stichting
mathematisch
centrum

MC

LR 3.5

augustus 1971

SPECIALE FUNCTIES

2e boerhaavestraat 49 amsterdam

Printed at the Mathematical Centre, 49, 2e Boerhaavestraat, Amsterdam.

The Mathematical Centre, founded the 11-th of February 1946, is a non-profit institution aiming at the promotion of pure mathematics and its applications. It is sponsored by the Netherlands Government through the Netherlands Organization for the Advancement of Pure Research (Z.W.O), by the Municipality of Amsterdam, by the University of Amsterdam, by the Free University at Amsterdam, and by industries.

0.1. Lijst van pagina's

Om na te gaan of een exemplaar al dan niet volledig is, kan gebruik gemaakt worden van onderstaande tabel.

LR 3.5.1	bestaat uit 3 blz.	[AUG 1971]
LR 3.5.2	bestaat uit 3 blz.	[AUG 1971]
LR 3.5.3	bestaat uit 3 blz.	[APRIL 1972]
LR 3.5.4	bestaat uit 3 blz.	[APRIL 1972]
LR 3.5.5	bestaat uit 3 blz.	[APRIL 1972]
LR 3.5.6	bestaat uit 3 blz.	[AUG 1972]
LR 3.5.7	bestaat uit 4 blz.	[AUG 1972]

Deze blz. zal bij elke aanvulling en/of wijziging worden vervangen.

0.2. Inhoudsopgave

gamma

auteur: N.M. Temme (MC)

Onderwerp:

berekening van de gammafunktie voor reële waarden van het argument.

Gebruiksaanwijzing.

deklaratie: real procedure gamma (x); value x; real x;
< body > ;

parameters:

x : het argument van de gammafunktie;

Gebruikte procedures: geen.

Voorbeeld van een aanroep:

p : = gamma (1) ; q : = gamma (10) ;
geeft als resultaat
p = + . 10000 00000 009 $_{10}+1$;
q = + . 36287 99999 986 $_{10}+6$;

Methode en prestaties

De berekening geschiedt voor $x \geq 6$ d.m.v. de asymptotische ontwikkeling van de logarithme van de gammafunktie.
Er worden 8 termen van de asymptotische reeks gebruikt. Voor $.5 \leq x < 6$ wordt recursie toegepast. Voor $x < .5$ wordt de reflectieformule gebruikt.

Voor $|x| < 35$ is de relatieve nauwkeurigheid kleiner dan 10^{-10};
voor $35 \leq |x| \leq 306$ kleiner dan 10^{-8}.
Het is niet zinvol de procedure te gebruiken voor $|x| > 306$, aangezien in dat geval gamma (x) groter is dan het op de EL X-8 grootst mogelijke, nog voorstelbare getal (ongeveer 10^{+628}).

Voor x = 0, -1, -2, -3, ... wordt als resultaat 10^{+628} afgeleverd.

<u>Literatuur</u>

Abramowitz, M. and Stegun, I.A. ;
Handbook of Mathematical Functions.
National Bureau of Standards.

Algoltekst:

```
real procedure gamma(x); value x; real x;
begin   if x < .5 then gamma:= if x = entier(x) then ₁₀+628 else
        3.1415 92653 590/gamma(1-x)/sin(3.1415 92653 590×x) else
        begin   real y,v,w,sum; integer k,n;
                k:= if x > 6 then 0 else -entier(x-6);
                y:= x + (if k > 0 then k else 0);
                v:= y × y; sum:= -.29550 65359 477₁₀-1;
                for w:=
                .64102 56410 256₁₀-2, -.19175 26917 527₁₀-2,
                .84175 08417 508₁₀-3, -.59523 80952 381₁₀-3,
                .79365 07936 508₁₀-3, -.27777 77777 778₁₀-2,
                .83333 33333 333₁₀-1 do
                sum:= sum/v + w; w:= 1;
                for n:= k-1 step -1 until 0 do w:= (n+x)×w;
                gamma:= exp((y-.5)×ln(y) + sum/y -y +.91893 85332 047)/w
        end
end gamma;
```

erf

auteur: N.M. Temme (MC)

Onderwerp

berekening van de errorfunktie $\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$ en de complementaire errorfunktie $\mathrm{erfc}(x) = 1 - \mathrm{erf}(x)$ voor reële waarden van x.

Gebruiksaanwijzing:

deklaratie: **real procedure** erf(x, erfc, toler);
value x, toler; **real** x, toler;

parameters:

x: het argument van de errorfunkties;

toler: de gewenste relatieve nauwkeurigheid;
de waarde van toler mag niet kleiner zijn dan de machine-nauwkeurigheid (2 ↑ (-40) ≃ 10 ↑ (-12) voor de EL X-8).

erfc: < variable > ;
krijgt de waarde van erfc(x);

erf : krijgt de waarde van erf(x);

Gebruikte procedures: geen.

