In[1]:= **<< RandFile`**

```
Package RandFile version 0.1.3 (last modification: 26/08/2915).

Usage notes:

1) Almost all provided functions require to set a global variable pointing to file with
   random data! This can be done by using SetTrueRandomDataFile function. For example
   SetTrueRandomDataFile["/home/user_name/data/sample_file.bin"] for GNU/Linux systems
   or SetTrueRandomDataFile["/Users/user_name/data/sample_file.bin"] for OS X systems.
   Please mind that it is advised to use this function only once during the session.

2) If you intend to use TrueRandomSequence function you must use
   SetMaxTrueRandomSequenceLength and declare at least one sequence. Currently
   declared sequences can be displayed by calling GetTrueRandomDataMarkers[].
   Once defined, the used maximal length cannot be changed during the session.
```

In[2]:= **Information[#, LongForm → False] & /@ Sort[Names["RandFile`*"]]**

BlockTrueRandom[ex] evaluates expression ex and stores the starting position in the file RandFile`Private`TrueRandomDataFile. The position in the file is restored after the execution. Note that this function works only with expressions using the global variable for pointing to a file with random data. See also BlockRandom built−in Mathematica function.

CloseTrueRandomDataFile[] closes the file assigned by the global variable set using SetTrueRandomDataFile.

GetMaxTrueRandomSequenceLength[] displays the maximal length of the random sequence which can be used during the session.

GetTrueRandomDataFile[] displays the value of the global variable TrueRandomDataFile.

GetTrueRandomDataMarkers[] displays the currently set markers in the data files that are used by TrueRandomDataSequence.

SetMaxTrueRandomSequenceLength[len] declares the maximal length, expressed in bytes, of the true random sequence used during the session.

SetTrueRandomDataFile[fName] sets the global variable TrueRandomDataFile used by the functions for generating random numbers. For example SetTrueRandomDataFile["/home/user/data/random_file.bin"].

TrueRandom[type,range] gives a true random number of type Real, Integer or Complex in a specified range. See built−in Mathematica function Random.

TrueRandomChoice[{e1,e2,...,ek}] gives a true random choice of one of {e1,e2,...,en}.
TrueRandomChoice[{e1,e2,...,ek}, n] gives a list of n true random choices.
TrueRandomChoice[elist, {d1,d2,...,dk}] gives a {d1,d2,...,dk}−dimensional array of true random choices.
TrueRandomChoice[{w1,w2,...,wn} −> {e1,e2,...,en}] gives a true random choice weighted by {w1,w2,...,wn}.
TrueRandomChoice[{w1,w2,...,wn} −> {e1,e2,...,en}, k] gives k true random choices weighted by {w1,w2,...,wn}.
TrueRandomChoice[{w1,w2,...,wn} −> {e1,e2,...,en}, {d1,d2,...,dk}] gives a {d1,d2,...,dk}−dimensional array of true random choices weighted by {w1,w2,...,wn}.

TrueRandomComplex[] gives a true random complex number with real and imaginary parts in the range 0 to 1.
TrueRandomComplex[{min,max}] gives a true random complex
number in the rectangle with corners given by the complex numbers min and max.
TrueRandomComplex[{min,max}, n] gives a list of d true random complex numbers in a given rectangle.
TrueRandomComplex[{min,max}, {d1,d2,...,dk}] gives a
{d1,d2,...,dk}–simensional array of true random complex numbers in a given rectangle.

TrueRandomDataFile is the global variable, defined in RandFile`Private
context, storing a string with the name of the file containing random data.

TrueRandomDataFileBytesCount is the global variable for storing a number of bytes in the TrueRandomDataFile.

TrueRandomDataMarkers is a global variable for accumulating positions in the file with random data used by
TrueRandomSequence function. Note that this array is sorted with respect to the second argument.

TrueRandomInteger[] produces an integer number in [0,1].
TrueRandomInteger[n] produces an integer number in [0,n].
TrueRandomInteger[n,k] produces k integer numbers in [0,n].
TrueRandomInteger[{a,b}] produces an integer number in [a,b).
TrueRandomInteger[{a,b}, k] produces k integer numbers in [a,b).
TrueRandomInteger[{a,b}, {d1,d2,...,dk}]
produces a {d1,d2,...,dk}–dimensional array of integer numbers in [a,b).

TrueRandomInteger[inStream] uses bytes from input stream inStream to produce a integer number in [0,1).
TrueRandomInteger[n, inStream] uses bytes from input stream inStream to produce a integer number in [0,n).
TrueRandomInteger[n, k, inStream] uses bytes
from input stream inStream to produce k integer numbers in [0,n).
TrueRandomInteger[{a,b}, inStream] uses bytes from input stream
inStream to produce an integer number in [a,b).
TrueRandomInteger[{a,b}, k, inStream] uses bytes from input stream
inStream to produce k integer numbers in [a,b).
TrueRandomInteger[{a,b}, {d1,d2,...,dk}, inStream] uses bytes from input stream
inStream to produce a {d1,d2,...,dk}–dimensional array of integer numbers in [a,b).

