MIDDLE EAST TECHNICAL UNIVERSITY, NORTHERN CYPRUS CAMPUS
CNG445 Software Development with Scripting Languages

Assignment 1: A Simple Eye-tracking Data Analyser for People with Autism

This assignment aims to help you practice Python basics including primitive types, variables, operators, decision making, loops, sequences, dictionaries, functions, modules, regular expressions, exception handling, command line arguments, and file processing. Your main task in this assignment is to develop a simple eye-tracking data analyser for people with autism.

Overview

Autism Spectrum Disorder (ASD) is a neurodevelopmental disorder characterised by differences in communication and social interaction, and previous research studies show that people with autism tend to have different processing strategies while interacting with web pages (Eraslan et. al. 2019). In this assignment, you will develop an application that analyses the eye-tracking data of people with and without autism recorded while they are interacting with web pages.

Eye tracking is the process of recording where people look at and how long they look at a particular spot. When people interact with web pages, their eyes become relatively stable at certain points called *fixations*, and the series of these fixations show their *scanpaths*. Eye-tracking data recorded on a web page is typically analysed based on the visual elements of the web page, especially for identifying which visual elements are frequently used, and in which order. Figure 1 shows a scanpath of a single user on a web page segmented into its visual elements where his/her fixations are represented as circles and visual elements are represented by blue boxes.

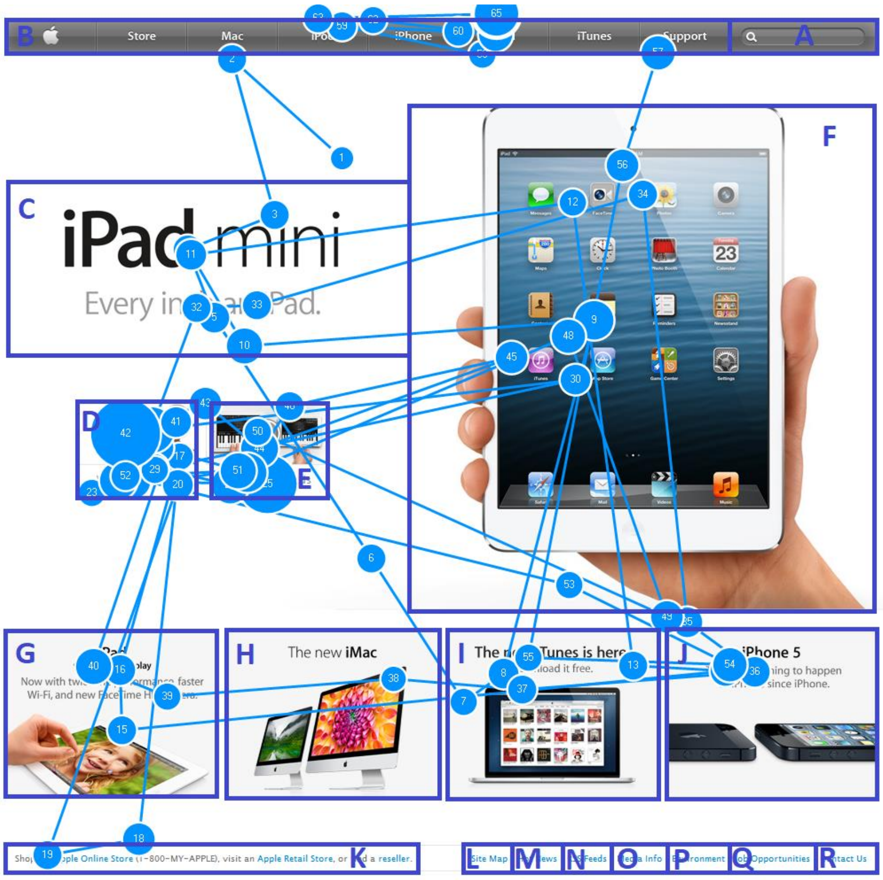


Figure 1: A scanpath of a user on a web page segmented into its visual elements

The application that you will develop in this assignment will process the txt data file in the following format:

```
PageName;ElementName;UserID;UserGender;UserGroup;TimeViewed;Fixations;Revisits
Apple;A;3;F;CONTROL;8.176;23;8
Apple;A;4;M;CONTROL;8.624;29;9
Apple;A;5;M;CONTROL;7.52;37;17
Apple;A;7;F;CONTROL;4.944;20;9
Apple;A;8;M;CONTROL;7.728;21;5
Apple;A;9;M;CONTROL;9.008;32;13
Apple;A;10;M;CONTROL;7.52;28;14
Apple;A;12;M;CONTROL;5.12;25;16
Apple;A;13;F;CONTROL;8.944;36;11
…
```