Voorbeeld van een aanroep:

p: = erf (1, q, 10 ↑ (-12)); r: = erf (5, s, 10 ↑ (-12));
geeft als resultaat

p = + . 84270 07929 513; q = + . 15729 92070 487;

r = + . 99999 99999 982; s = + . 15374 59794 $424_{10}-11$;

Methode en prestaties

Voor |x| < 1 wordt eerst erf(x) berekend d.m.v. een Taylor ontwikkeling van deze funktie.
Voor |x| > 1 wordt eerst erfc(x) berekend d.m.v. een kettingbreuk ontwikkeling van deze funktie.

Voor details wordt verwezen naar de literatuur, waarin eveneens een FORTRAN-programma staat vermeld.

De nauwkeurigheid hangt naast de op te geven waarde van toler, voornamelijk af van de precisie waarmee de exponentiële funktie wordt berekend. Voor grote waarden van x wordt de opgegeven nauwkeurigheid niet behaald. Met toler = 10^{-12} kan echter altijd een relatieve precisie van 10^{-10} worden behaald, tenzij x > 37.6 ; voor deze laatste waarden van x wordt erf = 1, erfc = 0 afgeleverd, aangezien dan erfc(x) kleiner is dan het op de EL X-8 kleinst mogelijke, nog voorstelbare positieve getal.

Literatuur

Irene A. Stegun and Ruth Zucker

Automatic Computing Methods for Special Functions.
Journal of Res. of NBS - B Vol. 74 B, No.3.

Algoltekst:

```
real procedure erf(x,erfc,toler); value x,toler; real x,erfc,toler;
begin     if x =  0     then begin erf:=  0; erfc:= 1 end else
          if x > 37.6   then begin erf:=  1; erfc:= 0 end else
          if x < -5.06  then begin erf:= -1; erfc:= 2 end else
          begin    real bn,d,f,fn,fnm1,fnm2,gn,gnm1,gnm2,prev,pwr;
                   integer an,wn;
                   pwr:= 2 × x × x; d:= exp(-pwr/2) × 1.1283 79167 096;
                   if abs(x) < 1 then
                   begin    f:= 0; gn:= 1;
                            for wn:= 3,2 + wn while gn > toler do
                            begin gn:= pwr × gn /wn; f:= f + gn end;
                            f:= erf:= (1 + f) × x × d; erfc:= 1 - f
                   end else
                   begin    fnm2:= prev:= 0; f:= gnm2:= 1; fnm1:= 2 × x;
                            gnm1:= 1 + pwr; f:= fnm1 / gnm1; bn:= 4 + gnm1;
                            for wn:= 1, 2 + wn while 1 - prev / f > toler do
                            begin    an:= wn × wn +wn;
                                     fn:= bn × fnm1 - an × fnm2;
                                     gn:= bn × gnm1 - an × gnm2;
                                     prev:= f; f:= fn/gn; fnm2:= fnm1; gnm2:= gnm1;
                                     fnm1:= fn; gnm1:= gn; bn:= 4 + bn
                            end;
                   erfc:= (if prev < f then f else prev) × d / 2 +
                                     (if x < 0 then 2 else 0); erf:= 1 - erfc
                   end
          end
end erf;
```

auteur: N.M. Temme (MC)

incomgam

Onderwerp:

berekening van de incomplete gammafuncties.

$$\gamma(a,x) = \int_0^x e^{-t}\, t^{a-1}\, dt \quad \text{en} \quad \Gamma(a,x) = \int_x^\infty e^{-t}\, t^{a-1} dt$$

voor reële waarden van a en x; $x \geq 0$, $a > 0$.

$$\gamma(a,x) + \Gamma(a,x) = \Gamma(a)$$

Gebruiksaanwijzing:

deklaratie : procedure incomgam (x, a, klgam, grgam, gam, eps);
value x, a, eps; real x, a, klgam, grgam, gam, eps;
<procedure body>;

parameters :

x, a : zie definitie van $\gamma(a,x)$ en $\Gamma(a,x)$;

klgam : <variable>; krijgt de waarde $\gamma(a,x)$;

grgam : <variable>; krijgt de waarde $\Gamma(a,x)$;

gam : <expression>; hiervoor dient de waarde van $\Gamma(a)$ ingevuld te worden (zie eventueel LR 3.5.1.);

eps : de gewenste relatieve nauwkeurigheid; de waarde van eps mag niet kleiner zijn dan de machine-nauwkeurigheid ($2\uparrow(-40) \simeq 10^{-12}$ voor de EL X 8);

Gebruikte procedures: geen

Voorbeeld van een aanroep:

incomgam (3, 4, klgam, grgam, 6, 2↑ (-40)) geeft als resultaat

klgam = .211 660 866 731 $5 \times 10^{+1}$
grgam = .388 339 133 268 $5 \times 10^{+1}$

(de werkelijke waarde van grgam is
$\Gamma(3,4)$ = .388 339 133 269 3 4 ...$\times 10^{+1}$).

