Started on	Saturday, 12 September 2020, 3:17 PM
State	Finished
Completed on	Wednesday, 16 September 2020, 4:38 PM
Time taken	4 days 1 hour
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question **1**

Complete

Mark 1.00 out of 1.00

Aims

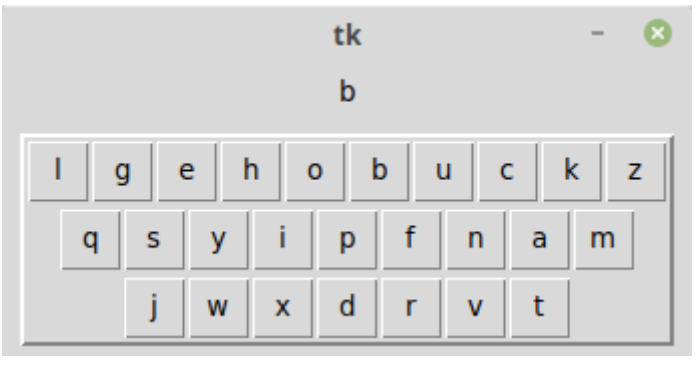
In Lab 2 you programmed a Python/TkInter keyboard GUI that logged the time taken to select a key in response to a key-prompt. In Lab 5 you analysed pointing data to determine your Fitts' law *a* and *b* parameters; these parameters allow you to predict your pointing time for targets of known distance and width. In this lab you will use logs generated by the random keyboard GUI (which randomises all key locations after every selection) to determine how much time is taken to visually search for targets. You will also use the keyboard GUI in its static setting (i.e., where the user doesn't know the key locations to begin with, but they remain static allowing you to learn them) to examine how visual search is replaced with rapid spatial decision as the user gains experience.

At the end of this lab you should understand the basics of the following:

- designing and conducting experiments to measure visual search time and decision time;
- that visual search is slow compared to rapid spatial decisions;
- that spatial stability is a critical factor in supporting rapid interaction;
- that learning curves are steep; users rapidly capitalize on stability;
- that there is more to target acquisition than motor movement alone.

Recap: The Keyboard GUI

Your keyboard GUI should look similar to the one shown below. The key buttons should be 32 pixels square.



The GUI should have two states: 'dynamic', in which all keys are randomly relocated after every correct selection; and 'static', in which the key locations are initially randomised but remain in the same location after every selection. Six target keys should be randomly selected from the alphabet, and these six letters should be exposed to users in six randomly ordered blocks (i.e., the user makes the first selection of each of the six letters; then a second selection of each of the six, and so on). Your program should record a log of selections with the letter in the first column, the block number (0 to 5) in the second column, and the time from cue exposure to selection in the third column, as follows:

```
c  0  2.46
```

If you were unable to complete the keyboard GUI, ask your tutor and they will make a copy available to you.

