Q1.: Predict the output each of the following codes without running the program, then compare your predicted result with actual program output (run the program).

Code #1A:

```
#include <iostream>
using namespace std;
int x = 40;
void myFunc(){
  cout<<"X2= "<<x<<endl;
  x=10;
  cout<<"X3= "<<x<<endl;
}

int main(){
  cout<<"X1= "<<x<<endl;
  myFunc();
  cout<<"X4= "<<x<<endl;

  return 0;
}
```

Code #1B:

```
#include <iostream>
using namespace std;
int x = 40;
void myFunc(){
  cout<<"X2= "<<x<<endl;
  int x=10;
  cout<<"X3= "<<x<<endl;
}

int main(){
  cout<<"X1= "<<x<<endl;
  myFunc();
  cout<<"X4= "<<x<<endl;

  return 0;
}
```

Code #2:

```
#include <iostream>
using namespace std;
void swap_func(int &a, int &b){
    int temp;
    temp = a;
    a = b;
    b = temp;
}
int main(){

    int v = 5, x = 10;
    cout << "v= "<<v << ", x= " << x << endl;
    swap_func(v,x);
    cout << "v= "<<v << ", x= " << x << endl;

    return 0;
}
```

Code #3:

```
#include <iostream>
using namespace std;

void myFunc(int &a, int &b){
    a=-10;
    b=-19;
}
int main() {

    int x=12, y=5;
    cout<<"x before calling = "<<x<<endl;
    cout<<"y before calling = "<<y<<endl;
    myFunc(x,y);
    cout<<"x before calling = "<<x<<endl;
    cout<<"y before calling = "<<y<<endl;

    return 0;
}
```

Code #4:

```cpp
#include <iostream>
using namespace std;

int x = 5;

void myFunction() {
  int x = 22;

  cout <<"Value of x in myFunction= "<< x << "\n";
}

int main() {
  myFunction();

  cout << "Value of x in main= "<<x·
  return 0;
}
```

Code #5:

```cpp
#include<iostream>
using namespace std;

int global = 5;

void display() {
    cout << global << end
}

int main() {

    display();

    global = 10;
    display();

    return 0;
}
```

Code #6:

```cpp
#include <iostream>
using namespace std;

int counter = -1;

int main() {

    int num = 0;
    do {
        num += 2;
        counter++;
    } while (num < 10);

    cout << "Counter: " << counter << endl;
    return 0;
}
```

Code #7:

```cpp
#include <iostream>
using namespace std;

void fun(int* ptr) {
    *ptr = 30;
}

int main() {
    int y = 20;
    fun(&y);
    cout << y << endl;

    return 0;
}
```

Code #8:

```cpp
#include <iostream>
using namespace std;

int main() {
    int* ptr;
    int x;

    ptr = &x;
    *ptr = 0;

    cout << "x = " << x << endl;
    cout << "*ptr = " << *ptr << endl;

    *ptr += 5;
    cout << "x = " << x << endl;
    cout << "*ptr = " << *ptr << endl;

    (*ptr)++;
    cout << "x = " << x << endl;
    cout << "*ptr = " << *ptr << endl;

    return 0;
}
```

Code #9:

```cpp
#include <iostream>
using namespace std;

int main()
{
    float arr[5] = {12.5, 10.0, 13.5, 90.5, 0.5};
    float *ptr1 = &arr[0];
    float *ptr2 = ptr1 + 3;

    cout << *ptr2 << " ";
    cout << (ptr2 - ptr1);

    return 0;
}
```

Code #11:

```cpp
#include <iostream>
using namespace std;

int x = 5;

void change() {
    int* p = &x;
    *p = 10;
}

int main() {
    change();
    cout << x;
    return 0;
}
```

Code #12:

```cpp
#include <iostream>
using namespace std;

int x = 3;

void update(int* p) {
    *p = *p + 4;
}

int main() {
    update(&x);
    cout << x;
}
```

Code #13:

```cpp
#include <iostream>
using namespace std;

int g = 4;

void test() {
    int* ptr = &g;
    cout << (*ptr)++;
}

int main() {
    test();
    cout << g;
}
```

Code #14:

```cpp
#include <iostream>
using namespace std;

int x = 9;

void fun() {
    int x = 0;
    int* p = &x;
    *p = 1;
}

int main() {
    fun();
    cout << x;
}
```

Code #15:

```cpp
#include <iostream>
using namespace std;

int y = 2;

void inc() {
    int* ptr = &y;
    *ptr = *ptr + *ptr;
}

int main() {
    inc();
    cout << y;
}
```

Code #16:

Q2.: find the maximum numbers in the following array, using function.
myArr={8, 2, 5, 1, 7, 4, 9, 3}

Q3.: find the minimum and maximum numbers in the following array, using pointer and function. myArr={8, 2, 5, 1, 7, 4, 9, 3}

Q4.: Find how many even and odd numbers we have in the following array, using pointer and function. myArr={8, 2, 5, 1, 7, 4, 9, 3}

Q5.: Find how many positive and negative numbers we have in the following array, using pointer and function. myArr={8.3, -4.2, 5, -0.1, 0.7, 49., -2.9, 5.3}

Q6.: Write a C++ program that prompts the user to input 4 double values. Use pointers to calculate the sum of the elements at even indices and the sum of the elements at odd indices. Then, compare the two sums and display which one is larger.