Methode en prestaties

Voor $x \leq$ (if $a < 3$ then 1 else a) wordt eerst $\gamma(a,x)$ berekend en daarna $\Gamma(a,x) = \Gamma(a) - \gamma(a,x)$;
voor grotere waarden van x wordt eerst $\Gamma(a,x)$ berekend en daarna $\gamma(a,x) = \Gamma(a) - \Gamma(a,x)$.
De funkties worden berekend m.b.v. rationale approximaties; voor kleine x via de Padé-approximatie die correspondeert met de Taylor ontwikkeling van $\gamma(a,x)$; voor grotere waarden van x via de Padé-ontwikkeling die correspondeert met de asymptotische ontwikkeling van $\Gamma(a,x)$ voor $x \to \infty$.
Voor details wordt verwezen naar de literatuur.

De nauwkeurigheid hangt, naast de op te geven waarde van eps, eveneens af van de nauwkeurigheid waarmee gam (= $\Gamma(a)$) en de exponentiele funktie worden berekend.
Vooral voor grote waarden van a en x wordt de nauwkeurigheid eps niet behaald.

Literatuur:

Y.L. Luke, Evaluation of the gammafunction by means of Padé approximations.
SIAM Journal Math. Anal. Vol. 1, No. 2 (1970)

Algoltekst :

```
procedure incomgam(x,a,klgam,grgam,gam,eps);
value x,a,eps; real x,a,klgam,grgam,gam,eps;
begin   real c0, c1, c2, d0, d1, d2, x2, ax, p, q, r, s, r1, r2;
        integer n;
        s:= exp(-x + a × ln(x));
        if x ≤ (if a < 3 then 1 else a) then
        begin   x2:= x × x; ax:= a × x; d0:= 1; p:= a; c0:= s;
                d1:=(a+1)×(a+2-x); c1:=((a+1) × (a+2)+x) × s;
                r2:= c1/d1;
                for n:=1, n+1 while abs((r2-r1)/r2) > eps do
                begin   p:= 2 + p; q:=(p+1) × (p×(p+2)-ax);
                        r:= n × (n+a) ×(p+2) ×x2;
                        c2 := (q×c1 + r×c0)/p; d2:=(q×d1 +r×d0)/p;
                        r1:=r2; r2:=c2/d2;
                        c0:=c1; c1:=c2; d0:=d1; d1:=d2
                end; klgam:= r2/a; grgam:= gam - klgam
        end else
        begin   c0:=a×s; c1:=(1+x)× c0; q:= x + 2 - a;
                d0:= x; d1:= x × q; r2:= c1/d1;
                for n:=1, n+1 while abs((r2-r1)/r2)>eps do
                begin   q:= 2 + q; r:= n × (n+1-a);
                        c2:= q×c1-r×c0; d2:= q×d1-r×d0;
                        r1:=r2; r2:=c2/d2;
                        c0:=c1; c1:=c2; d0:=d1; d1:=d2
                end; grgam:= r2/a; klgam:= gam - grgam
        end
end incomgam;
```

kO auteur: N.M. Temme (MC)

Onderwerp:
Berekening van de gemodificeerde Besselfuncties $K_0(x)$ en $K_1(x)$ voor reële positieve waarden van het argument.

Gebruiksaanwijzing:

deklaratie: real procedure k0(x, k1); value x; real x, k1; <body>;

parameters:

x : het argument van de Besselfuncties;

k1 : <variable>;
krijgt de waarde van $K_1(x)$;

k0 : krijgt de waarde van $K_0(x)$.

Gebruikte procedures: geen.

Voorbeeld van een aanroep:
K [0] := k0(1, K[1]) geeft als resultaat
K [0] = .421 02443 82402;
K [1] = .601 90723 01970.

