Generate a random array

$\text{appVersion}(4) = \text{"0.99.7691.4821"}$

```
RND(m, n) := | [ $M := 0$  $r := [1..m]$  $c := [1..n]$ ]
             | $M_{r\,c} := 10^{-9} \cdot \text{random}\left(10^{9}\right)$
```

and simple two test function

```
f(M) := | norme(M)

g(M) := | norme(M)
        | norme(M)
```

$A := RND(100, 100)$

$to := \text{time}(0)$ $\quad f(A) = 57.4732$ $\quad \text{time}(0) - to = 0.242\ \text{s}$

$to := \text{time}(0)$ $\quad g(A) = 57.0345$ $\quad \text{time}(0) - to = 0.229\ \text{s}$

Two questions:
1) Why are different?
2) Why g is always faster than f?

Well, are different because A isn't parsed as a true numeric array. Using eval now are the same:

$A := \text{eval}(RND(100, 100))$

$to := \text{time}(0)$ $\quad f(A) = 57.3581$ $\quad \text{time}(0) - to = 0.216\ \text{s}$

$to := \text{time}(0)$ $\quad g(A) = 57.3581$ $\quad \text{time}(0) - to = 0.195\ \text{s}$

Remain the second question.

BTW, we now are sure that with eval( ) we can parse A as a numberic entity, and can't do nothing for speed up a procedure involving A. NDTM Amarasekera shows that's false. Here an example

```
h(M) := | $m := M_1$
        | for $k \in [2..\text{length}(M)]$
        |   if $m < M_k$
        |     $m := M_k$
        |   else
        |     continue
        | $m$

h'(Mo) := | $M := Mo$
          | $m := M_1$
          | for $k \in [2..\text{length}(M)]$
          |   if $m < M_k$
          |     $m := M_k$
          |   else
          |     continue
          | $m$
```

$A := \text{eval}(RND(75, 75))$

$to := \text{time}(0)$ $\quad h(A) = 0.9996$ $\quad \text{time}(0) - to = 32.273\ \text{s}$

$to := \text{time}(0)$ $\quad h'(A) = 0.9996$ $\quad \text{time}(0) - to = 0.71\ \text{s}$

Alvaro