1. WAP to perform Stack and Queue

**Stack**

#include <iostream>

using namespace std;

int stack[100], n=100, top=-1;

void push(int val) {

   if(top>=n-1)

   cout<<"Stack Overflow"<<endl;

   else {

      top++;

      stack[top]=val;

   }

}

void pop() {

   if(top<=-1)

   cout<<"Stack Underflow"<<endl;

   else {

      cout<<"The popped element is "<< stack[top] <<endl;

      top--;

   }

}

void display() {

   if(top>=0) {

      cout<<"Stack elements are:";

      for(int i=top; i>=0; i--)

      cout<<stack[i]<<" ";

      cout<<endl;

   } else

   cout<<"Stack is empty";

}

int main() {

   int ch, val;

   cout<<"1) Push in stack"<<endl;

   cout<<"2) Pop from stack"<<endl;

   cout<<"3) Display stack"<<endl;

   cout<<"4) Exit"<<endl;

   do {

      cout<<"Enter choice: "<<endl;

      cin>>ch;

      switch(ch) {

         case 1: {

            cout<<"Enter value to be pushed:"<<endl;

            cin>>val;

            push(val);

            break;

         }

         case 2: {

            pop();

            break;

         }

         case 3: {

            display();

            break;

         }

         case 4: {

            cout<<"Exit"<<endl;

            break;

         }

         default: {

            cout<<"Invalid Choice"<<endl;

         }

      }

   }while(ch!=4);

   return 0;

}

**Queue**

#include <iostream>

using namespace std;

int queue[100], n = 100, front = - 1, rear = - 1;

void Insert() {

   int val;

   if (rear == n - 1)

   cout<<"Queue Overflow"<<endl;

   else {

      if (front == - 1)

      front = 0;

      cout<<"Insert the element in queue : "<<endl;

      cin>>val;

      rear++;

      queue[rear] = val;

   }

}

void Delete() {

   if (front == - 1 || front > rear) {

      cout<<"Queue Underflow ";

      return ;

   } else {

      cout<<"Element deleted from queue is : "<< queue[front] <<endl;

      front++;;

   }

}

void Display() {

   if (front == - 1)

   cout<<"Queue is empty"<<endl;

   else {

      cout<<"Queue elements are : ";

      for (int i = front; i <= rear; i++)

      cout<<queue[i]<<" ";

         cout<<endl;

   }

}

int main() {

   int ch;

   cout<<"1) Insert element to queue"<<endl;

   cout<<"2) Delete element from queue"<<endl;

   cout<<"3) Display all the elements of queue"<<endl;

   cout<<"4) Exit"<<endl;

   do {

      cout<<"Enter your choice : "<<endl;

      cin>>ch;

      switch (ch) {

         case 1: Insert();

         break;

         case 2: Delete();

         break;

         case 3: Display();

         break;

         case 4: cout<<"Exit"<<endl;

         break;

         default: cout<<"Invalid choice"<<endl;

      }

   } while(ch!=4);

   return 0;

}