Methode en prestaties:

Voor $0 < x \leq 2.5$ worden K_0 en K_1 berekend via een reeksontwikkeling (zie Abramowitz formule 9.6.11). Voor $x > 2.5$ worden de functies berekend met behulp van een recurrente betrekking (13.4.15) voor de confluente hypergeometrische functies. K_0 en K_1 kunnen als dergelijke functies opgevat worden (13.6.21). De hypergeometrische functies $U(k + \frac{1}{2}, 1, x)$ kunnen voor $k = 0, 1, 2, \ldots$ berekend worden met terugwaartse recursie (methode van Miller) via (13.4.15). Als normeringsrelatie wordt gebruikt

$$\sum_{k=0}^{\infty} \frac{\Gamma(k+\frac{1}{2}) \quad \Gamma(k+\frac{1}{2})}{\Gamma(\frac{1}{2}) \ \Gamma(\frac{1}{2})k!} U(k+\tfrac{1}{2}, 1, x) = x^{-\frac{1}{2}}.$$

Als $U(\frac{1}{2}, 1, x)$ en $U(3/2, 1, x)$ bekend zijn kunnen K_0 en K_1 gevonden worden door gebruik te maken van enkele recurrente betrekkingen voor $U(a, b, x)$. De methode van berekening wordt voor een soortgelijke klasse van functies uitvoerig beschreven in een TW-rapport "Numerical evaluation of functions arising from transformations of formal series", dat binnenkort zal verschijnen.

De relatieve nauwkeurigheid is kleiner dan 10^{-11}.

De tijdsduur voor de ELX8 bedraagt maximaal 0.05 seconde.

De overige gemodificeerde Besselfuncties $K_n(x)$ kunnen voor $n \geq 2$ berekend worden uit

$$K_{n+1}(x) = K_{n-1}(x) + \frac{2n}{x} K_n(x)$$

Deze recursie (met toenemende n) is numeriek stabiel.

<u>Literatuur</u>:

(voor nadere omschrijving van deze Besselfuncties)

Abramowitz, M. and Stegun, I.A.
Handbook of Mathematical Functions,
National Bureau of Standards.

Algoltekst:

```
real procedure k0(x,k1); value x; real x,k1;
begin   real c,d,r,s,sum0,sum1,t,term,t0,t1;
        integer k,nu;
        if  x < 2.5  then
        begin   sum0:= d:= ln(2/x) - .5772 15664 9015;
                sum1:=  c:= -1 - 2 × d; r:= term:= 1; t:= x × x/4;
                for k:= 1,k+1 while abs(t0/sum0) + abs(t1/sum1) > ₁₀-12 do
                begin   term:=  t × term × r × r; d:= d + r; c:= c - r;
                        r:= 1/(k+1); c:= c - r;
                        t0:= term × d; t1:= term × c × r;
                        sum0:= sum0 + t0; sum1:= sum1 + t1
                end;
                k0:= sum0; k1:= (1 + t × sum1) / x
        end else

        begin   t:= 2 × x; c:= 1; nu:= entier(10 + 65 / x);
                for  k:= 1  step  1  until  nu  do
                c:= c × ((2 × k - 1) ↑ 2) / k / 4;
                r:= s:= 0; d:= (2 × nu + 1) ↑ 2;
                for  k:= nu  step  -1  until  1  do
                begin   t0:= -4 × (2 × k + t) / d; t1:= 4 / d;
                        r:= -t1 / (r + t0); s:= r × (c + s);
                        d:= d - 8 × k; c:= c × k × 4 / d
                end;
                c:= k0:= 1.7724 53850 9055 × exp(-x)/sqrt(2 × x)/(1+s);
                k1:= c × (1 + t - r / 2) / t
        end
end k0;
```

auteurs: R. Hofhuis (MC)
N.M. Temme (MC)

incbeta

Onderwerp:

berekening van de incomplete betafunktie

$$I_x(a, b) = \frac{\Gamma(a+b)}{\Gamma(a).\Gamma(b)} \int_0^x t^{a-1}(1-t)^{b-1}dt.$$

Gebruiksaanwijzing:

deklaratie: real procedure incbeta (x, a, b, eps);
value x, a, b, eps; real x, a, b, eps;
<body>;

parameters:

x, a, b: <expression>;
x, a, b zijn de variabelen van de incomplete betafunktie, die de volgende waarden mogen aannemen:
$0 \leq x \leq 1$, $a > 0$, $b > 0$;

eps : <expression>;
eps is de relatieve precisie;
de waarde van eps mag niet kleiner zijn dan de machine-nauwkeurigheid.
(2 ↑ (-40) ≃ 10 ↑ (-12) voor de ELX -8).

Gebruikte procedures: gamma (zie LR 3.5.1).

Voorbeeld van aanroep:

y: = incbeta (.3, 5, 10, 10^{-10}); geeft als resultaat y = +.3761 4607 2947.

