Note: It is often desirable to generate a really large number of random numbers, in order to get good statistics. If you generate all of them in one large array, you are likely to use up most or all of the system memory, which will affect not only you, but also others that are working on the system.

To see how much memory you use, you have the choice between
(a) checking from IDL with

```
print, (memory())[0]/2.^20    ; print IDL's memory usage in MB
N=400 & a=fltarr(N,N,N)
print, (memory())[0]/2.^20    ; ditto
a=0.
print, (memory())[0]/2.^20    ; ditto
```

(b) checking from the shell:

```
top
# (then press 'M' and look for the column labelled '%MEM')
```

To avoid over-using memory by accident, you can start IDL using

```
(ulimit -v 300000; idl)
```

As a long-term solution, replace the alias for idl in your '`/.bashrc`' by

```
alias idl='(ulimit -v 300000; rlwrap -a -c /opt/rsi/idl/bin/idl)'
```

Do not forget to source your '`/.bashrc`', then try it out with

```
N=500 & a=fltarr(N,N,N)
```

Question 1 *Uniformly distributed random numbers*

(a) Use the `randomu` function to generate 10 000 uniformly distributed (pseudo) random numbers u_i.

(b) Calculate the mean value and standard deviation and compare to expectation value and variance of the distribution. Does the mean get more precise if you increase the sample count?

(c) Calculate the correlation coefficient $\varrho(u_{0..499}, u_{500..999}$. What can you conclude about the quality of the random number generator?

(d) Using `histogram()` and a plot style `PSYM=10`, plot a histogram of the PDF for your random numbers. Make sure the x axis and the normalization are correct.

(e) Write a function `randomu2(seed,rand,x0,width)` that implements the uniform distribution with offset x_0 and width w. Use it to plot the histogram of the PDF for $\mathcal{U}(-3, 5)$.

(f) Plot histograms of the sum of two, three and four uniformly distributed random numbers.

Question 2 *Normal distribution*

(a) Write a function `randomnormal()` that returns an array of (pseudo) random numbers x_i that are distributed according to the normal distribution $\mathcal{N}(0, 1)$ (use the special transformation method from the lecture).

(b) Determine the average and the standard deviation and plot a histogram of the PDF.

(c) Compare these results to random numbers obtained using the built-in function `randomn()`.

Question 3 *Superposition method*

We want to generate random numbers X with the probability density function

$$f_X(x) = \frac{3}{8}(1 + x^2)\, x \qquad -1 \le x \le 1\ . \tag{1}$$

(a) Can we use the transformation method?

(b) Use the superposition method to write a generator for this distribution. Plot a histogram of the PDF.

(c) Now use the acceptance-rejection method with a constant comparison function to write another generator.

Question 4 *Birthday problem*

In a group of 25 people with randomly distributed (and independent) birthdays, how large is the probability that two of the birthdays are on the same day?

[Do this numerically]