

# Advanced R programming: practical 4

Dr Colin Gillespie

## Reference classes

The example in the notes created a random number generator using a reference class.

- Reproduce the randu generator from the notes and make sure that it works as advertised.<sup>1</sup>
- When we initialise the random number generator, the very first state is called the seed. Store this variable and create a new function called `get_seed` that will return the initial seed, i.e.

```
r = randu(calls=0, seed=10, state=10)
r$r()

## [1] 0.0003051898

r$get_state()

## [1] 655390

r$get_seed()

## [1] 10
```

<sup>1</sup> The reference class version, not the function closure generator.

Reference classes also have an initialise method - that way we would only specify the seed and would then initialise the other variables. I'll give you an example in the solutions.

- Create a variable that stores the number of times the generator has been called. You should be able to access this variable with the function `get_num_calls`

```
r = randu(calls=0, seed=10, state=10)
r$get_num_calls()

## [1] 0

r$r()

## [1] 0.0003051898

r$r()

## [1] 0.001831097

r$get_num_calls()

## [1] 2
```

## *Solutions*

Solutions are contained within the course package

```
library("nclRadvanced")  
vignette("solutions4", package="nclRadvanced")
```