Methode en prestaties:

$I_x(a,b)$ wordt berekend d.m.v. een kettingbreukrepresentatie die, zoals uit de literatuur blijkt, de beste resultaten geeft als $x \leq$ grens is.

grens:= <u>if</u> a > 1 $\wedge$ b > 1 <u>then</u> (a-1)/(a+b-2) <u>else</u> a/(a+b);
In het andere geval (x > grens) wordt gebruik gemaakt van de betrekking
$I_x(a,b) = 1-I_{1-x}(b,a)$

De nauwkeurigheid waarmee de kettingbreuk berekend wordt, is bepaald door eps. De precisie van $I_x(a,b)$ is mede afhankelijk van de precisie van gamma (LR 3.5.1). In verband hiermee is het mogelijk dat de opgegeven relatieve nauwkeurigheid niet behaald wordt als a+b > 35.
Als a < 100 en b < 100 dan is echter altijd een relatieve nauwkeurigheid van 10^{-9} te bereiken. De gebruiker wordt met nadruk gewezen op de beperkte range van de parameters:
$0 \leq x \leq 1$, $a > 0$, $b > 0$.

<u>Literatuur</u>:

Abramowitz, M. and Stegun, I.A.;
Handbook of Mathematical Functions,
National Bureau of Standards.

Algoltekst:

```
real procedure incbeta(x,a,b,eps); value x,a,b,eps; real x,a,b,eps;
begin   integer n; real grens, g, f, fn , fn1, fn2, gn, gn1, gn2, dn;
        boolean n even;
        grens:= if a>1 ∧ b>1 then (a-1)/(a+b-2) else a/(a+b);
        if x>grens  then incbeta:= 1-incbeta(1-x,b,a,eps) else
        begin   g:= fn2:= 0;  f:= fn1:= gn1:= gn2:= 1; n even:= false;
                for n:= 1,n+1 while abs((f-g)/f) > eps do
                begin   dn:= if n even then .5×n×x×(b-.5×n)/((a+n-1)×(a+n))
                        else -x×(a+.5×(n-1))×(a+b+.5×(n-1))/((a+n-1)×(a+n));
                        g:=f; fn:= fn1+dn×fn2;  gn:= gn1+dn×gn2;
                        n even:=¬ n even; f:= fn/gn;
                        fn2:= fn1;  fn1:= fn;  gn2:= gn1;  gn1:= gn
                end;  incbeta:= f × x↑a × (1-x)↑b × gamma(a+b) /
                               (gamma(a+1) × gamma(b))
        end
end incbeta
```

gam

auteur: N.M. Temme (MC)

Onderwerp:
berekening van de gammafunktie $1/\Gamma(1-a)$ voor $|a| \leq .5$.

Gebruiksaanwijzing:

deklaratie : **real** **procedure** gam(a,odd,even);
value a; **real** a,odd,even;
< body >;

parameters:

a : < expression >;
het argument van de gammafunktie;
de range van a is: $|a| \leq .5$;

odd : < variable >;
krijgt de waarde $(1/\Gamma(1-a) - 1/\Gamma(1+a))/(2a)$;

even : < variable >;
krijgt de waarde $(1/\Gamma(1-a) + 1/\Gamma(1+a))/2$;

gam : de waarde van $1/\Gamma(1-a)$
(gam = a $\times$ odd + even).

Gebruikte procedures: geen.

Voorbeeld van een aanroep:

p:= gam(0,q,r); a:= gam(.5,b,c);
geeft als resultaat

$p = +.10000\ 00000\ 002_{10}1$;	$(0.2_{10}{-11})$
$q = -.57721\ 56649\ 027_{10}0$;	$(0.12_{10}{-11})$
$r = +.10000\ 00000\ 002_{10}1$;	$(0.2_{10}{-11})$
$a = +.56418\ 95835\ 470_{10}0$;	$(.8_{10}{-12})$
$b = -.56418\ 95835\ 479_{10}0$;	$(.2_{10}{-12})$
$c = +.84628\ 43753\ 214_{10}0$;	$(.2_{10}{-12})$

(tussen haakjes staat de absolute fout vermeld).

