Complex Number Subroutines

William R Harlow, Department of Mechanical and Industrial Engineering, 836 Rhodes Hall, University of Cincinnati, Cincinnati OH 45221

I teach numerical methods to engineering students at the University of Cincinnati, where we have an Amdahl computer. Also, various departments have purchased Heath, IMSAI, Radio Shack, and Wang systems. Although the big system has built-in hardware to perform complex operations, the smaller systems must have them implemented as subroutines.

Besides the four fundamental operations of addition, subtraction, multiplication, and division, there are several important functions of a complex variable. These include log(z), e^z, sin(z), cos(z), z^p, and others. Since addition and subtraction are so easy to handle, they are not included in the routines listed here.

Listing 1 gives a set of BASIC routines to do the complex operations listed in table 1. Other functions not

Listing 1: *Subroutines for manipulation of complex numbers. See table 1 for a description of the functions calculated. Note that some of the routines use the constant #PI, which should be set to 3.1415926535.*

```
1000 REM
1010 M1=A1*A2-B1*B2:M2=A1*B2+A2*B1:RETURN
2000 REM
2010 D=A2↑2+B2↑2
2020 Q1=(A1*A2+B1*B2)/D:Q2=(A2*B1-A1*B2)/D:RETURN
3000 REM
3010 R=SQR(A1↑2+B1↑2):I=SGN(A1)+3*SGN(B1)+4
3020 ON I GOTO 3050,3060,3070,3110,3080,3090,3100,3060
3030 B=ARCTAN(B1/A1)-#PI:GOTO 3120
3050 B=(-#PI/2):GOTO 3120
3060 B=ARCTAN(B1/A1):GOTO 3120
3070 B=#PI:GOTO 3120
3080 B=0:GOTO 3120
3090 B=#PI+ARCTAN(B1/A1):GOTO 3120
3100 B=#PI/2:GOTO 3120
3110 P1,P2=0:GOTO 3120
3120 R0=P*LOG(R):R=EXP(R0)
3130 P1=R*COS(P*B):P2=R*SIN(P*B):RETURN
4000 REM
4010 I=SGN(A1)+3*SGN(B1)+4
4020 IF I=4 THEN 4120
4030 L=.5*LOG(A1↑2+B1↑2)
4040 ON I GOTO 4060,4070,4080,4120,4090,4100,4110,4070
4050 L2=ARCTAN(B1/A1)-#PI:GOTO 4130
4060 L2=(-#PI/2):GOTO 4130
4070 L2=ARCTAN(B1/A1):GOTO 4130
4080 L2=(#PI):GOTO 4130
4090 L2=0:GOTO 4130
4100 L2=#PI+ARCTAN(B1/A1):GOTO 4130
4110 L2=#PI/2:GOTO 4130
4120 PRINT "LOG(Z) IS UNDEFINED":STOP :RETURN
4130 L1=L:RETURN
5000 REM
5010 E1=EXP(A1)*COS(B1):E2=EXP(A1)*SIN(B1):RETURN
6000 REM
6010 U1=(EXP(B1)-EXP(-B1))/2:U2=(EXP(B1)+EXP(-B1))/2
6020 S1=SIN(A1)*U2:S2=COS(A1)*U1:RETURN
7000 REM
7010 U1=(EXP(B1)-EXP(-B1))/2:U2=(EXP(B1)+EXP(-B1))/2
7020 C1=COS(A1)*U2:C2=SIN(A1)*(-U1):RETURN
8000 REM
8010 IF B1<>0 THEN 8050
8020 IF A1<0 THEN 8040
8030 R1=SQR(A1):R2=0:RETURN
8040 R1=0:R2=SQR(-A1):RETURN
8050 R=SQR(A1↑2+B1↑2)
8060 R1=SQR((R+A1)/2):R2=SGN(B1)*SQR((R-A1)/2):RETURN
```

Line Number	Operation type	Input; Use	Other Variables Used	Output
1000	product $z_1 \times z_2$	A1,B1;A2,B2		M1,M2
2000	quotient z_1 / z_2	A1,B1;A2,B2	D	Q1,Q2
3000	power z^p	A1,B1	P,R,I,B	P1,P2
4000	natural logarithm Ln z	A1,B1	I,L	L1,L2
5000	exponential e^z	A1,B1		E1,E2
6000	sine sin z	A1,B1	U1,U2	S1,S2
7000	cosine cos z	A1,B1	U1,U2	C1,C2
8000	square root $z^{1/2}$	A1,B1	R	R1,R2

Table 1: *Table of complex number operations performed by subroutines in listing 1. In the "Input" column (A1, B1) refers to the complex number A1+B1i, where i is the square root of −1. In the "Output" column, the two numbers listed are the real and imaginary parts of the answer; eg: the output variables M1 and M2 of the multiplication routine mean that the result of the multiplication is the complex number M1+M2i.*

included could be the hyperbolic and inverse trigonometric functions. The square root of a complex number was included even though it is a special case of z^p. The only complicated ones are the power and the logarithm. This is due to the angle utilized.

