**Recursion -**

1 - Basic Understanding of Recursion -

2 - Factorial Program using function

3 - Sum of N- Natural numbers using recursion

4 - n power of x number

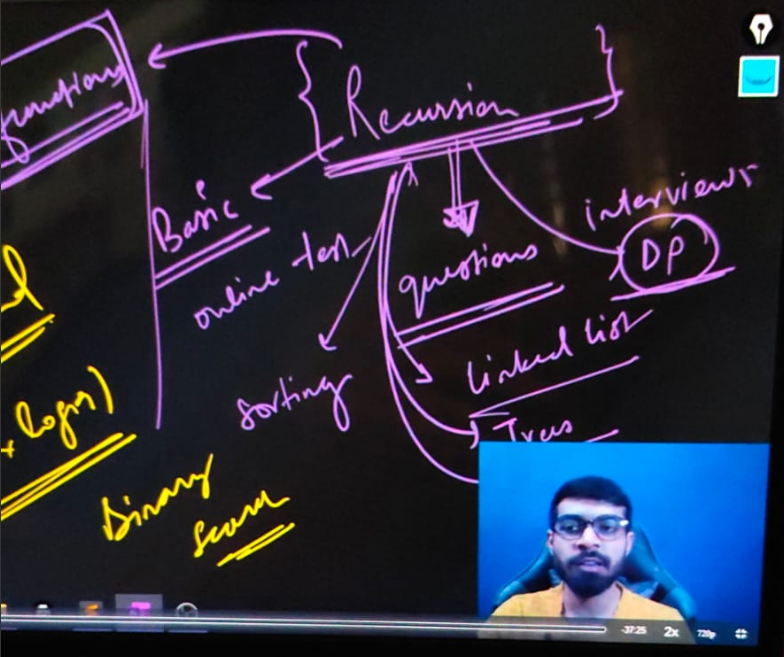
5 - Sum of elements of an array –

5.1 - Mul of elements of an array -

6 - Odd No. from n to m using recursion.

**Recursion in CPP –**

What after learning Recursion -

****

#include <bits/stdc++.h>

// #include <iostream>

// #include<algorithm>

// #include<climits>

// #include<string>

// #include<cctype>

using namespace std;

**//Qun 1 - Basic Understanding of Recursion -**

// void welcome(int n)

// {

//     if (n==0)

//     {

//         return;

//     }//if not use return to anything or any coondition then the funcaiton calling will be conutinuoes for the infinite times

//     cout<<"Hii, Future version of Shubham Mahajan, is going to placed in Microsoft which opens new Corporate office in Pune "<<endl;

//     welcome(n-1);

// }

// int main()

// {

//     // Recursion - Re-Occur. Function calling again and again. Function calling itself

//     welcome(10); // - if we'll not pass the arguements then it'll be running for the infinite time, so passing arguements for how many times .

// /\*

// Hii, Future version of Shubham Mahajan, is going to placed in Microsoft which opens new Corporate office in Pune

// Hii, Future version of Shubham Mahajan, is going to placed in Microsoft which opens new Corporate office in Pune

// Hii, Future version of Shubham Mahajan, is going to placed in Microsoft which opens new Corporate office in Pune

// Hii, Future version of Shubham Mahajan, is going to placed in Microsoft which opens new Corporate office in Pune

// Hii, Future version of Shubham Mahajan, is going to placed in Microsoft which opens new Corporate office in Pune

// Hii, Future version of Shubham Mahajan, is going to placed in Microsoft which opens new Corporate office in Pune

// Hii, Future version of Shubham Mahajan, is going to placed in Microsoft which opens new Corporate office in Pune

// Hii, Future version of Shubham Mahajan, is going to placed in Microsoft which opens new Corporate office in Pune

// Hii, Future version of Shubham Mahajan, is going to placed in Microsoft which opens new Corporate office in Pune

// Hii, Future version of Shubham Mahajan, is going to placed in Microsoft which opens new Corporate office in Pune

