

# Aplikace matematiky

---

Algoritmy 5. RG. 6. Spence. 7. Gauss

*Aplikace matematiky*, Vol. 12 (1967), No. 3, 237–240

Persistent URL: <http://dml.cz/dmlcz/103095>

## Terms of use:

© Institute of Mathematics AS CR, 1967

Institute of Mathematics of the Czech Academy of Sciences provides access to digitized documents strictly for personal use. Each copy of any part of this document must contain these *Terms of use*.



This document has been digitized, optimized for electronic delivery and stamped with digital signature within the project *DML-CZ: The Czech Digital Mathematics Library* <http://dml.cz>

ALGORITMY

5. RG

RECIPROKÁ FUNKCE GAMA

Jiří RAICHL, Matematicko-fyzikální fakulta KU, Malostranské n. 25, Praha 1.

Tato procedura počítá reciprokou hodnotu funkce gama  $1/\Gamma(x)$  pro libovolná  $x$ . Pro  $0 < x \leq 1$  je tato funkce aproximována podle [1], pro  $x > 1$  a  $x < 0$  užíváme vztahu  $\Gamma(x - 1) = \Gamma(x)/(x - 1)$ . Pro  $x = 0, -1, -2, \dots$  dává tato procedura hodnotu 0. Obdobné algoritmy byly publikovány v CACM [2], [3].

```

real procedure RG(x);
value x; real x;
begin real y;
y: = 1;
D: if x < 0 then go to A;
   if x = 0 then begin RG: = 0; go to K end;
   if x  $\leq$  1 then begin B: RG: = y/(((((((( .67710610 - 4  $\times$  x -
      .0003442342)  $\times$  x + .0015397681)  $\times$  x - .0024667480)  $\times$  x +
      .0109736958)  $\times$  x - .0002109075)  $\times$  x + .0742379076)  $\times$  x +
      .0815782188)  $\times$  x + .4118402518)  $\times$  x + .4227843370)  $\times$  x +
      1.0)  $\times$  (x  $\times$  (x + 1));
      go to K
   end;
F: x: = x - 1;
   y: = y/x;
   if x > 1 then go to F else go to B;
A: y: = y  $\times$  x;
   x: = x + 1;
   go to D;
K:
end

```

Algoritmus pro argumenty  $x$  v absolutní hodnotě velké, pracuje (vzhledem k užívání vztahu  $\Gamma(x - 1) = \Gamma(x)/(x - 1)$ ) podstatně pomaleji než pro  $0 < x \leq 1$  (např. již pro  $|x| \doteq 6$  asi dvakrát pomaleji). Vadí-li nám tyto časové ztráty, je lépe užít algoritmus [4], který udává M. C. Pike, J. D. Hill.

Funkci dostáváme s relativní nepřesností až  $10^{-9}$  (podle toho, s jakou nepřesností jsou čísla zobrazena v počítači).

Kontrolní hodnoty:

$$\text{RG}(0,25) = 0,275\ 815\ 663$$

$$\text{RG}(1,75) = 1,088\ 065\ 252$$

$$\text{RG}(-0,5) = -0,282\ 094\ 792$$

Program byl užít na počítači ELLIOT. Výsledky měly relativní nepřesnost menší než  $4 \cdot 10^{-8}$ .

#### Literatura

- [1] *Werner H., Collinge R.*: Math. of Computation, vol. 15 (1961), str. 195.
- [2] *Gautschi W.*: Algorithm 221, CACM vol. 7 (1964), str. 143.
- [3] *Holsten W.*: Algorithm 80, CACM vol. 5 (1962), str. 166.
- [4] *M. C. Pike, I. D. Hill*: Algorithm 291, Logarithm of Gamma Function, CACM vol. 9. (1966), str. 685.

## 6. SPENCE

### SPENCOVA FUNKCE

JIŘÍ RAICHL, Matematicko-fyzikální fakulta KU, Malostranské n. 25, Praha 1.

