stack

```
#include<iostream>
using namespace std;
const int x = 5;
int stack[x];
int top = -1;
bool empty()
{
	if (top == -1)
	{
		cout << "  \n The stack is empty : ";
		return true;
	}
	else
	{
		cout << " \n The stack is not empty : ";
		return false;
	}
}
void push(int newitem)
{
	if (top == x - 1)
	{
		cout << "\n stack is full \n";
	}
	else
	{
		top++;
		stack[top] = newitem;
	}
}
int pop()
{
	if (top == -1)
	{
		cout << " stack is empty \n";
	}
	else
	{

		top--;

		return stack[top];
	}
}
int peek()
{
	if (top == -1)
	{
		cout << " stack is empty"<<endl;
	}
	else
	{
```

```
                return stack[top];
        }
}
void display()
{
        if (top == -1)
        {
                cout << " stack is empty \n";
        }
        else
        {
                cout << " The value is item =";
                for (int i = top; i >= 0; i--)
                {
                        cout << stack[i] << " ";
                }
        }
}
int main()
{

        push(51);
        push(60);
        push(70);
        push(80);
        push(90);
        display();
        pop();
        cout << " \n item peek = " << stack[top + 1]; peek();
    display();
        empty();
        system("pause");
        return 0;
}
```

Queue

```
#include<iostream>
using namespace std;
const int si = 5;
int quaue[si];
int r = -1, f = -1;
void enquaue(int newitem)
{
	if (r == si - 1 && f == 0 || f == r + 1)
	{
		cout << "Quaue is full \n";
	}
	else
	{
		if (r == -1 && f == -1)
		{
			r++;
			f++;
			quaue[r] = newitem;
		}
		else
		{
			r++;
			quaue[r] = newitem;
		}
	}
}
int dequaue()
{
	if (r == -1 && f == -1 || f>r)
	{
		cout << "Quaue is empty \n";
	}
	else
	{

		f++;
		return quaue[f];
	}
}
bool empty()
{
	if (r == -1 && f == -1 || f>r)
	{
		cout << "Quaue is empty : ";
		return true;
	}
	else
	{
		cout << " Quaue is not empty : ";
```

```
                return false;
        }
}
int peek()
{
        if (r == -1 && f == -1 || f>r)
        {
                cout << "Quaue is empty \n";
        }
        else
        {

                return quaue[f];
        }
}
int rear()
{
        if (r == si - 1 && f == 0 || f == r + 1)
        {
                cout << "Quaue is full \n";
        }
        else
        {
                cout << "\n the item is rear : "<< quaue[r];
                return quaue[r];
        }
}
void display()
{
        if (r == -1 && f == -1 || f>r)
        {
                cout << " Quaue is empty \n";
        }
        else
        {
                cout << "\n The value is item = ";
                for (int i = f; i <= r; i++)
                {
                        cout << quaue[i] << " ";
                }
        }
}
int main()
{
        /*int nam;
        cin >> nam;
        enquaue(nam);*/
        enquaue(10);
        enquaue(20);
        enquaue(30);
        enquaue(40);
        enquaue(50);
        display();
        dequaue();
        cout << "\n the item is peek : " << quaue[f - 1]; peek();
        rear();
        display(); cout << endl;
```

```
        empty();
        system("pause");
        return 0;
}
```