Project 3: Stroop Task

One of the most-studied constructs in psychology is self-regulation, loosely defined as the ability to control one's own behavior. Behavioral tasks that purport to measure self-regulation vary widely in their structure, from simple reaction-based tasks like the Eriksen Flanker Task and the Stroop Task to the famous "Marshmallow experiment" in which young children are told that if they don't eat a delicious marshmallow now, they will be rewarded with two marshmallows later. Reaction time tasks are used (controversially, perhaps) as behavioral markers of mental disorders, and "discounting" tasks like the Marshmallow experiment have been used to predict subsequent life success and vulnerability to harmful behaviors such as drug abuse. In short, behavioral scientists care about self-regulation.

As mentioned, many tasks measure self-regulation on varieties of real or hypothetical time scales. A common form of this task is to ask the participant to make a judgment quickly in the presence or absence of conflicting information. When information on what to do is clear and obvious, people are very good at simple tasks like pressing a particular button, reading a particular word, and stating the color of a particular object. However, when a stimulus indicates that a particular response is required, and that stimulus is presented with other stimuli which indicate a different response, most people slow down considerably, leading to a difference in *response time* (the time required to make a decision and act on it) between the two conditions called an "interference" or "conflict" effect.

As an example, consider two separate tasks: reading words which indicate colors (for example, saying "blue" when presented with the text "blue") and saying the colors that words are displayed in (for example, saying "blue" when presented with the text "brouhaha". These tasks are inherently different; people are trained from a very young age to read words, making reading a fairly automatic process that requires little effort, whereas naming the colors of things is a relatively rarer task which, while not too difficult, isn't as ingrained in your behavior.

Now consider what happens when these two tasks conflict: specifically, when words indicate different colors than they are displayed in. For example, read the words below out loud, starting with "blue", or to yourself in your head:

blue red green purple brown

Now, read the *color in which these words are displayed, not the words themselves*, starting with "blue"

baked parrot golds rag beef

Now name the colors, starting with "purple":

green red brown blue purple

Harder, huh? The conflict between the color and the word, combined (or so researchers believe) with the reflex-like tendency to read words, makes it much harder to state the colors *in which the words are displayed* when they conflict with the colors *indicated by the words*.

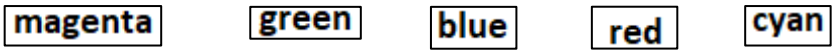
Note: The hardest part of *talking about* the Stroop task is distinguishing between the *colors indicated by the words* and the colors *in which the words are displayed*. If you have questions about Project 3, and you want to clearly distinguish between the two, use this language, otherwise I may ask for clarification.

Project 3 Instructions

In the Stroop (1935) paper (link in the Syllabus), the words were presented on a large poster board to be read aloud by participants. For Project 3, you will program the Stroop Task to display the words in Figure windows in MATLAB and the user will respond by typing into the Command Window using the `input` function. The task will progress as a series of stimuli presented simultaneously with the input prompt: an image containing a word displayed in a color will be presented in a (docked) Figure window (more on this later), and at the same time a prompt will appear asking the user the name of the color of the word. After a response is submitted by the user, the console will clear (`clc`), a word displayed in a color will be presented in the same Figure window, and a new prompt will appear. Unlike in the Stroop task, all colors in which words are displayed will conflict with those indicated by the words; in short, unless you do the extra credit, you will only be presenting one condition (conflicting condition) of the Stroop task.

Images and Colors (8 points)

The images you will use, which also indicate the colors you will use, are copied here (3 points):



These pictures are on CCLE. There are no borders in the CCLE images, those are just for display here.

These colors are nice because they can be created by setting RGB values to zero and one. Here are those values. Note that while the background of the above images is white, the words themselves are in black, which means you have to get creative with how to change these colors programmatically. Also, these images are not only black-and-white; they have small amounts of color in them, which makes this task a bit more difficult.

