

```

#include <iostream>
using namespace std;

template<class T>
class Stack
{
private:
    int top;
    T size,*p;

public:
    Stack(T a)
    {
        top=-1;
        size=a;
        p=new T[size];
    }

    bool push(T value)
    {
        if(top==size-1)
        {
            cout<<"stack overflow"<<endl;
            return 0;
        }
        else
        {
            p[++top]=value;
            return 1;
        }
    }

    T pop()
    {
        if(top==-1)
            cout<<"stack underflow"<<endl;
        else
        {
            T r = p[top--];
            return r;
        }
    }

    void display()
    {
        for(int i=top; i!=-1; i--)
        {
            cout<<p[i]<<"\t";
        }
        cout<<endl;
    }
};

int main()
{
    int option,size;
    cout<<"enter the size of your stack"<<endl;
    cin>>size;
    Stack<int> a(size);

    while(true)
    {

```

```

        cout<<endl<<"enter the options"<<endl<<"1.to enter data"<<endl<<"2.to
display stack"<<endl<<"3.to remove last entry"<<endl<<"4.to exit"<<endl;
        cout<<"enter your choice"<<endl;
        cin>>option;

        switch (option)
        {
        case 1:int value;
            cout<<endl<<"enter the value";
            cin>>value;
            a.push(value);
            break;

        case 2:a.display();
            break;

        case 3:int pop;
            pop =a.pop();
            cout<<pop<<"is removed"<<endl;
            break;

        case 4:cout<<"exited"<<endl;
            return 0;

        default:cout<<endl<<"try again"<<endl;

        }
    }
    return 0;
}

```