K výpočtu Spencovy funkce pro  $0 \leq x < \frac{1}{2}$  užíváme řady  $\sum_{i=1}^{\infty} x^i/i^2$ , již sčítáme tak, aby nepřesnost součtu nepřekročila *eps*. Pro argumenty ležící vně tohoto intervalu, užíváme vztahů, které je do tohoto intervalu převádějí [1].\*)

**real procedure** *Spence* (*x*, *eps*);

**value** *x*, *eps*; **real** *x*, *eps*;

**begin real procedure** *S*(*X*);

**value** *X*; **real** *X*;

**begin real** *c*, *s*, *p*; **integer** *i*;

*s* := *X*;

*p* := *X* × *X*;

*c* := *p*/4;

*eps* := *eps*/2;

**for** *i* := 2, *i* + 1 **while** *c* > *eps* **do**

**begin** *s* := *s* + *c*;

*p* := *p* × *X*;

*c* := *p*/(*i* + 1) ↑ 2

**end**;

*S* := *s*

**end** Konec deklarace procedury sčítající řadu  $\sum x^i/i^2$ ;

\*) Tato funkce se často vyskytuje při výpočtech v jaderné fyzice.

```

Spence: = if x ≤ -1 then -1.6449341 + S(1/(1 - x)) - .5 × ln(1 - x)
          × ln(x ↑ 2/(1 - x)) else
          if x < 0 then -S(x/(x - 1)) - .5 × ln(abs(x - 1)) ↑ 2 else
          if x < .5 then S(x) else
          if x < 1 then 1.6449341 - S(1 - x) - ln(x) × ln(1 - x) else
          if x = 1 then 1.6449341 else
          if x ≤ 2 then 1.6449341 + S((x - 1)/x) - .5 × ln(x) × ln((x - 1) ↑
          2/x)
          else 3.2898681 - S(1/x) - .5 × ln(x) ↑ 2
end

```

end

Hodnotu Spencovy funkce dostaneme pro argumenty  $0 \leq x < \frac{1}{2}$  s absolutní nepřesností menší než *eps*, pokud *eps* dostatečně převyšuje zaokrouhlovací nepřesnosti v počítači, jehož užíváme. Pro argumenty mimo tento interval je relativní nepřesnost zhruba *eps*, pro některé argumenty (např.  $x \approx 1$ ) může být i větší.

Kdybychom chtěli mimo interval  $\langle 0, \frac{1}{2} \rangle$  dosáhnout relativní nepřesnosti vyšší než asi  $10^{-7}$  musili bychom konstantu 1.6449341, která aproximuje  $\pi^2/6$  a konstantu 3.2898681, která aproximuje  $\pi^2/3$  nahradit konstantami udanými přesněji.

Kontrolní hodnoty:  $Spence(-1,5) = -1,147\ 381$   
 $Spence(0,1655) = 0,172\ 903\ 7$   
 $Spence(2,5) = 2,420\ 791$

Algoritmus byl užíván na EPOS 1 ve strojovém kódu.

#### Literatura

[1] K. Mitchell: Phil. Mag. 40 (1949), str. 351.

## 7. GAUSS

### NORMÁLNÍ ROZLOŽENÍ

Tento algoritmus je převzat z CACM od D. IBBETSONA [1].

Tato procedura počítá hodnoty funkce  $(1/\sqrt{2\pi}) \int_{-\infty}^x \exp(-\frac{1}{2}u^2) du$  na základě aproximace jedním polynomem v intervalu  $\langle 0, 2 \rangle$  a druhým v intervalu  $\langle 2, 6 \rangle$ .

```

real procedure Gauss (x);
value x; real x;
begin real y, z, w;
if x = 0 then z: = 0
else
begin y: = abs(x)/2;

```

```

if  $y \geq 3$  then  $z := 1$ 
else if  $y < 1$  then
  begin  $w := y \times y$ ;
     $z := ((((((( (0.000124818987 \times w$ 
       $- 0.001075204047) \times w + 0.005198775019) \times w - 0.019198292004) \times w$ 
       $+ 0.059054035642) \times w - 0.151968751364) \times w + 0.319152932694) \times w$ 
       $- 0.531923007300) \times w + 0.797884560593) \times y \times 2$ 
    end
  else
    begin  $y := y - 2$ ;
       $z := ((((((((((( ( -0.000045255659 \times y$ 
         $+ 0.000152529290) \times y - 0.000019538132) \times y - 0.000676904986) \times y$ 
         $+ 0.001390604284) \times y - 0.000794620820) \times y - 0.002034254874) \times y$ 
         $+ 0.006549791214) \times y - 0.010557625006) \times y + 0.011630447319) \times y$ 
         $- 0.009279453341) \times y + 0.005353579108) \times y - 0.002141268741) \times y$ 
         $+ 0.000535310849) \times y + 0.999936657524$ 
      end
    end;
   $Gauss := \text{if } x > 0 \text{ then } (z + 1)/2 \text{ else } (1 - z)/2$ 
end

```

Tato procedura byla užívána na počítači ELLIOT, kde dávala výsledky s přesností  $10^{-8}$ .

[1] Ibbetson D.: Algorithm 209. CACM, vol. 6 (1963), str. 616.