RGB Triplet	Short Name	Long Name
[1 0 1]	m	magenta
[0 1 1]	c	cyan
[1 0 0]	r	red
[0 1 0]	g	green
[0 0 1]	b	blue

To help you out, I've provided some brief starting code for this project, provided below and on CCLE:

```
clear
close all
nTrials=10;
blue=imread('blue.png');
%use this line only on desktop MATLAB
%comment out or delete for Online or Mobile
f=figure('WindowStyle','docked');
%loop
for i=1:nTrials
    clc %clear console
    %word used on this trial
    word=blue;
    %change its color: remember, R, G, B
    word=255-word;
    word(:,:,3)=0;
    word=255-word;
    %show the word
    imshow(word)
    %collect input
    input('What color?','s');
end
%we're done!
close all
```

A few things to note:

- The sixth line is needed *only* if you are working in installed, desktop MATLAB. It is not needed on the mobile platforms, and should be commented out if those are used, but please leave the (commented) code in your file. Whether it is commented out in your submission will not affect your grade, but leaving it in will make grading slightly easier (we won't have to copy it in).
- To change the black text to white, I needed to first reverse all color levels, then set the blue value to 0 everywhere, then reverse all color values again. You'll need to do similar things for the other colors.
    - Also note, the table above gives RGB values from `[0 1]`, but here I'm using `[0 255]`. For Project 3, either is fine, appearance is what matters.
- Here, only blue is displayed, and it is displayed in blue. You'll, of course, need to change this. The code `randi(5)` generates a random integer from 1 to 5, and combined with a conditional, this will help you select a color for either the word itself or the display color, after which you must randomly select a *different* color for the other aspect of the stimulus (2 points). Use the relevant image file (I'd read them all in before the loop starts), manipulate its colors by modifying RGB values or using some other method (3 points), and you're done with the display portion.

Data Collection (4 points)

In addition to displaying stimuli, a computerized experiment will collect data from participants, and the remaining 4 points of Project 3 require collecting data. Specifically, your code should create a `table` called `data` which contains four variables:

- `word`: Presented word (`cell` array of `character` vectors) (1 point)
- `color`: Color of word (`cell` array of `character` vectors) (1 point)
    - Please use lower-case for both `word` and `color`
- `correct`: Whether response was correct (1) or incorrect (0) (1 point)
    - Assume input is all lower-case, no spaces, no punctuation, i.e. do not take extra steps to ensure that the participant gives a valid response
- `rt`: Response time in seconds, from `tic` and `toc` functions (1 point)

Extra Credit: Reproduce the Stroop (1935) Experiment 2 (4 points)

In addition to displaying stimuli and collecting data, a *true* experiment will have multiple conditions such that you can compare performance and see the effect of your manipulation. Stroop's Experiment 2 is just such an experiment. In that experiment, some trials were presented as conflicting word-colors (as above) and some trials were presented as simply boxes of colors to name. Specifically, Stroop's Experiment 2 had ***N**aming **c**olor of **w**ord test where the color of the display and word are **d**ifferent* (**NCWd**) and ***N**aming **c**olor test* (**NC**)

Stroop's Experiment 2 also had a specific way the stimuli were ordered:

- 100 trials of **NC**, then 100 of **NCWd**, then 100 of **NCWd**, then 100 of **NC**
    - 2nd 100 trials of each test were reversed versions of 1st 100 trials of that test
- Within **NCWd**…
    - No word or color appears twice in a row (but blue can follow green)
    - Each color and word (but not each combo) appears the same number of times
    - **NC** has same color order as **NCWd** (for each 100 trials: 1st half matches 1st half, 2nd half matches 2nd half), but colors are displayed as squares.

For extra credit:

1. Pre-allocate `data` (fill in `word` and `color`) before the loop (2 points) based on those constraints
    - Set initial values of `correct` and `rt` to `NaN`
    - Create a new variable `condition` (`cell` of `characters`) indicating `NCWd` or `NC`
        - Use the order of rows in `data` to determine which stimulus is presented on each trial
    - Order rows of `data` in accordance with Stroop (1935) **(programmatically, no hard-coding!)**
    - 400 total trials, as described above.
2. When displaying NC condition, fill the entire box (**from that same word's image file**) with the randomly chosen color. (2 points)

Grading Notes

- This project only requires the creation of the variable `data`, as described above, but of course you can use as many other variables as you like. I will also be running your code and grading on the structure of the experiment.
- If you are colorblind or otherwise have difficulty seeing the stimuli, please contact me and I will either (a) tailor the project to your specific needs, such that you complete an equivalent project, or (b) pre-grade whatever aspects of the assignment I must in order to ensure you can complete it.
- **If you only need a small number of points to get the grade you want, you are welcome to only complete that much of the project, but I would recommend completing a little more so that you have a buffer in case you made a mistake. If you need help figuring out how much to do, ask!**