Methode en prestaties:

De berekening geschiedt door middel van een Chebyshev-ontwikkeling in de vorm

$$1/\Gamma(1-a) = \sum_{k=0}^{11}{}' \, b_k \, T_k(2a) \; ,$$

($\sum'$ betekent dat de eerste term van de reeks gehalveerd moet worden).
De coefficienten b_k zijn berekend via de coefficienten c_k, gedefinieerd door

$$1/\Gamma(z) = \sum_{k=0}^{\infty} c_k z^k .$$

Deze coefficienten staan getabelleerd in [1, p. 256]. Door z^k door een combinatie van Chebyshev-polynomen te vervangen, kan men de Chebyshev-ontwikkeling van $1/\Gamma(z)$ verkrijgen (en dan ook die van $1/\Gamma(1-z)$ via $\Gamma(1-z) = -z\Gamma(1-z)$). Voor deze methode en voor de sommatie van Chebyshev-reeksen wordt verwezen naar [2], (p. 4 en 9 respectievelijk).

Voor algemeen gebruik is het niet erg interessant dat het even en oneven gedeelte (t.o.v. a) apart uitgerekend wordt

$$(\text{even} = \sum_{k=0}^{5}{}' b_{2k}T_{2k}(2a) \quad , a \times \text{odd} = \sum_{k=0}^{5} b_{2k+1}\, T_{2k+1}\,(2a)) .$$

Van deze gedeelten wordt echter een essentieel gebruik gemaakt in een procedure voor Besselfuncties (zie LR 3.5.7.).

Voor waarden van a buiten het interval [$-$ 1/2, 1/2] kan de gammafunktie berekend worden door (eventueel herhaald) toepassen van de recursie-formule $\Gamma(z+1) = z\Gamma(z)$. Voor grote waarden van a kan echter beter LR 3.5.1. gebruikt worden.

De procedure is getest door na te gaan of voldaan is aan de betrekking

$$\Gamma(1-a) \times \Gamma(1+a) = a\pi/\sin a\pi .$$

Voor a = 0.002 (0.002) .5 bleek het verschil
1 − gam(a,b,c) × gam(−a,b,c) × a × (pi)/sin(a × (pi))
in absolute waarde hoogstens $.73_{10}{-11}$ te zijn.

<u>Literatuur:</u>

[1] M. Abramowitz and I.A. Stegun (eds)
: Handbook of Mathematical Functions.
National Bureau of Standards.

[2] C.W. Clenshaw : Chebyshev series for mathematical functions.
Nat. Phys. Lab. Math. Tables, vol. 5.
H.M. Stationery Office, London. (1962)

Algoltekst:

```
real procedure gam(a,odd,even); value a; real a,odd,even;
begin   real p,q,r,x; integer n; boolean first; array b[0:5];
        first:= true; x:=16×a×a-2;
        b[0]:= -2.8387 65422 760 ₁₀-1; b[1]:=  1.7063 050711 ₁₀-3;
        b[2]:=  7.6309 5976 ₁₀-5;      b[3]:= -8.6592 08 ₁₀-7;
        b[4]:=  1.7451 ₁₀-9;           b[5]:=  9.2 ₁₀-12;
label:  q:= r := 0;
        for n:= 5 step -1 until 0 do
        begin   p:= x×q-r+b[n];
                if n≠0 then begin r:= q; q:= p end
        end;  if first then
        begin   odd:= 2×(p-q); first:= false;
                b[0]:= 1.8437 40587 301; b[1]:= -7.6852 840845₁₀-2;
                b[2]:= 1.2719 27137 ₁₀-3; b[3]:= -4.9717 37 ₁₀-6;
                b[4]:=-3.3126 ₁₀-8;       b[5]:= 2.42 ₁₀-10;
                goto label
        end  else q:= (p-r)/2;
        gam:= odd×a+q; even:= q
end gam;
```

ka auteur: N.M. Temme (MC)

Onderwerp:
berekening van de gemodificeerde Besselfuncties $K_a(x)$ en $K_{a+1}(x)$ voor $0 \le a \le 1$ en $x > 0$.

Gebruiksaanwijzing:

deklaratie : **real procedure** ka(a,x,ka1);
value a,x; **real** a,x,ka1;
< body >;

parameters:

a : < expression >;
de orde van de Besselfunctie;

x : < expression >;
het argument van de Besselfunctie;

ka1 : < variable >;
krijgt de waarde van $K_{a+1}(x)$;

ka : krijgt de waarde van $K_a(x)$.

Gebruikte procedures: gam; zie LR 3.5.6.

Voorbeeld van een aanroep:

p:= ka(0,1,q); geeft als resultaat (vergelijk LR 3.5.4.)
p = .421 02443 82407;
q = .601 90723 01988.

