RANDOM NUMBER GENERATOR FOR THE COMMAND LINE
RG 0.4.2 BETA

RUDOLF B. BLAŽEK
RBLAZEK@FIT.CVUT.CZ

1. Proper Use of the Generator to Preserve its Internal Ingegrity

When executed repeatedly from the command line or a script, rg preserves the integrity of its internal random number generator via the option -I (capital i). Generating N random numbers using the -N option is thus the same as calling rg N times with the -I option.

The generator returns a single line of output. It consists of a space-separated list of random numbers, followed by ' :' and a comma-separated list of shuffle table values that are to be used with the -I option during the next call.

```
#!/bin/bash
# The initialization option -I is a capital i, not a small ell.

# Initialization of rg
nextInit=`rg -S 0 | awk -F' :' '{print $2}'`

# Random values in a single variable
out=`rg -I $nextInit -N 6 -D Exponential -m 5`
randomValues=`echo $out | awk -F' :' '{print $1}'`
nextInit=`echo $out | awk -F' :' '{print $2}'`

# do something with the random numbers
echo Random values in a single variable
for value in $randomValues; do
  echo $value
done

# Random values in an array - note the () for randomValues
out=`rg -I $nextInit -N 5 -D Pareto -k 83456 -b 1.1`
randomValues=(`echo $out | awk -F' :' '{print $1}'`)
nextInit=`echo $out | awk -F' :' '{print $2}'`

# do something with the random numbers
numberOfValues=${#randomValues[@]}
echo ""; echo Random values in an array
for (( i=0; i<$numberOfValues; i++ )); do
  echo ${randomValues[$i]}
done
```

Usage of rg

Usage:

```
rg -S seed | -I initialization_vector [-N sample_size]
      [-D distribution_name] [-m mu] [-l lambda] ...
      [-E | -F | -A] [-O output_format] [-V] [-Q]
```

`rg` generates a sample of pseudo random numbers. It supports the following distributions: Uniform on (0,1), Exponential, Erlang, Gamma, Normal, Lognormal, Weibull, Pareto, Geometric, and Poisson.

When executed repeatedly from the command line, `rg` preserves the integrity of the internal generator via the option `-I`. Generating N random numbers using the `-N` option is thus the same as calling `rg` N times with the `-I` option.

Parameters

- **-S** - `seed` of the generator. A non-negative integer value.
- **-I** - `initialization` (capital `I`) of the generator. The argument is a comma-separated list of shuffle table values that were returned with random numbers generated during the previous call.
- **-N** - the `size` of the generated random sample
- **-D** - the desired probability `distribution`; one of: Uniform, Exponential, Erlang, Gamma, Normal, Lognormal, Weibull, Pareto, Poisson
- **-m** - `mu` (it is the mean for the Exponential, Poisson, and Normal distributions)
- **-l** - the rate parameter `lambda` (small `ell`) of the Exponential and Erlang distribution
- **-s** - `sigma squared` - the variance of the (underlying) Normal distribution
- **-a** - the shape parameter `alpha` of the Gamma and Weibull distribution
- **-b** - the scale parameter `beta` of the Gamma and Weibull distribution, the shape parameter `beta` of the Pareto distribution
- **-k** - the `scale` parameter (minimum) of the Pareto distribution, or the `shape` parameter of the Erlang distribution
- **-p** - the `probability` of success for the Geometric distribution
- **-E** - engineering (fixed-point) output format, with C's default number of significant digits (%e). For continuous distributions only.
- **-F** - floating-point output format, with C's default number of significant digits (%f). For continuous distributions only.
- **-A** - automatic output format - preserve as many significant digits as possible, using the '%e' or '%f' format as necessary (%.30g). For continuous distributions only.
- **-O** - specify the output format as for printf in C. E.g. '%.6g'. For continuous distributions only.
- **-V** - `verbose` ouput to stderr, including the `version` of the generator
- **-Q** - `quiet` ouput (suppress output of initialization vector)

Returns

Space-separated list of random numbers, followed by ' :' and a comma-separated list of shuffle table values that are to be used with the `-I` option during the next call.

Version

0.4.2 beta

Author

Rudolf B. Blazek

Copyright

Rudolf B. Blazek

Date

2012

2. Supported Discrete Distributions and their Parameters

In the following sections X will denote a random variable with the discussed distribution.

2.1. **Geometric Distribution.**

$$X \sim \text{Geom}(p) \quad \text{if} \quad P(X = k) = p(1-p)^k \quad \text{for } k = 1, 2, 3, \ldots$$

where $0 < p \leq 1$. The variable X can be viewed as the index of the first successful trial in a sequence of independent Bernoulli trials with probability of success equal to p. We often think of waiting for the first Head in repeated independent coin tosses with $P(\text{Head}) = p$. Then X is the toss number of the first Head.

Geometric distribution is closely related to the exponential distribution, which is often used to model a continuous random waiting time for the next customer. Geometric distribution can be viewed as a discrete version of such waiting time. Both these distributions share many properties, for example they are both memoryless.

```
rg -S <seed> -D Geometric -p <probability>
```

2.2. **Poisson Distribution.**

$$X \sim \text{Poisson}(\mu) \quad \text{if} \quad P(X = k) = \frac{e^{-\mu}\mu^k}{k!} \quad \text{for } k = 0, 1, 2, 3, \ldots$$

where $\mu > 0$ is the mean of X, i.e. $EX = \mu$.

```
rg -S <seed> -D Poisson -m <mu>
```

3. Supported Continuous Distributions and their Parameters

In the following sections X will denote a random variable with the discussed distribution, and $f(x)$ will represent its probability density function (pdf).

