**Part A:**

You must write a program that receives polynomial coefficients of polynomial ***p5(x)*** of level 5 with a positive coefficient ***x5*** and finds one real root (which always exists) of the polynomial.  
Use one of the iterative methods of finding a nonlinear function root, The Newton and Bisection methods have been tested and known to work.  
In any case, a calculation of "initial guesses" should be made. In this case, this can be done by checking ***t1= 2k***, **k=0,1, 2 …** until we find a **k1** which results ***p5(t1(k1)) > 0,*** *and* ***t2= -2k***, **k=0,1, 2 …** which results ***p5(t2(k2)) < 0.****Then, either take the*  ***(t1,t2)*** *As a starting area of ​​Algorithm for the* Bisection method, Or ***(t1+t2)/2*** *As a starting point of the Newton method.*

*Example:*

***P(X) = 720x5 – 612 x4 - 500 x3 + 449 x2 + 52 x – 60***

*Then the output can be:*

**Enter 6 ploy5 coefficients, a5 must be > 0:**

**720.0 -612.0 -500.0 449.0 52.0 -60.0**

**a5 = 720.000**

**a4 = -612.000**

**a3 = -500.000**

**a2 = 449.000**

**a1 = 52.000**

**a0 = -60.000**

**ploy5(0.833333) = 0.000000**

Hence one real root is: **x\* = 0.83333 = 5/6**

**Part B:**

You must write a program that receives polynomial ***P5(X)*** (rank 5) coefficients and some real root **r**And narrows it down to a level 4 polynomial by long division.

Example:  
***(720 x5 – 612 x4 - 500 x3 + 449 x2 + 52 x – 60) / (x-0.8333) = 720*** ***x4 -12 x3*** ***- 510 x2*** + ***24 x – 72***

*\*Note that you know that this is exactly a Level 5 polynomial.*

*Possible output of your program would be:*

**Enter 6 ploy5 coefficients:**

**720.0 -612.0 -500.0 449.0 52.0 -60.0**

**Enter solution:**

**0.83333**

**a5 = 720.000**

**a4 = -612.000**

**a3 = -500.000**

**a2 = 449.000**

**a1 = 52.000**

**a0 = -60.000**

**xstar = 0.833330**

**Divided polynom:**

**b4 = 720.000**

**b3 = -12.002**

**b2 = -510.002**

**b1 = 24.000**

**a0 = 72.000**

**Part C:**

You need to implement a program that receives some level 5 polynomial and prints all of its roots, including the complexes.

*Possible output of your program would be:*

**Enter 6 coefs, first must be > 0:**

**1 -1 -7 7 -18 18**

**a5 = 1.000**

**a4 = -1.000**

**a3 = -7.000**

**a2 = 7.000**

**a1 = -18.000**

**a0 = 18.000**

**Solutions are:**

**1.000000 + 0.000000 i**

**-0.000000 + -1.414214 i**

**0.000000 + 1.414214 i**

**3.000000 + 0.000000 i**

**-3.000000 + -0.000000 i**

**Part D:**

You need to execute a program that receives any polynomial of Level 6 and prints all of its actual minimum points.

Solution Method:

Take the program of Part C and call it with the derived polynomial of the given polynomial, check the original polynomial values ​​of the roots of the derived polynomial, and declare the minimum value and the points receiving it.

For example, possible outputs of the program are:

**Enter 7 coefs, first must be > 0:**

**1.0 0.0 -12.0 0.0 23 0.0 36**

**org6 = 1.000000  
org5 = 0.000  
org4 = -12.000  
org3 = 0.000  
org2 = 23.000  
org1 = 0.000  
org0 = 36.000  
a5 = 6.000  
a4 = 0.00  
a3 = -48.000  
a2 = 0.000  
a1 = 46.000  
a0 = 0.000**

**Equation: 1.000000 x\*\*4 + 0.000000 x\*\*3 + -8.000000 x\*\*2 + 0.000000 x + 7.666667**

**Solutions are:**

**0.000000 + 0.000000 i  
1.055106 + 0.000000 i  
-1.055106 + -0.000000 i  
2.624262 + 0.000000 i  
-2.624262 + -0.000000 i**

**p6(0.000000) = 36.000000  
p6(1.055106) = 48.112522  
p6(-1.055106) = 48.112522  
p6(2.624262) = -48.112522  
p6(-2.624262) = -48.112522**

**Analyzing real roots of the derivative:**

**Global Minimum value = -48.112522:**

**Global Minimum points:  
x\* = 2.624262  
x\* = -2.624262**