Methode en prestaties:

Voor $x > 3$ is de berekening analoog aan die van $K_0(x)$ en $K_1(x)$ voor $x > 2.5$ (zie LR 3.5.4.). Als normering wordt hier gebruikt

$$\sum_{n=0}^{\infty} \frac{(-1)^n}{n!} \frac{\Gamma(a+\frac{1}{2}+n)}{\Gamma(a+\frac{1}{2}-n)} U(a+n+\tfrac{1}{2},2a+1,x) = (x)^{-a-\frac{1}{2}} .$$

Voor $0 < x \le 3$ is de berekening eveneens analoog, maar voor a = 0 is deze veel eenvoudiger. De berekening verloopt als volgt. Uitgangspunt is de formule

$$K_a(x) = \frac{\pi}{2} \frac{I_{-a}(x) - I_a(x)}{\sin(a\pi)} ,$$

waarin $I_a(x)$ een gemodificeerde Besselfunctie is, die gedefinieerd kan worden door

$$I_a(x) = \left(\frac{x}{2}\right)^a \sum_{k=0}^{\infty} \frac{(\frac{1}{4}x^2)^k}{k!\,\Gamma(a+k+1)} .$$

Hieruit volgt voor $K_a(x)$ dan

$$K_a(x) = \sum_{k=0}^{\infty} f_k ,$$

$$f_0 = \frac{\pi}{2\sin(a\pi)} \left\{ \frac{(\frac{x}{2})^{-a}}{\Gamma(1-a)} - \frac{(\frac{x}{2})^{a}}{\Gamma(1+a)} \right\} ,$$

$$f_1 = \left[f_0 + \frac{a\pi}{2\sin(a\pi)} \left\{ \frac{(\frac{x}{2})^{-a}}{\Gamma(1-a)} + \frac{(\frac{x}{2})^{a}}{\Gamma(1+a)} \right\} \right] / (1-a^2) ,$$

en

$$f_k = \frac{1}{4}x^2 \, \frac{(k-1)\,(2k-1)\, f_{k-1} - \frac{1}{4}x^2 \, f_{k-2}}{k(k-1)\,(k^2-a^2)} , \quad k = 2,3,4, \dots .$$

Voor $K_{a+1}(x)$ vinden we

$$K_{a+1}(x) = A + \sum_{k=0}^{\infty} g_k ,$$

$$A = \frac{a\pi}{2\sin a\pi} \, \frac{(\frac{x}{2})^{-a-1}}{\Gamma(1-a)} ,$$

$$g_0 = - \frac{x}{2} \left(f_0 + A \frac{x}{2}\right) / (a+1)$$

$$g_k = \frac{x}{2} \left\{ - f_k + \frac{x}{2} \, \frac{g_{k-1} + \frac{x}{2k} f_{k-1}}{a + k + 1} \right\} / (k+1)$$

voor k = 1,2,3, ...

Het lastige punt van de berekening betreft f_0. Als deze term bekend is volgt de rest zonder moeite. Rechtstreekse berekening van f_0 levert moeilijkheden op voor kleine waarden van a, d.w.z. voor die waarden van a, waarvoor de termen tussen de accolades nagenoeg gelijk zijn aan elkaar. Hoewel de limiet voor a -> 0 bestaat, levert rechtstreekse numerieke evaluatie voor a -> 0 een grote relatieve fout op door wegvallen van significante cijfers. De term f_0 kan echter als volgt geschreven worden, waarmee het gevaar voor cijferverlies is opgeheven.

$$f_0 = \frac{a\pi}{\sin a\pi} \left\{ g_1(a) \, ch\left(a \ln\left(\frac{2}{x}\right)\right) + g_2(a) \ln\left(\frac{2}{x}\right) \frac{sh\left(a \ln\left(\frac{2}{x}\right)\right)}{a \ln\left(\frac{2}{x}\right)} \right\}$$

waarin

$$g_1(a) = \left\{\frac{1}{\Gamma(1-a)} - \frac{1}{\Gamma(1+a)}\right\} / (2a) \; ; \quad g_2(a) = \frac{1}{2} \left\{\frac{1}{\Gamma(1-a)} + \frac{1}{\Gamma(1+a)}\right\} ,$$

(g_1 en g_2 zijn respectievelijk odd en even uit LR 3.5.6.).

Het is voldoende deze algorithme te beschouwen voor $|a| \leq 1/2$, aangezien $K_{-a}(x) = K_a(x)$ en $K_{a+1}(x) = K_{a-1}(x) + (2a/x)K_a(x)$.

