# Stroop Experiment

1. Experiment overview:

The Stroop effect was revealed in a cognitive psychology experiment first conducted by J. Ridley Stroop (Stroop 1935/1992). It is related to selective attention, which is the ability to respond to certain environmental stimuli while ignoring others. It is very easy to name the colour of the word ‘red’ when it is printed in red ink colour. It is difficult, though, when the word and the ink colour are different!

The Stroop Task is frequently used to measure how well people can do something that clashes with their typical response pattern. This task requires a certain level of "mental control". That is, you need to be aware of the task you are doing now and ignore how you would normally respond to words. This requires ‘control’ over your own default cognitive processing. The Stroop effect is essentially the degree of difficulty people have with naming the colour of the ink rather than the word itself. In Stroop’s words, there is ‘interference’ between the colour of the ink and the word meaning. This interference occurs no matter how hard you try. It implies that at least part of our information processing occurs automatically, outside of conscious control.

One of the explanations for the difficulty is that we are so used to processing word meaning while ignoring the physical features of words, that it is a learned response. The Stroop task requires us to do something which we have never learned and which is opposite what we normally do.

In this online experiment, I attempt to replicate a simplified version of the classic Stroop study as efficiently as possible.

2. Description of procedure:

Participants are presented with the name of a colour presented in a colour font, and they must name the colour of the font.

There are two conditions:

Congruent: In this condition the colour name and the colour of the font are the same.  For example when presented with **BLUE**, the correct answer is "blue."  Participants will probably respond quickly because the word and the font colour match.

Incongruent: In this condition the colour name and the colour of the font differ.  For example, when presented with **BLUE**, the correct answer is "red."  Participants take longer to respond because the word and the font colour do not match; reading the word interferes with identifying the font colour.

During the experiment, the participant presented with a number of words in the two conditions.  For each one, he/she will need to identify the font colour of the word.  The program will automatically record reaction time.

3. Experimenter's manual:

This is a purely online experiment, and you are recommended to run it as a quantitative study.

There are 3 checks to be made during study set-up.

1. Determine how many colours you want to test and what are the colours you want in your study. You can choose from the following QT colours: [**red, green, blue, cyan, magenta, yellow, grey, white]**. To input the colour, just insert the string name as per the list above in line 21.
   * For example: expColours = [‘red’, ‘blue’, ‘green]
2. Determine how many trials you want the respondent to go through in total and type in the correspondent integer in line 24. Example
   * For example: trialNum=20
3. Account for the key presses due to the colours
   * Line 229: if keyPrs[-1] == 'r' or keyPrs[-1] == 'b' or keyPrs[-1] == 'g'
   * At the end just insert more keyPrs[-1] == '(First alphabet of Colour)' depending on the no. of colours chosen.
   * Same for Line 250: ui.error.setText ("ERROR: Please press either 'r', 'b', or 'g' key!") 🡪 Add on the correspondent colour alphabets

4. Program highlights:

* Opening animation to peak interest in respondent
* Have tried to make this program as easy for the experimenter as possible
  + All files and folders are automatically generated if absent.
* Easy modification:
  + Experiment instructions will modify automatically based on conditions determined by experimenter
  + Almost all of the experiment code will not require modification when changing conditions.
    - Colours tested and no. of trials can be changed at top of code.
    - Immediate random generation of the word / colour flashed
    - Only need to add in 2 additional lines to account for new key press
* Error label on demographic page gives specific information on what is wrong with response
* Layout is simple but intuitive for respondent to follow.
* Key pressed is auto-converted to small caps just in case respondent caps lock is on. Burden of correct set-up is taken away from respondent.
* Data rendering has been designed to ease burden on experimenter
  + Individual results files are tidily collated in a folder.
  + File Name is ordered based on Participant Number and Name of Participant. This will enable convenient data checking later on.
  + A new folder is automatically created if experiment condition changes. For example, all the 3 Colour, 20 Trial results will be in one folder. Meanwhile, 4 Colour, 30 Trial results will be in a separate folder.
  + A results summary sheet is created to collate the summary data. The program automatically does the results tabulation and generates simple statistics like mean. The experimenter can review data trend without needing to go through individual results later on.
  + Like above, if experimenter chooses to modify conditions, a new results summary sheet is automatically created upon program initialization.
* Informative data provided to participants
  + The experiment results page shows the participant both their individual result as well as the sample average.
  + A boxplot is automatically generated so they know the spread + where they stand amongst the sample population.
  + **Note:** The boxplot and sample average data is only generated if more than one participants exist. As such, I have provided four sample respondent results so that the full capabilities of the program can be visualized