S1) WAP in C++ to find the roots of quadratic eqn

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    int a=6, b=-17, c=12;
    float det = b*b - 4*a*c;
    if (det >0) {
        float root1, root2;
        root1 = (-b + sqrt(det)) / (2*a);
        root2 = (-b - sqrt(det)) / (2*a);
        cout << "Roots are: " << root1 << " and " << root2;
    }
    else if (det==0) {
        float root1, root2;
        root1 = root2 = -b/(2*a);
        cout << "Roots are: " << root1 << " and " << root2;
    }
    else {
        cout << "Roots are imaginary";
    }
    return 0;
}
```

Output:-

Roots are: 1.5 and 1.33333

8) Write to find biggest element from an array

```
#include <iostream>
using namespace std;
int main(){
    int arr[] = {1,2,3,4,6,8};
    int arr_length= sizeof(arr)/sizeof(arr[0]);
    int max= arr[0];
    for (int i=0; i<arr_length; i++){
        if (arr[i] > max){
            max =arr[i];
        }
    }
    cout << "Max= " << max;
    return 0;
}
```

Output:

Max=8

Q3) Define a structure with proper data members. Input and display the details of a student. (3)

```
#include <iostream>
#include <cmath>
struct student{
    int roll;
    char name[20];
    float marks;
} s[2];
int main(){
    for (int i=0; i<2; ++i){
        s[i].roll = i+1;
        cout<< "\n"Enter name \n";
        cin >> s[i].name;
        cout << "Enter marks: ";
        cin >> s[i].marks;
    }
    for(int i=0; i<2; ++i){
        cout << "\nRoll number: << i+1 << "\n";
        cout << "First name: ";
        puts(s[i].name);
        cout << "Marks: << s[i].marks;
        cout << "\n";
    }
    return 0;
}
```

Output:
Roll no. 1
First name: entered_name
Marks : 100
Roll no: 2
First name: name2
Marks : 99

Input:
Enter name:
Enter marks:

LAB - 2 : Assignments (4)

Q1. Write a program to find size of fundamental data types in C++ language.

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

int main() {
    int a;
    char b;
    float c;
    cout<<"Size of int a: "<<sizeof(a)<<"\n";
    cout<<"Size of char b: "<<sizeof(b)<<"\n";
    cout<<"Size of float c: "<<sizeof(c)<<"\n";

    return 0;
}
```

Output:

```
Size of int a: 4
Size of char b: 1
Size of float c: 4
```

Q2 Write a program (5)
to check whether a number is prime or not.

```
#include<iostream>
using namespace std;
int main(){
    cout<<"Enter the number: ";
    int n, flag=1;
    cin >>n;
    if (n==1){
        cout <<"1 is neither prime nor compusite\n";
        return 0;
    }
    for(int i=2; i<n; i++){
        if (n%i==0){
            flag =0;
            cout <<"Not Prime";
            break;
        }
    }
    if (flag==1){
        cout <<"Prime";
    }
    return 0;
}
```

Output:
Enter the number: 8
Not Prime

Q3) Write a program to demonstrate the concept of call-by-value, call-by-reference & call-by-address. (6)

```
#include <iostream>
using namespace std;
void swap(int a, int b){
    int temp=a;
    a=b;
    b=temp;
}
void swap2(int &a, int &b){
    int temp=a;
    a=b;
    b=temp;
}
void swap3(int *a, int *b){
    int temp=*a;
    *a=*b;
    *b=temp;
}
int main(){
    int a,b;
    cout<<"Enter 1st number: ";
    cin>>a;
    cout<<"Enter 2nd number: ";
    cin>>b;
    cout<<"Before swapping\n"<<a<<"\t"<<b<<"\n";
    swap(a,b);
    cout<<"After swapping using value\n"<<a<<"\t"
        <<b<<"\n";
    swap2(a,b);
    cout<<"After swapping using reference\n"<<a<<"\t"<<b;
    swap3(a,b);
    cout<<"After swapping using address"<<a<<"\t"<<b;
    return 0;
}
```

6.1

Output:

```
Enter 1st number: 2
Enter 2nd number: 4
Before swapping
2 4
After swapping using value
2 4
After swapping using reference
4 2
After swapping using address
2 4
```

Q) WAP to find out the area of a circle & volume of a sphere by using function overloading.

```
#include<iostream>
using namespace std;
void result(int r){
    float area= 3.14 * r *r;
    cout<< "Area of the circle= "<<area;
}
void result(float r) {
    float vol=(4 * 3.14 * r*r*r)/3;
    cout << "Volume of the sphere=" << vol;
int main() {
    int r;
    cout<<"\nEnter radius of the circle: ";
    cin >>r;
    result(r);
    cout <<"\nEnter radius of the sphere: ";
    cin >>r;
    float rad= (float)r;
    result(rad);
    return 0;
}
```

Output:

```
Enter radius of the circle: 5
Area of the circle= 78.5
Enter radius of the sphere:5
Volume of the sphere= 523.333
```

(Q5) WAP to find out area or volume of an shape/object by using one function named as FUN-AREA only. If the function takes an argument then calc. area of circle, two arguments then area of rectangle & 3 for volume of box. (2)

```
#include<iostream>
using namespace std;
void fun_area(float r){
    float area= 3.14*r*r;
    cout<<"Area of the circle = "<<area;
}
Void fun_area(float l,float b){
    float area=l*b;
    cout<<"Area of the rectangle="<<area;
}
Void fun_area(float l,float b,float h){
    float vol=l*b*h;
    cout<<"Volume of the box="<<vol;
}
int main(){
    int n;
    cout<<"How many arguments you want to pass:";
    cin>>n;
    if(n==0){
        cout<<"Nothing to do";
    }
    else if(n==1){
        float r;
        cout<<"Enter radius:";
        cin>>r;
        fun_area(r);
    }
```

~~else~~

```
    else if (n==2){
        float l,b;
        cout << "Enter length and breadth: ";
        cin >> l >> b;
        fun_area(l,b);
    }
    else if(n==3){
        float l,b,h;
        cout << "Enter length, breadth and
                height: ";
        cin >> l >> b >> h;
        fun_area(l,b,h);
    }
    return 0;
}
```

Output:

```
How many arguments you want to pass: 2
Enter length and breadth: 5 10
Area of the rectangle=50
```

86) WAP to find summation of three numbers by using one function only with function name sum having three arguments. If at runtime one argument is given to the function sum, then 2nd & 3rd argument will be assumed by default as 10 & 20 respectively. Use function with default argument concepts. (9)

```
# include <iostream>
using namespace std;
void sum(int a, int b=10, int c=20){
        int sum= a+b+c;
        cout << "Sum = " << sum;
}
int main(){
    int a,b,c;
    int n;
    cout << "How many arguments to pass: ";
    cin >>n;
    if (n==1){
        cin >>a;
        sum(a);
    } else if (n==2){
        cin >> a >> b;
        sum(a, b);
    } else if (n==3){
        cin>>a>>b>>c;
        sum(a,b,c);
    }
    return 0;
}
```

Output:

```
How many arguments to pass: 2
5 6
Sum= 31
```