2/1/2021

Q. Find maximum in O(1) ?

=> Given stack is

2
5
1.
3
64
72 - - -> Maximum

So, output must be 64 when getMax() is called.

Program:-

```
# include <stdio.h>
# include < conio.h>
void main()
using namespace std;
struct Mystack
{
    stack < int > s;
    int max ele;
    void get Max()
    {
        if (s.empty())
            cout << "Stack is empty\n";
        else
            cout << "Maximum element in the
                          stack is : "
                  << max ele << "\n";

    }
    void peak()
     {
        if(s.empty())
        {
            cout << "Stack is empty";
            return;
        }
    int t = s.top();
    cout << "Top most element is:";
    (t > max ele) ? cout << max ele : cout << t;
}

void pop()
{
    if (s.empty())
    {
        cout << "Stack is empty\n";
        return;
    }
    cout << "Top most element removed:";
    int t = s.top();
    S.pop();

    if (t > max ele)
    {
        cout << max ele << "\n";
        max ele = 2* max Ele - t;
    }
    else
        cout << t << "\n";
}

void push (int x)
{
    if (s.empty())
    {
        max Ele = x;
        s.push(x);
        cout << "No. inserted: "<<x<<"\n";
        return;
    }
    if (x> max Ele)
    {
        s.push(2* x- max Ele);
        max Ele = x;
    }
    else
        s.push(x);
    cout <<"No inserted:"<<x<<"\n";
    }
};

int main()
{
My stack s;
s.push(3);
s.push(5);
s.get Max();
s.push(7);
s.push(19);
s.get Max();
```

```
        S. pop ();
        S. getMax ();
        S. pop ();
        S. peek ();

        return 0;

}.
```

O/P.

```
Number inserted : 3
Number inserted : 5
Maximum element in Stack is : 5
Number inserted : 7.
Number inserted : 19
Maximum Element in stack is : 19.
Top Most Element Removed : 19
Maximum Element in stack is : 7.
Top Most Element Removed : 7
Top Most Element is : 5
```