3.1. **Continuous Uniform Distribution.**

In general, the distribution is uniform over an interval $[a, b]$, but only Uniform(0,1) is implemented so far. This is the default distribution.

```
rg -S <seed>
rg -S <seed> -D Uniform -a <minimum> -b <maximum>
```

3.2. **Exponential Distribution.**

$$X \sim \text{Exp}(\lambda) \quad \text{if its pdf is} \quad f(x) = \begin{cases} \lambda\, e^{-\lambda x} & x \geq 0 \\ 0 & \text{otherwise,} \end{cases}$$

where $\lambda > 0$ is the rate parameter. The expectation (mean) of X is $EX = \mu = 1/\lambda$.

Exponential distribution is often used to model a continuous random waiting time for the next customer. A related distribution is the geometric distribution. It can be viewed as a discrete version of such waiting time. Both these distributions share many properties, for example they are both memoryless.

```
rg -S <seed> -D Exponential -l <lambda>
rg -S <seed> -D Exponential -m <mean>
```

3.3. **Erlang Distribution.**

$$X \sim \text{Erlang}(k, \lambda) \quad \text{if its pdf is} \quad f(x) = \begin{cases} \frac{1}{(k-1)!}\lambda^k x^{k-1} e^{-\lambda x} & x \geq 0 \\ 0 & \text{otherwise,} \end{cases}$$

where $k \in \mathbb{N}^+$ (a positive integer) is the shape parameter, and $\lambda > 0$ is the rate parameter. The sum of k independent Exp(λ) random variables has the Erlang(k, λ) distribution.

```
rg -S <seed> -D Erlang -k <shape> -l <lambda>
```

3.4. **Gamma Distribution.**

$$X \sim \text{Gamma}(\alpha, \beta) \quad \text{if its pdf is} \quad f(x) = \begin{cases} \frac{1}{\Gamma(\alpha)}\beta^{-\alpha} x^{\alpha-1} e^{-x/\beta} & x \geq 0 \\ 0 & \text{otherwise,} \end{cases}$$

where $\alpha > 0$ is the shape parameter, and $\beta > 0$ is the scale parameter. The expectation of X is $EX = \alpha\beta$. If α is an integer, then the Gamma(α, β) distribution is the same as the Erlang$(k = \alpha, \lambda = 1/\beta)$ distribution. Notice the inverse of the scale parameter.

Caution, sometimes the Gamma distribution is defined with the scale parameter $1/\beta$ as in the Erlang distribution. Make sure that you use the correct scale parameter! If you cannot verify the form of the pdf, you can use the expectation $EX = \alpha\beta$ if its value is known. Then the scale parameter to use with the rg generator is $\beta = EX/\alpha$.

```
rg -S <seed> -D Gamma -a <alpha> -b <beta>
```

3.5. Normal Distribution (Gaussian Distribution).

$$X \sim N(\mu, \sigma^2) \quad \text{if its pdf is} \quad f(x) = \frac{1}{\sqrt{2\pi}\,\sigma}\, e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

where μ is the mean, and $\sigma^2 > 0$ is the variance of the distribution. Caution, some books use the standard deviation σ instead of the variance σ^2 in the normal distribution notation. Make sure that you use the variance σ^2 with rg!

```
rg -S <seed> -D Normal -m <mu> -s <sigma_squared>
```

3.6. Lognormal Distribution.

$$X \sim \ln N(\mu, \sigma^2) \quad \text{if} \quad \ln(X) \sim N(\mu, \sigma^2).$$

The parameters μ and $\sigma^2 > 0$ are the mean and variance, respectively, of the *underlying normal* distribution. Caution, some books use the standard deviation σ instead of the variance σ^2 in the normal distribution notation. Make sure that you use the variance σ^2 with rg!

```
rg -S <seed> -D Lognormal -m <mu> -s <sigma_squared>
```

3.7. Weibull Distribution.

$$X \sim \text{Weibull}(\alpha, \beta) \quad \text{if its pdf is} \quad f(x) = \begin{cases} \frac{\alpha}{\beta}\left(\frac{x}{\beta}\right)^{\alpha-1} e^{-(x/\beta)^{\alpha}} & x \geq 0 \\ 0 & \text{otherwise,} \end{cases}$$

where $\alpha > 0$ is the shape parameter, and $\beta > 0$ is the scale parameter.

```
rg -S <seed> -D Weibull -a <alpha> -b <beta>
```

3.8. Pareto Distribution.

$$X \sim \text{Pareto}(k, \beta) \quad \text{if its pdf is} \quad f(x) = \begin{cases} \beta k^{\beta} x^{-(\beta+1)} & x \geq k \\ 0 & \text{otherwise,} \end{cases}$$

where $k > 0$ is the scale parameter, and $\beta > 0$ is the shape parameter. The parameter k is the minimal possible value of X. The parameter β is sometimes called the tail index.

```
rg -S <seed> -D Pareto -k <minimal_value> -b <beta>
```