ST661: Exam December 2023 Part B

Place your name and student number here

- Download fin23.Rmd from Moodle and fill in your answers.
- Place your name and student number in the space for `author:`.
- Upload the html file produced by knitr to Moodle before 3.20pm. There will be a penalty of 10 marks for not uploading the html file.
- Do not include in your html file long data listings. There will be a 5 mark penalty for this.
- If your code for any answer does not run, use `{r, eval=FALSE}` for your code chunk. If the answer to a question relates to the answer to a previous question that you did not complete, you may still give code in an `eval=FALSE` code chunk.

Question 1 (48 marks, 8 marks per part)

The file provided on Moodle `children.Rdata` gives information on the age, height, weight and BMI of children measured multiple times. Download the data and place in the same folder as your Rmarkdown file. Access the data with

```
load("children.Rdata") # gives dataset called kids
```

Write code for each of the following. You can use base R, tidyverse or a mix.

1a. Change the class of the date variable to be class `POSIXct`. Show the result of `glimpse(d)` to verify your code.

1b. Find the IDs of the children whose age.bin is 12 at the start of the study (the earliest date)

1c. For the children whose age.bin is 12 at the start of the study, plot `height` versus `date` as a line plot. Colour the lines by gender.

1d. Make a subset of the data consisting of kids whose `age.bin` is 12. For children who appear multiple times in this subset, remove all but the first occurrence. Call this subset `kids12`. Show the result with `glimpse(kids12)`.

1e. Using `kids12` , find the proportion of boys and girls in each of the `BMIcat` groups. Omit those whose `BMIcat` is `NA` from the calculation.

1f. Make a barplot showing the proportion of boys and girls in `kids12` in each of the BMI categories. Omit those whose `BMIcat` is `NA`. Make sure that the `BMIcat` labels are in order of increasing `BMIcat`.

Question 2. (18 marks)

Write a function called `runzero` that given a numeric vector return the length of the longest consecutive subsequence of zeros. Run the following to test your answers.

runzero(c(4,5,6,1))

runzero(c(4,0,0,0,6,0,0,0,0))

runzero(c(0,0,3,3,3,3))