CERTIFICATION OF ALGORITHM 279 [D1]
CHEBYSHEV QUADRATURE [F. R. A. Hopgood and C. Litherland, *Comm. ACM 9*, 4 (Apr. 1966), 270]

KENNETH HILLSTROM (Recd. 16 Dec. 1966 and 30 Jan. 1967)
Applied Mathematics Division, Argonne National Laboratory, Argonne, Illinois

Work performed under the auspices of the US Atomic Energy Commission

The 40th line of the first column on page 270 should read: *badda* := .5 × (*b*+*a*);

So corrected, Chebyshev quadrature was coded in CDC 3600 ALGOL. A modified version of this quadrature scheme was coded in 3600 Compass language. In this modification the cosine values are program constants, with 3600 single-precision accuracy, as opposed to program generated values, which tests show have maximum absolute errors of 2^{-35}. These errors are carried into the integrand argument evaluation, resulting in large relative errors in the integrand evaluation, for functions bounded by unity over the interval of integration, for example, e^{-x^2} over (0, 4.3) and $\sin(x)$ over $(0, 2\pi)$, which in turn delays convergence.

Since 3600 Compass does not permit dynamic allocation of storage, the dimension of the cosine array must be fixed. The choice of $129 = 2^7 + 1$ terms is based on the recommendation in the comments of Algorithm 279, "A reasonable value for *n*max is probably 7."

The Chebyshev quadrature 3600 ALGOL program, the modified 3600 Compass routine, and 3600 FORTRAN-coded Romberg and Havie integration routines were tested with six integrands. The Romberg and Havie routines are based upon Algorithm 60, Romberg Integration [*Comm. ACM 4*, (June 1961), 225], and Algorithm 257, Havie Integration [*Comm. ACM 8* (June 1965), 381].

The results of these tests are tabulated in Table I. In the table, *A* is the lower limit of the interval of integration, *B* is the upper limit, *EPS* the convergence criterion, *VI* the value of the integral, and *VA* the value of the approximation.

Due to storage requirements, Chebyshev quadrature is restricted to a maximum of 129 function evaluations. For reasons of comparison, this limit is also imposed on Romberg and Havie quadratures. Thus, in some cases the accuracy called for was not obtained.

TABLE I

Integrand	*A*	*B*	*EPS*	*VI*	*Routine*	*VA*	*Number of function evaluations*
e^{-x^2}	0	4.3	10^{-6}	0.886226924	Havie	0.886226924	17
					Romberg	0.886226925	65
					Chebyshev	0.886095576	129
					Chebyshev (Rev.)	0.886226926	17
$\sin(x) + 1$	0	2π	10^{-6}	6.283185308	Havie	6.268233308	129
					Romberg	6.268233309	129
					Chebyshev	6.282993876	129
					Chebyshev (Rev.)	6.283185309	5
$(x)^{-(1/2)}\ln\left(\frac{e}{x}\right)$	0	1	10^{-6}	6.0	Havie	5.034254231	129
					Romberg	5.034254231	129
					Chebyshev	5.829597734	129
					Chebyshev (Rev.)	5.701177427	129
$\ln(x)$	1	10	10^{-6}	14.02585088	Havie	14.02585084	65
					Romberg	14.02585085	65
					Chebyshev	14.02585096	17
					Chebyshev (Rev.)	14.02585097	17
$\ln\left(\frac{e}{x}\right)$	0	1	10^{-6}	2.0	Havie	1.979745104	129
					Romberg	1.979745104	129
					Chebyshev	1.999599461	129
					Chebyshev (Rev.)	1.997983436	129
$\frac{1}{(x^4+x^2+0.9)}$	−1	1	10^{-6}	1.5822329[a]	Havie	1.582238946	17
					Romberg	1.582238946	17
					Chebyshev	1.582232967	17
					Chebyshev (Rev.)	1.582232967	17

[a] The value $\int_{-1}^{+1} \frac{dx}{(x^4 + x^2 + 0.9)} = 1.5822329$ is obtained from C. W. Clenshaw and A. R. Curtis, "A method for numerical integration on an automatic computer," *Numer. Math. 2* (1960), 203.

Algorithms Policy • Revised August, 1966

A contribution to the Algorithms Department should be in the form of an algorithm, a certification, or a remark. Contributions should be sent in duplicate to the editor, typewritten double spaced. Authors should carefully follow the style of this department with especial attention to indentation and completeness of references.

An algorithm must normally be written in the ALGOL 60 Reference Language [*Comm. ACM 6* (Jan. 1963), 1–17] or in ASA Standard FORTRAN or Basic FORTRAN [*Comm.* ACM 7 (Oct. 1964), 590–625]. Consideration will be given to algorithms written in other languages provided the language has been fully documented in the open literature and provided the author presents convincing arguments that his algorithm is best described in the chosen language and cannot be adequately described in either ALGOL 60 or FORTRAN.

An algorithm written in ALGOL 60 normally consists of a commented procedure declaration. It should be typewritten double spaced in capital and lower-case letters. Material to appear in **boldface** type should be underlined in black. Blue underlining may be used to indicate italic type, but this is usually best left to the Editor. An algorithm written in FORTRAN normally consists of a commented subprogram. It should be typewritten double spaced in the form normally used for FORTRAN or it should be in the form of a listing of a FORTRAN card deck together with a copy of the card deck. Each algorithm must be accompanied by a complete driver program in its language which generates test data, calls the procedure, and produces test answers. Moreover, selected previously obtained test answers should be given in comments in either the driver program or the algorithm. The driver program may be publishedwith the algorithm if it would be of major assistance to a user.

For ALGOL 60 programs, input and output should be achieved by procedure statements, using any of the following eleven procedures (whose body is not specified in ALGOL) [See "Report on Input-Output Procedures for ALGOL 60," *Comm. ACM 7* (Oct. 1964), 628–629]:

insymbol *inreal* *outarray* *ininteger*
outsymbol *outreal* *outboolean* *outinteger*
length *inarray* *outstring*

If only one channel is used by the program for output, it should be designated by 1 and similarly a single input channel should be designated by 2. Examples:

outstring (1, '*x*='); *outreal* (1,*x*);
for *i* := 1 **step** 1 **until** *n* **do** *outreal* (1,*A*[*i*]);
ininteger (2, *digit* [17]);

For FORTRAN programs, input and output should be achieved as described in the ASA preliminary report on FORTRAN and Basic FORTRAN.

It is intended that each published algorithm be well organized, clearly commented, syntactically correct, and a substantial contribution to the literature of Algorithms. It is necessary but not sufficient that a published algorithm operate on some machine and give correct answers. It must also communicate a method to the reader in a clear and unambiguous manner. All contributions will be refereed both by human beings and by an appropriate compiler. Authors should pay considerable attention to the correctness of their programs, since referees cannot be expected to debug them.

Certifications and remarks should add new information to that already published. Readers are especially encouraged to test and certify previously uncertified algorithms. Rewritten versions of previously published algorithms will be refereed as new contributions and should not be imbedded in certifications or remarks.

Galley proofs will be sent to authors; obviously rapid and careful proofreading is of paramount importance.

Although each algorithm has been tested by its author, no liability is assumed by the contributor, the editor, or the Association for Computing Machinery in connection therewith.

The reproduction of algorithms appearing in this department is explicitly permitted without any charge. When reproduction is for publication purposes, reference must be made to the algorithm author and to the *Communications* issue bearing the algorithm.—J.G.Herriot