The subroutines have been given large line numbers so that they may be put at the end of a program. Users can certainly renumber these lines or use only those needed for a particular problem.

Two rather simple problems (see listings 2 and 3) are included to demonstrate the use of the functions. Both make use of Newton's method to solve for the roots of a function. This is done using the following iterative formula to obtain a better approximation of z, z_{k+1}, from the current approximation, z_k:

$$z_{k+1} = z_k - f(z_k)/f'(z_k) \text{ where } k=1,2,\ldots$$

An initial or starting value of z is selected ($z=x+iy$). Thus $z_1 = x_1 + iy_1$ is used in $f(z_1)$ and $f'(z_1)$. This will generate a z_2 which is fed back into the right-hand side of the equation to give a z_3, and so on.

The method is rapid in convergence and quite stable. If a certain z_k should make $f'(z_k)$ very small or zero, however, it is best to restart with a new z_1. In the programs shown, a test to stop cycling is made on the $f(z)$:

IF SQR(F1↑2+F2↑2) <1E−6 THEN . . .

This statement stops the iteration when the complex error has a magnitude of less than 10^{-6}.■

Listing 2: *Example program using the subroutines of listing 1. The program given in listing 2a attempts to find a root of the function $f(z)=e^z - z^2$. Note that its derivative $f'(z)=g(z)=e^z -$ 2z. Listing 2b shows two separate runs of the program with starting points of (1,1) and (−1,0); the final results are underlined. Due to the cyclic nature of e^z, there are an infinite number of solutions to this problem.*

(2a)

```
10 INPUT "                          KEY IN X,Y ",X,Y
12 PRINT
15 PRINT TAB(14);X,Y
20 A1=X:B1=Y
30 GOSUB 5000
40 P=2
50 GOSUB 3000
60 F1=E1-P1:F2=E2-P2
65 IF SQR(F1↑2+F2↑2)<1E-6 THEN 120
70 G1=E1-2*A1:G2=E2-2*B1
80A1=F1:B1=F2:A2=G1:B2=G2
90 GOSUB 2000
100 X=X-Q1:Y=Y-Q2
110 GOTO 15
120 STOP "           ROOT DETERMINED. KEY RUN FOR A NEW SET"
```

(2b)

X_1 =	Y_1 =
1	1
2.912389622375	2.575157181739
2.187132232955	2.174648753578
1.760811047732	1.808824533853
1.603663701734	1.596954184978
1.58722527008	1.54253028231
1.588042823737	1.540223443863
1.588047264669	1.540223501065

X_1 =	Y_1 =
−1	0
−.733043605249	0
−.7038077863239	0
−.7034674683272	0

Listing 3: *Example program using the subroutines of listing 1. The program given in listing 3a attempts to find a root of the function $f(z)=2z^2 + (-6 - i)z + (20 - i) = (2z + 4 - i)(z - 5)$. (Its roots are $(-2 + 0.5i)$ and 5.) The derivative $f'(z)=g(z)=4z + (-6 - i)$. Two runs of the program are shown in listing 3b, with the final results underlined.*

(3a)

```
10 INPUT "                          KEY IN X,Y ",X,Y
12 PRINT
15 PRINT TAB(14);X,Y
20 A1=X:B1=Y
40 P=2
50 GOSUB 3000
60 F1=2*P1:F2=2*P2
70 A2=-6:B2=-1
80 GOSUB 1000
90 F1=F1+M1-20:F2=F2+M2+5
95 IF SQR(F1↑2+F2↑2)<1E-6 THEN 200
100 G1=4*A1-6:G2=4*B1-1
110 A1=F1:B1=F2:A2=G1:B2=G2
120 GOSUB 2000
130 X=X-Q1:Y=Y-Q2
140 GOTO 15
200 STOP          "ROOT DETERMINED. KEY RUN FOR A NEW SET"
```

(3b)

X_1 =	Y_1 =
1	1
−3.307692307727	−4.461538461515
−1.45941644561	−1.379310344755
−1.434942737807	.532192367931
−2.053130882705	.4886935917174
−2.00036624035	.4998063289297
−2.00000001228	.4999999788526

X_1 =	Y_1 =
2	2
2.207547169882	−2.226415094319
2.830440251643	1.193459119487
4.902563504007	−1.877088064073
4.604564248345	−.193451138577
5.015324400454	2.68292464E−02
4.999923902019	1.12126002E−04
4.999999999177	−2.49665620E−09