

**nag\_poisson\_dist (g01bkc)****1. Purpose**

**nag\_poisson\_dist (g01bkc)** returns the lower tail, upper tail and point probabilities associated with a Poisson distribution.

**2. Specification**

```
#include <nag.h>
#include <nagg01.h>

void nag_poisson_dist(double rlamda, Integer k, double *plek,
                     double *pgtk, double *peqk, NagError *fail)
```

**3. Description**

Let  $X$  denote a random variable having a Poisson distribution with parameters  $\lambda (> 0)$ . Then

$$\text{Prob}\{X = k\} = e^{-\lambda} \frac{\lambda^k}{k!}, \quad k = 0, 1, 2, \dots$$

The mean and variance of the distribution are both equal to  $\lambda$ .

This routine computes for given  $\lambda$  and  $k$  the probabilities:

$$\begin{aligned} \text{plek} &= \text{Prob}\{X \leq k\} \\ \text{pgtk} &= \text{Prob}\{X > k\} \\ \text{peqk} &= \text{Prob}\{X = k\}. \end{aligned}$$

The method is described in Knüsel (1986).

**4. Parameters****rlamda**

Input: the parameter  $\lambda$  of the Poisson distribution.  
Constraint:  $0.0 < \mathbf{rlamda} \leq 10^6$ .

**k**

Input: the integer  $k$  which defines the required probabilities.  
Constraint:  $\mathbf{k} \geq 0$ .

**plek**

Output: the lower tail probability,  $\text{Prob}\{X \leq k\}$ .

**pgtk**

Output: the upper tail probability,  $\text{Prob}\{X > k\}$ .

**peqk**

Output: the point probability,  $\text{Prob}\{X = k\}$ .

**fail**

The NAG error parameter, see the Essential Introduction to the NAG C Library.

**5. Error Indications and Warnings****NE\_INT\_ARG\_LT**

On entry,  $\mathbf{k}$  must not be less than 0:  $\mathbf{k} = \langle \text{value} \rangle$ .

**NE\_REAL\_ARG\_LE**

On entry,  $\mathbf{rlamda}$  must not be less than or equal to 0.0:  $\mathbf{rlamda} = \langle \text{value} \rangle$ .

**NE\_REAL\_ARG\_GT**

On entry, **rlamda** must not be greater than  $10^6$ : **rlamda** =  $\langle value \rangle$ .

**NE\_INTERNAL\_ERROR**

An internal error has occurred in this function. Check the function call and any array sizes. If the call is correct then please consult NAG for assistance.

**6. Further Comments**

The time taken by the routine depends on  $\lambda$  and  $k$ . For given  $\lambda$ , the time is greatest when  $k \approx \lambda$ , and is then approximately proportional to  $\sqrt{\lambda}$ .

**6.1. Accuracy**

Results are correct to a relative accuracy of at least  $10^{-6}$  on machines with a precision of 9 or more decimal digits, and to a relative accuracy of at least  $10^{-3}$  on machines of lower precision (provided that the results do not underflow to zero).

**6.2. References**

Knüsel L (1986) Computation of the Chi-square and Poisson Distribution. *SIAM J. Sci. Statist. Comput.* **7** 1022–1036.

**7. See Also**

nag\_binomial\_dist (g01bjc)  
nag\_hypergeom\_dist (g01blc)

**8. Example**

This example program reads values of  $\lambda$  and  $k$  from a data file until end-of-file is reached, and prints the corresponding probabilities.

**8.1. Program Text**

```

/* nag_poisson_dist(g01bkc) Example Program.
 *
 * Copyright 1996 Numerical Algorithms Group.
 *
 * Mark 4, 1996.
 *
 */

#include <nag.h>
#include <nag_stdlib.h>
#include <stdio.h>
#include <nagg01.h>

main()
{
    double plek, peqk, pgtk;
    double rlamda;

    Integer k;

    Vprintf("g01bkc Example Program Results\n");

    /*      Skip heading in data file */
    Vscanf("%*[^\\n] ");
    Vprintf("\\n      rlamda      k      plek      pgtk      peqk\\n\\n");

    while((scanf("%lf %ld%*[^\\n] ", &rlamda, &k)) != EOF)
    {
        g01bkc(rlamda, k, &plek, &pgtk, &peqk, NAGERR_DEFAULT);
        Vprintf(" %10.3f%6ld%10.5f%10.5f%10.5f\\n", rlamda,k,plek,pgtk,peqk);
    }
    exit(EXIT_SUCCESS);
}

```

**8.2. Program Data**

```
g01bkc Example Program Data
0.75    3      : rlamda, k
9.20    12
34.00   25
175.00  175
```

**8.3. Program Results**

```
g01bkc Example Program Results

   rlamda      k      plek      pgtk      peqk
   0.750      3      0.99271    0.00729    0.03321
   9.200     12      0.86074    0.13926    0.07755
  34.000     25      0.06736    0.93264    0.02140
 175.000    175      0.52009    0.47991    0.03014
```

---