THE CUBIC FORMULA – TI-84 PROGRAM

In the program listings which follow, the italic comments on the right, which are not part of the program, give hints for the key strokes. Other descriptive comments, which also are not part of the program, are preceded with a `*`.

The alternate form of the Cubic Formula is based on the three cube roots of a number, which number may be real or complex. The `prgmCUBEROOT` directly below calculates the required three cube roots. Storing `Z` in your calculator, and then running `prgmCUBEROOT`, stores the three cube roots of `Z` in `I`, `J` and `K`. Finally, note that the $+0i$ terms in the `abs` and `angle` functions are required for them to work properly in all cases when `Z` is real.

```
     prgmCUBEROOT
     :ClrHome                          (PRGM I/O 8)
     :Degree                           (MODE DEGREE)
     :abs(Z+0i)→ R                     (MATH CPX 5 & STO>R)
     :angle(Z+0i)→ Θ                   (MATH CPX 4)
+----:If Θ<0                           (PRGM CTL 1 & 2ND TEST 5)
|    :Then                            (PRGM CTL 2)
|    :Θ+360 → Θ  *this keeps 0 ≤ Θ < 360
+----:End                             (PRGM CTL 7)
     :R^(1/3)→ R
     :Θ/3 → Θ
     :R*(cos(Θ)+i*sin(Θ))→ I
     :R*(cos(Θ+120)+i*sin(Θ+120))→ J
     :R*(cos(Θ+240)+i*sin(Θ+240))→ K
     :Disp "ROOTS [I,J,K]:",I,J,K     (PRGM I/O 3)
     :Return                          (PRGM CTL E)
     :End
```

The `prgmCUBIC`, listed on the next page, implements the alternate form of the Cubic Formula. Specifically, it solves $x^3 + px^2 + qx + r = 0$ for x. Just store the coefficients of the polynomial in `P`, `Q` and `R`, and run `prgmCUBIC`, at which point the three values of x will be stored in `L`, `M` and `N`. Note that `prgmCUBIC` calls `prgmCUBEROOT` above. Also note that the $+0i$ term under the radical is included so that the calculator knows that the square root may be imaginary.

THE CUBIC FORMULA – TI-84 PROGRAM

```
     prgmCUBIC
     :ClrHome
     :(1/9)*(3*Q-P^2)→ C
     :(1/54)*(27*R-9*P*Q+2*P^3)→ D
+----:If C=0                          (2ND TEST 1)
|    :Then
| +--:If D=0
| |  :Then
| |  :(-1/3)*P → L          *c = 0 & d = 0 case
| |  :L → M
| |  :L → N
| +--:Else                            (PRGM CTL 3)
| |  :-2*D → Z
| |  :prgmCUBEROOT                    (PRGM EXEC)
| |  :(-1/3)*P+I → L        *c = 0 & d ≠ 0 case
| |  :(-1/3)*P+J → M
| |  :(-1/3)*P+K → N
| +--:End
+----:Else
|    :-D-√(D^2+C^3+0i)→ Z
|    :prgmCUBEROOT
|    :(-1/3)*P+I-C/I → L    *c ≠ 0 case
|    :(-1/3)*P+J-C/J → M
|    :(-1/3)*P+K-C/K → N
+----:End
     :Disp " "
     :Disp " "
     :Disp "X [L,M,N]:",L,M,N
     :Stop                          (PRGM CTL F)
     :End
```