Exercise 4

This exercise involves manipulating some of **mirtCAT**s more adavanced functionality, ultimately leading to a tailored *shadowCAT* design with a number of item and person constraints.

Use the following code to define the required IRT objects for this exercie.

```r
library('mirtCAT')

set.seed(1234)
bank <- 300 #bank size
a <- rlnorm(bank, .2,.3)
d <- rnorm(bank)
pars <- data.frame(a1=a, d=d)
mod <- generate.mirt_object(pars, itemtype = '2PL')
pattern <- generate_pattern(mod, Theta = 0)
```

Question 1

Try building your own `customNextItem()` function and supplying it to `mirtCAT()`. In this selection scheme, select the items that provide the most information given the current ability estimate until 30 items are administered. However, instead of always picking the most optimal item,

- randomly sample from the top 5 items instead and using the one that has been selected. (hint: see `?findNextItem` with the `all_index` argument), and
- ensure that the first 15 items are selected from the first 150 items in the item bank, while the last 15 items are selected from the remaining 150 items in the item bank (hint: see the `subset` argument for `findNextItem()`)

Question 2

Next, lets create a customized map for updating the latent trait parameter. Specifically, for the first 5 selected items compute $\hat{\theta}$ with the EAP criteria, followed by MAP for the next 5 selected items, and finally ML for the remainder of the CAT session. Combine this extension with the previous item selection method in Question 1. To help you get started, here's a quick template

```r
mythetafun <- function(design, person, test){
    # ......
    # ADD CODE HERE
    browser() # comment out when ready
    # ......
    thetas <- fscores(test@mo, response.pattern = person$responses,
                      method = "FIXME") # change FIXME
    person$thetas <- matrix(thetas[,'F1'], 1L)
    person$thetas_SE_history <- rbind(person$thetas_SE_history,
                                      thetas[,'SE_F1', drop=FALSE])
    person$thetas_history <- rbind(person$thetas_history, person$thetas)
    invisible()
}
```

Question 3

This question includes prior information which describe characteristics about the items that test administrators may want to consider in the item selection process. Specifically,

1) The type of item response stimuli (e.g., true-false, multiple-choice, fill-in-the-blank),
2) The content category (e.g., addition or subtraction), and
3) Average time it takes to answer the item (in seconds)

```
test_properties <- data.frame(stimuli=rep(c('TF', 'MC', 'fill'), each=bank/3),
                              content=rep(c('add', 'sub'), times=150),
                              time=sample(10:30, bank, TRUE))
```

Given these test properties, create a constrained yet optimal item selection design whereby items are selected based on the maximum information criteria subject to the following constraints:

- exactly 30 total items administered
- between 3 to 5 fill-in-the-blank items only
- items 5 and 6 should not appear in the same test (though neither must be)
- items 10 and 11 should not appear in the same test (though *exactly* one must be)
- item 17 and 18 should appear in the test no matter what
- item 15 should not appear
- the same number of addition and subtraction questions should be included
- the average expected time it takes to answer all the questions should be less than 20 minutes

This example will require the use of a constraint function, `computeCriteria()`, `findNextItem()`, and `customNextItem()`. After the shadowCAT is complete, inspect the results to determine whether all the constraints have been met. (Note: to ensure the test is not terminated too early, set `min_SEM = 0` or `min_items = 30`).