Voor a = 1/2 zijn de Besselfuncties overigens elementair, namelijk

$$K_{\frac{1}{2}}(x) = \sqrt{\frac{\pi}{2x}}\, e^{-x} \quad , \quad K_{\frac{3}{2}}(x) = \sqrt{\frac{\pi}{2x}}\, e^{-x} \left(1+\frac{1}{x}\right) .$$

Voor waarden van a buiten het interval [0,1] kunnen de Besselfuncties $K_a(x)$ berekend worden uit $K_a(x) = K_{-a}(x)$ en uit de recurrente betrekking

$$K_{a+k+1}(x) = K_{a+k-1}(x) + \frac{2(a+k)}{x} K_{a+k}(x) ,$$

k = 1,2,3, ... Voor toenemende k is deze recursie numeriek stabiel.

De <u>real procedure</u> ka(a,x,ka1) is getest door na te gaan of voldaan is aan de betrekking

$$K_{a+1}(x)\, I_a(x) + K_a(x)\, I_{a+1}(x) = \frac{1}{x} .$$

De Besselfuncties $I_a(x)$ en $I_{a+1}(x)$ kunnen voor $0 \leq a < 1$ en $x > 0$ berekend worden met behulp van Iaplusn (algorithme 236 uit [2], p 479 ev).

Voor 1250 random-waarden voor a en x in de intervallen [0,1], [0,10] bleek het verschil

$$1 - x\,\{K_{a+1}(x)\, I_a(x) + K_a(x)\, I_{a+1}(x)\}$$

in absolute waarde hoogstens $.85_{10}{-10}$ te zijn.

<u>Literatuur</u>:

[1] M. Abramowitz and I.A. Stegun (eds) : Handbook of Mathematical Functions. National Bureau of Standards.

[2] W. Gautschi : Comm. ACM, 7 (1964).

Algoltekst:

```
real procedure ka(a,x,ka1); value a,x; real a,x,ka1;
begin real a0,a2,x1,x2,y,sinapi,og,eg,sh,ch,f0,f1,f2,
        g0,g1,sum0,sum1; integer n,nu;
        procedure hyp function(x,sh,ch); value x; real x,sh,ch;
        begin    real y; y:= exp(x); ch:= (y+1/y)/2;
                 if abs(x) > .1666 then sh:= (y-1/y)/2/x else
                 begin    y:= x×x; sh:= (42.023 14452 751 + y×(
                          6.0038 57420 809 + 0.18352 62841 200×y))
                          /(42.023 14452 751 -y)
                 end
        endhf;
        if x<3 ∧ abs(a-.5)>⏨-11 then
        begin    x1:= 2/x; if a<.5 then
                 begin    y:= x×x/4; a2:= a×a; a0:= 4×a×arctan(1);
                          x2:= ln(x1); sinapi:= if abs(a0)<⏨-4
                          then 1+a0×a0/6 else a0/sin(a0);
                          f2:= gam(a,og,eg)×x1↑a×sinapi/2;
                          hyp function(a×x2,sh,ch);
                          f0:= sinapi×(og×ch+eg×x2×sh);
                          g0:= -(f0+f2)/x1/(a+1);
                          f1:= y×(f0+sinapi×(eg×ch+a2×x2×og×sh))/
                          (1-a2); sum0:= f0+f1; sum1:= g0+x1×f2;
                          for n:= 2,n+1 while
                          abs(f2/sum0)+abs(g0/sum1)>⏨-13 do
                          begin f2:= y×((n-1)×(2×n-1)×f1-y×f0)/
                            n/(n-1)/(n×n-a2);
                            g0:= (-f1/x1+y×(g0+f0/x1/(n-1))/(a+n))/n;
                            sum0:= sum0+f2;   sum1:= sum1+g0;
                            f0:=f1; f1:=f2
                          end; ka:=sum0; ka1:= sum1
                 end else
                 begin g0:= ka(a-1,x,g1); ka:= g1;
                       ka1:= g0+x1×a×g1
                 end
        end else
        begin    x2:= 2×x; if abs(a-.5)<⏨-11 then
                 begin g1:=0; g0:=1+1/x; goto half end;
                 nu:= entier(10+75/x); a2:= 4×a×a; a0:= 1;
                 for n:= 1 step 1 until nu do
                 a0:= ((2×n-1)↑2-a2)×a0/n/4;
                 g0:= g1:= 0; y:=(2×nu+1)↑2-a2; f0:= 4/y;
                 for n:= nu step -1 until 1 do
                 begin    f1:= -(2×n+x2)×f0; g0:= -f0/(f1+g0);
                          g1:=g0×(a0+g1); y:= y-8×n;
                          f0:= 4/y; a0:= f0×n×a0
                 end;
                 g0:= ((a2-1)×g0/2+1+2×a+x2)/x2;
        half:    y:= ka:= 1.7724538509055×exp(-x)/sqrt(x2)/(1+g1);
                 ka1:= y×g0
        end
end ka;
```