// \*/

// }

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

// void welcome(int n)

// {

//     if (n==0)

//     {

//         return;

//     }

//     cout<<"n = "<<n<<endl;

//     welcome(n-1);// Responsible for Fucniton Calling

// }

// int main()

// {

//     welcome(10);

// }

// /\*

// n = 10

// n = 9

// n = 8

// n = 7

// n = 6

// n = 5

// n = 4

// n = 3

// n = 2

// n = 1

// \*/

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

// void welcom2(int n)

// {

//     if (n==0)

//     {

//         return;

//     }

//     welcom2(n-1);// Responsible for Fucniton Calling - yaha kya hua ki function print hone ke phle hi call ho gya, to jb 10 se 1 tk call hua phr condition return ho gyi. ab print pr aayagea

//     cout<<"n = "<<n<<endl;

// }

// int main()

// {

//     welcom2(10);

// }

// /\*

// n = 1

// n = 2

// n = 3

// n = 4

// n = 5

// n = 6

// n = 7

// n = 8

// n = 9

// n = 10

// \*/

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**//Qun 2 -  Factorial Program using funciton**

// int factorial(int n)

// {

//     if (n==0)

//     {

//         return 1;

//     }

//     return n\*factorial(n-1);

// }

// int main()

// {

//     int n;

//     cout<<"Write the number - "<<endl;

//     cin>>n;

//     cout<<factorial(n);

// }

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// Recursion Understanding -**

// void abcd(int num)

// {

//     if (num == 0)

//     {

//         return;

//     }

//     cout << num << endl;

//     abcd(num - 1);

// }

// int main()

// {

//     abcd(5);

// }

// /\*

// 5

// 4

// 3

// 2

// 1

// \*/

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

// void abcd2(int num)

// {

//     if (num == 0)

//     {

//         return;

//     }

//     abcd2(num - 1);

//     cout << num << endl;

// }

// int main()

// {

//     abcd2(5);

// }

// /\*

// 1

// 2

// 3

// 4

// 5

// \*/

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

// void abcd2(int num)

// {

//     cout << num << endl;

//     if (num == 10)

//     {

//         return;

//     }

//     abcd2(num + 1);

// }

// int main()

// {

//     abcd2(0);

// }

// /\*

// 0

// 1

// 2

// 3

// 4

// 5

// 6

// 7

// 8

// 9

// 10

// \*/

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// Qun 3 - Sum of N- Natural numbers using recursion -**

// int sum(int n)

// {

//     if (n==1)

//     {

//         return 1;

//     }

//     return n+sum(n-1);

// }

// int main()

// {

//     cout<<sum(5);

// }

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// Qun 4 - n power of x number -**

// Simple method -

// int main()

// {

//     int n,p;

//     cout<<"values - "<<endl;

//     cin>>n>>p;

//     cout<<pow(n,p);

// }

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// using recursion-**

// int power(int x, int n)

// {

//     if (n == 0)

//     {

//         return 1;

//     }

//     return x \* power(x, n - 1);

// }

// int main()

