        Node* current=top;

        while(current!=nullptr){
            cout<<current->data<<" ";
            current=current->next;
        }
        cout<<endl;
    }
};

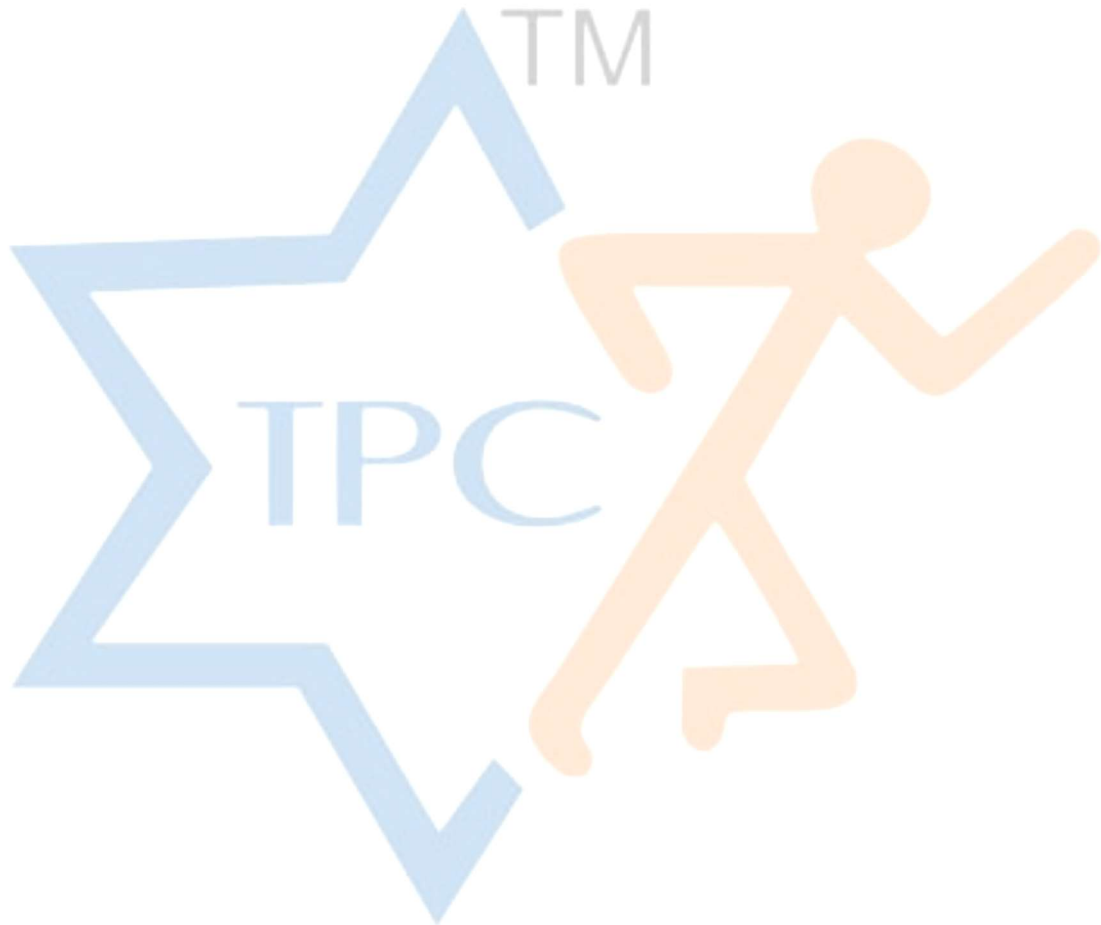
int main() {
    Stack S,S1;
    S.Push(20);
    S.Push(50);
    S.Push(60);
    S.Pop();
    cout<<"peek"<<S.peek()<<endl;
    S.printstack();
    while(!S.isEmpty()){
        int val=S.peek();
        S.Pop();
        S1.Push(val);
    }
}

```

```
S1.printstack();
```

```
return 0;
```

```
}
```



www.tpcglobal.in