ALGORITHM 53
NTH ROOTS OF A COMPLEX NUMBER

JOHN R. HERNDON
Stanford Research Institute, Menlo Park, California

```
procedure  NTHROOT (n, r, u, REAL, UNREAL); value
           n, r, u; integer n;
           real r, u; real array REAL, UNREAL;
comment  This procedure computes the n roots of the equation
  x^n = r+ui. The real parts of the roots are stored in the vector
  REAL [ ]. The imaginary parts are stored in the corresponding
  locations in the vector UNREAL [ ];
begin      integer n1, n2; real en, th, s, th 1;
           REAL [n] := 0;
           en := 1/n;
           if u=0 then
             begin s := (abs(r)) ↑ en;
             th := 0,
             go to main end;
           if r=0 then
             begin s := (abs(u)) ↑ en;
             th := 1.5707963;
             if u < 0 then
               th := -th
             go to main end;
             s := (r × r+u × u) ↑ (en/2);
             th := arctan (u/r);
main:      if r < 0 then
             th := th + 3.1415926;
           th := en × th;
           th1 := 6.2831853 × en;
           for n2 := 1 step 1 until n do
             begin REAL [n2] := s × cos (th);
             UNREAL [n2] := s × sin (th);
             th = th+th 1 end
end NTHROOT;
```

REMARK ON ALGORITHM 53
Nth ROOTS OF A COMPLEX NUMBER (John R. Herndon, *Comm. ACM 4*, Apr. 1961)

C. W. NESTOR, JR.
Oak Ridge National Laboratory, Oak Ridge, Tennessee

A considerable saving of machine time for $N \geqq 3$ would result from the use of the recursion formulas for the sine and cosine in place of an entry into a sine-cosine subroutine in the do loop associated with the Nth roots of a complex number. That is, one could use

$$\sin(n+1)\theta = \sin n\theta \cos\theta + \cos n\theta \sin\theta$$
$$\cos(n+1)\theta = \cos n\theta \cos\theta - \sin n\theta \sin\theta,$$

at the cost of some additional storage.

We have found this procedure to be very efficient in problems dealing with Fourier analysis, as suggested by G. Goerzel in chapter 24 of *Mathematical Methods for Digital Computers*.