TrueRandomInteger[PoissonDistribution[μ]] produces random integer from PoissonDistribution[μ].
TrueRandomInteger[PoissonDistribution[μ], k] produces k random integers from PoissonDistribution[μ].
TrueRandomInteger[PoissonDistribution[μ], {d1,d2,...,dk}] produces a
{d1,d2,...,dk}–dimensional array of integer numbers from PoissonDistribution[μ].
TrueRandomInteger[PoissonDistribution[μ], inStream] uses bytes from input
stream inStream to produce a random integer from PoissonDistribution[μ].
TrueRandomInteger[PoissonDistribution[μ], k, inStream] uses bytes from input
stream inStream to produce k random integers from PoissonDistribution[μ].
TrueRandomInteger[PoissonDistribution[μ], {d1,d2,...,dk}, inStream] uses bytes from input stream inStream
to produce ka {d1,d2,...,dk}–dimensional array of integer numbers from PoissonDistribution[μ].

TrueRandomReal[] produces 32−bit real number in [0,1].
TrueRandomReal[b] produces 32−bit real number in [0,b].
TrueRandomReal[{a,b}] produces 32−bit real number in [a,b].
TrueRandomReal[b, n] produces n 32−bit real numbers in [0,b].
TrueRandomReal[{a,b}, n] produces n 32−bit real numbers in [a,b].
TrueRandomReal[b, {d1,d2,...,dk}] produces
a {d1,d2,...,dk}−dimensional array of 32−bit real numbers in [0,b].
TrueRandomReal[{a,b}, {d1,d2,...,dk}] produces a {d1,d2,...,dk}−dimensional
array of 32−bit real numbers in [a,b].

TrueRandomReal[inStream] uses bytes from the input stream inStream to produce 32−bit real number in [0,1].
TrueRandomReal[b, inStream] uses bytes from
the input stream inStream to produce 32−bit real number in [0,b].
TrueRandomReal[b, n, inStream] uses bytes from the input stream
inStream to produce to produce n 32−bit real numbers in [0,b].
TrueRandomReal[{a,b}, inStream] uses bytes from the input stream
inStream to produce 32−bit real number in [a,b].
TrueRandomReal[{a,b}, n, inStream] uses bytes from the input stream
inStream to produce n 32−bit real numbers in [a,b].
TrueRandomReal[b, {d1,d2,...,dk}, inStream] uses bytes from the input stream inStream
to produce a {d1,d2,...,dk}−dimensional array of 32−bit real numbers in [0,b].
TrueRandomReal[{a,b}, {d1,d2,...,dk}, inStream] uses bytes from the input stream inStream
to produce a {d1,d2,...,dk}−dimensional array of 32−bit real numbers in [a,b].

TrueRandomReal[NormalDistribution[μ,σ]] produces 32−bit real number from NormalDistribution[μ,σ].
TrueRandomReal[NormalDistribution[μ,σ], n] produces n 32−bit real numbers from NormalDistribution[μ,σ].
TrueRandomReal[NormalDistribution[μ,σ], {d1,d2,...,dk}] produces a
{d1,d2,...,dk}−dimensional array of 32−bit real numbers from NormalDistribution[μ,σ].
TrueRandomReal[NormalDistribution[μ,σ], inStream] uses bytes from the input
stream inStream to produce 32−bit real number from NormalDistribution[μ,σ].
TrueRandomReal[NormalDistribution[μ,σ], n, inStream] uses bytes from the input
stream inStream to produce n 32−bit real numbers from NormalDistribution[μ,σ].
TrueRandomReal[NormalDistribution[μ,σ], {d1,d2,...,dk}, inStream] uses bytes from the input stream inStream
to produce a {d1,d2,...,dk}−dimensional array of 32−bit real numbers from NormalDistribution[μ,σ].

TrueRandomSample[l] gives a true random permutation of the list l.
TrueRandomSample[l,n] gives n elements from the true random
sample of the list l. Note that it is possible to take at most Length[l] elements.
TrueRandomSample[{w1,w2,...,wn} −> {e1,e2,...,en}, k] gives k elements from the
non−uniform sample of elements {e1,e2,...,en} with weights {w1,w2,...,wn}.

TrueRandomSequence[] changes the current position in the file assigned
by global variable TrueRandomDataFile so that the numbers generated after the
execution of this function will not overlap with the previously generated numbers.
TrueRandomSequence[pos] marks the current position in the file with random data
or, if the position pos is already marked, returns to this position in the file.

Out[2]= {Null, Null, Null, Null, Null, Null, Null,
Null, Null, Null, Null, Null, Null, Null, Null, Null, Null, Null}