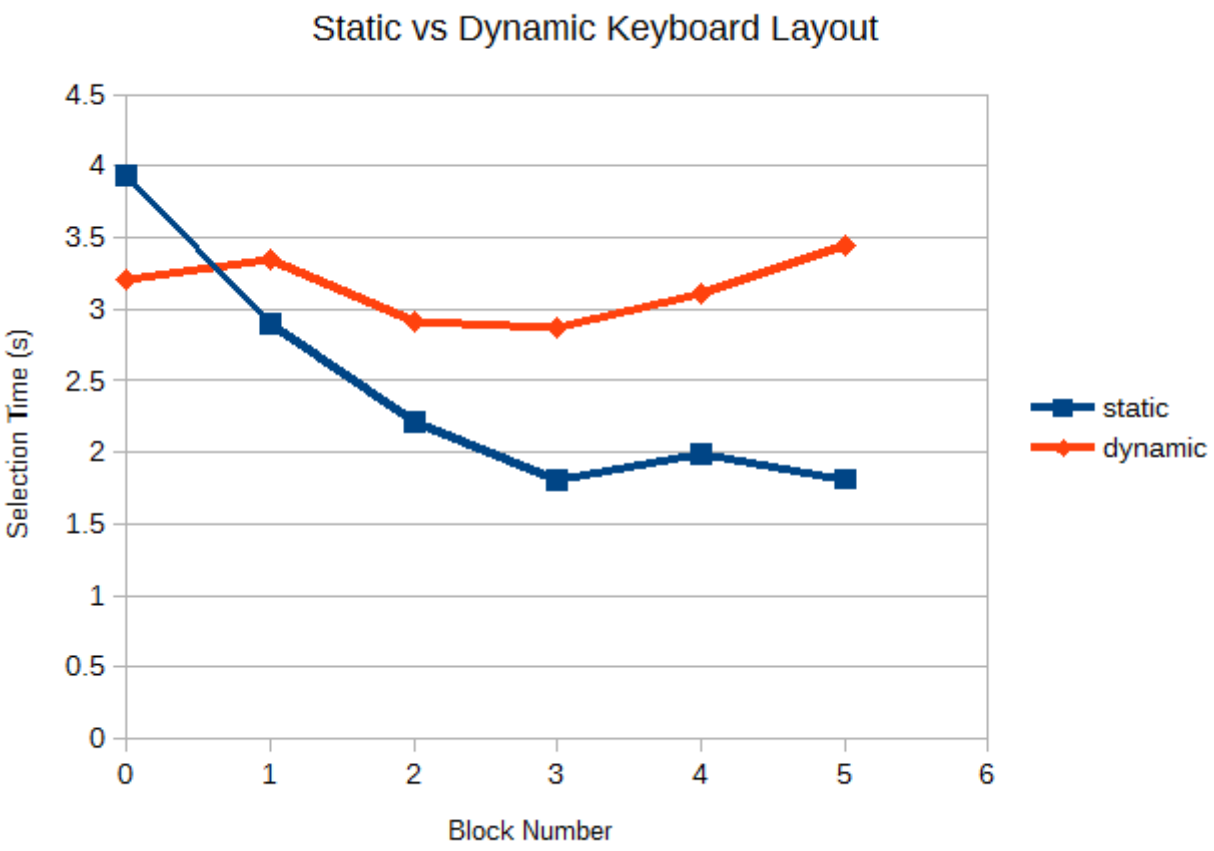
Run the GUI to generate data in static and dynamic conditions

Run the GUI in the static and dynamic conditions. If your birthday is on an even day of the month run the static condition first; otherwise run the dynamic condition first. Concentrate, and try to click the targets as quickly and accurately as possible.

After completing both conditions, reflect on the experience. Which condition felt faster? Did you notice any change in performance across blocks in either condition?

Analyse your data in Excel

Import your data for the two conditions into Excel. Calculate a mean time for each condition in each block, and plot the results using a line chart such as that shown below



Estimate predicted pointing time using Fitts' law

Use the *a* and *b* parameters established from last week's Fitts' law analysis to calculate a mean predicted pointing time. A simple distance estimate can be established by assuming the average distance to target is a quarter of the width of the keyboard (mean target distance from the keyboard center to targets).

Estimated pointing time should be somewhere around 600ms.

From this analysis you should observe that most of the time spent acquiring targets is consumed by other activities than moving to the target (i.e., visually searching for the target, or deciding about its location).

Combine data from others in the class in Excel

Much more representative means can be determined by combining data with others in the class. Do so, and see how the lines in the plot above smooth out. Eventually you should see fairly constant performance across blocks for the 'dynamic' condition, and a learning curve (steep at first, then asymptoting) for static.

Add 'Power' trendlines to each of the Excel plot lines, and include the equation and R^2 values. You should find that the static condition follows a traditional learning curve while dynamic does not.

Upload your Excel file to Learn.

Confounds?

What are the limitations in this study? How might the scope of the findings have been broadened (to give richer insights into interaction), and what aspects of the experimental method are sources of inaccuracy and error?

lab6.xls

Comment:

◄ Lab 5: Sketching Interactions Jump to... keyboard_src.zip ►