

```
#include <iostream> // to use cin/cout
#include <math.h> // for function "sqrt"
```

```
using namespace std;
```

```
double getroot(double X)
```

```
{
```

```
    // next line finds root of number by standart function in c++
```

```
    // if you can use standart functions, just uncomment next line ans delete other code in function
```

```
    //// return sqrt(X); // uses standart library "math.h"
```

```
    //finding root using binsearch
```

```
    double l = 0, r = X, mid;
```

```
    while(r-l > 0.0000001)
```

```
    {
```

```
        mid = (l+r) / 2.0;
```

```
        if(mid*mid < X)
```

```
            l = mid;
```

```
        else
```

```
            r = mid;
```

```
    }
```

```
    return l;
```

```
}
```

```
int main()
```

```
{
```

```
    double X;
```

```
do
{
    // write information for user
    cout << "Enter the number(for end of program enter negative number): ";

    // read number
    cin >> X;

    // call our function
    if(X >= 0)
        cout << "Root of a number is: " << getroot(X) << "\n";
}
while(X >= 0);

return 0;
}
```

```
#include <iostream> // to use cin/cout
#include <math.h> // for function "sqrt"
```

```
using namespace std;
```

```
double getroot(double X)
{
    // next line finds root of number by standart function in c++
    // if you can use standart functions, just uncomment next line ans delete other code in function

    //// return sqrt(X); // uses standart library "math.h"

    //finding root using binsearch
    double l = 0, r = X, mid;
    while(r-l > 0.0000001)
    {
        mid = (l+r) / 2.0;
        if(mid*mid < X)
            l = mid;
        else
            r = mid;
    }
    return l;
}
```

```
int main()
```

```

{

    cout << "A*x^2 + B*x + C = 0\n\n";

    double A, B, C, D, x1, x2, im_x1, im_x2;


    do

    {

        // write information for user

        cout << "\nEnter coefficients A, B, C(for end of program enter coefficient A equals to
zero):\n";


        // read coefficients

        cin >> A >> B >> C;


        // let D = B*B - 4*A*C

        // we know that roots x1, x2 = (-B +/- sqrt(D)) / (2*A)

        // so, if D < 0 - there are imaginary roots

        //          if D == 0 - we have one real root

        //   and if D > 0 - we have two real roots


        if(A != 0)

        {

            D = B*B - 4.0*A*C;


            cout.precision(2);

            if(D < 0)

            {

                // imaginary roots

```

```

        cout << "There are imaginary roots\n";

x1 = -B/(2*A);    //real part of first root
im_x1 = -getroot(-D) / (2*A); // imaginary part of first root

x2 = -B/(2*A);    //real part of second root
im_x2 = getroot(-D) / (2*A); // imaginary part of second root


cout << x1 << " + i*(" << im_x1 << ")\n";
cout << x2 << " + i*(" << im_x2 << ")\n";
    }
    else if(D == 0)
    {
// one real root

        cout << "There is one real root\n";

x1 = B / (-2.0*A);

        cout << fixed << x1 << "\n";
    }
    else
    {
// two real roots

        cout << "There are two real roots\n";

x1 = (-B - getroot(D)) / (2.0*A);
x2 = (-B + getroot(D)) / (2.0*A);

        cout << fixed << x1 << " " << x2 << "\n";
    }
}

```

```
        }  
    }  
    while(A != 0);  
  
    return 0;  
}
```