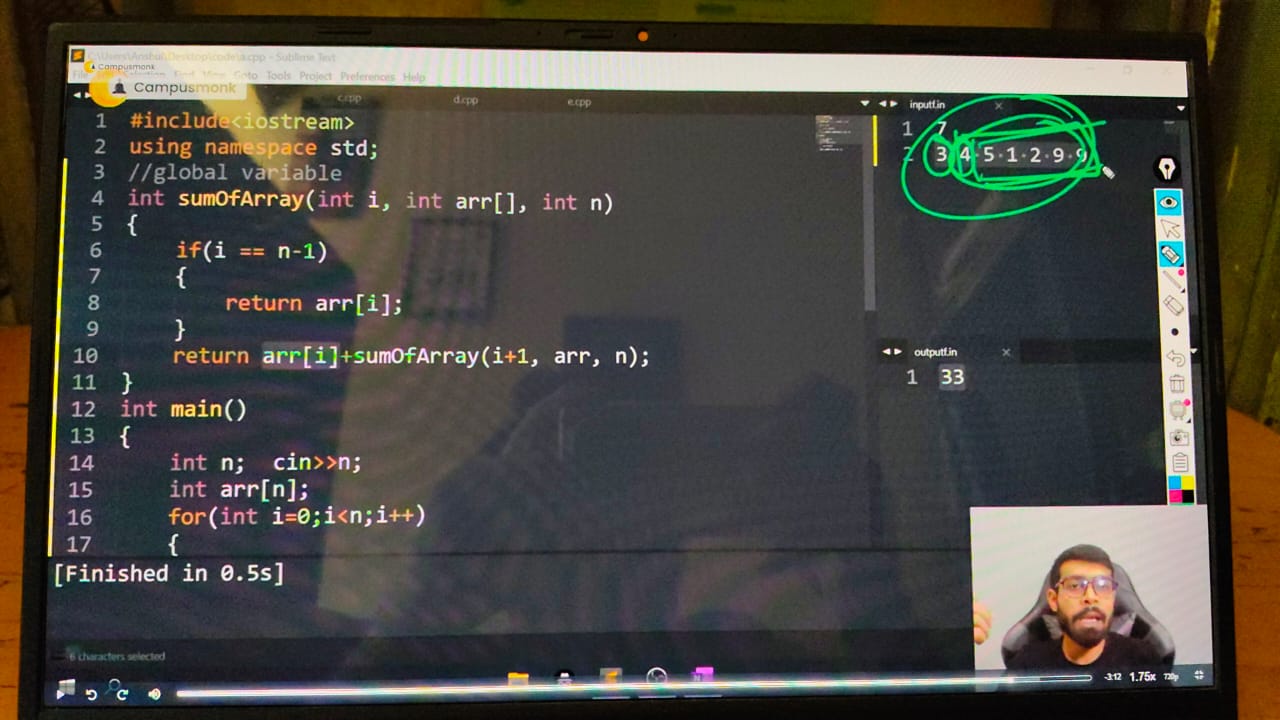
// {

//     cout<<power(3,3);

// }

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**//Qun 5 - Sum of elements of an array –**



// Global variable

// int MultiplcnofArray(int i, int arr[], int n)

// {

// // Base Condition

//     if (i==n-1)

//     {

//         return arr[i];

//     }

//     return arr[i]+MultiplcnofArray(i+1,arr,n);

// }

// int main()

// {

//     int n;

//     cout<<"Size of the array is - "<<endl;

//     cin>>n;

//     int arr[n];

//     cout<<"Enter the array elements - "<<endl;

//     for (int i = 0; i < n; i++)

//     {

//         cin>>arr[i];

//     }

//     cout<<MultiplcnofArray(0,arr, 7);

// }

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**//Qun 5.1 - Mul of elements of an array -**

// Global variable

// int MultiplcnofArray(int i, int arr[], int n)

// {

// // Base Condition

//     if (i==n-1)

//     {

//         return arr[i];

//     }

//     return arr[i]\*MultiplcnofArray(i+1,arr,n);

// }

// int main()

// {

//     int n;

//     cout<<"Size of the array is - "<<endl;

//     cin>>n;

//     int arr[n];

//     cout<<"Enter the array elements - "<<endl;

//     for (int i = 0; i < n; i++)

//     {

//         cin>>arr[i];

//     }

//     cout<<MultiplcnofArray(0,arr, n);

// }

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**//Qun 6 - Odd No. from n to m using recursion.**

// void Oddnumber(int n, int m)

// {

//     if (n > m)

//     {

//         return;

//     }

//     if (n % 2 != 0)

//     {

//         cout << n << endl;

//     }

//     return Oddnumber(n + 1, m);

// }

// int main()

// {

//     int n; int m;

//     cout<<"Enter the range"<<endl;

//     cin>>n>>m;

//     Oddnumber(n, m);

//     return 0;

// }

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_