Column Name	Explanation
PageName	The name of the web page
ElementName	The name of the visual element of the web page
UserID	The identifier of the person
UserGender	The gender of the person – M: Male, F: Female.
UserGroup	The group that the person belongs to – the group of people with autism is ASD and the group of people without autism is CONTROL
TimeViewed	The total amount of the time spent by the person for each element on the web page.
Fixations	The number of fixations of the person for each element on the web page
Revisits	The number of times the person revisits each element on the web page

The application should support the following functionalities:

- Compare the total time viewed, the total number of fixations or the total number of revisits for people with and without autism **for a particular element on a specific web page**
- Compare the total time viewed, the total number of fixations or the total number of revisits for people with and without autism **on a specific web page**

Implementation Requirements

This application should receive the data file name as a **command-line argument** and then prepare a dictionary as it reads data from a file. It means that the keys of the dictionary should not be created first and then filled. Instead, the dictionary is created while reading the file. The structure of the dictionary should be as follows (see Figure 2) where the rounded boxes show the dictionaries and the rectangles in the rounded boxes show the dictionary keys.

As can be seen from Figure 2, this dictionary should keep the total time viewed, the total number of fixations and the total number of revisits for people with autism (ASD) and people without autism (CONTROL) for each element for each web page. For example, the dictionary should keep the total number of fixations and the total number of revisits for the element A on the Apple page.

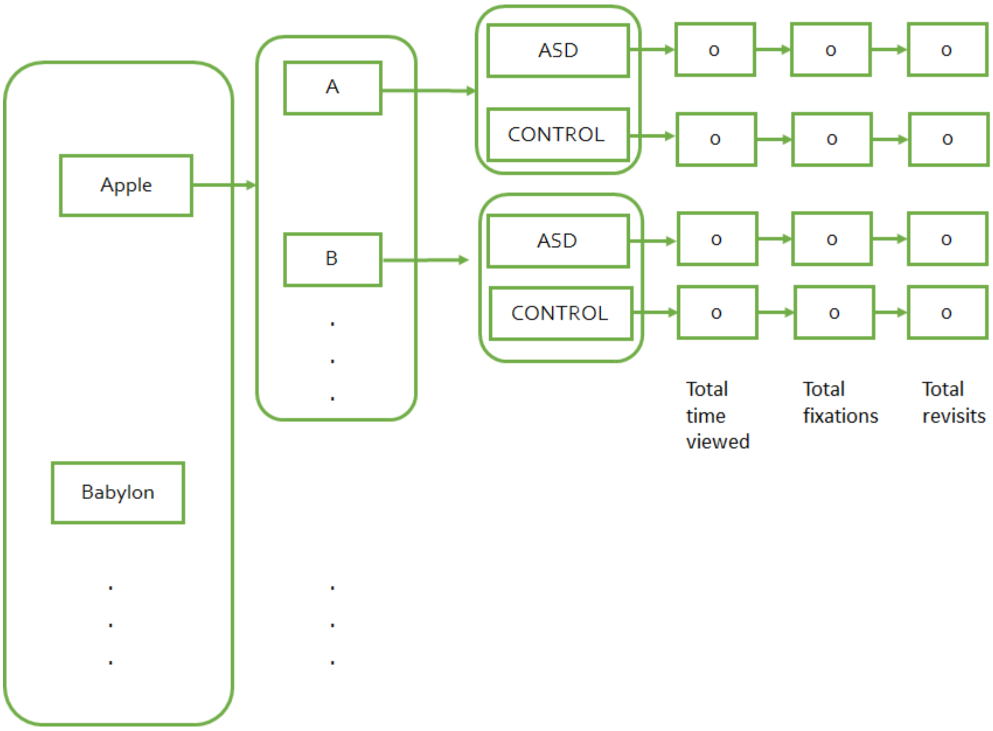


Figure 2: The required structure to keep the total time viewed, the total number of fixations and the total number of revisits for people with autism (ASD) and people without autism (CONTROL) for each element for each web page

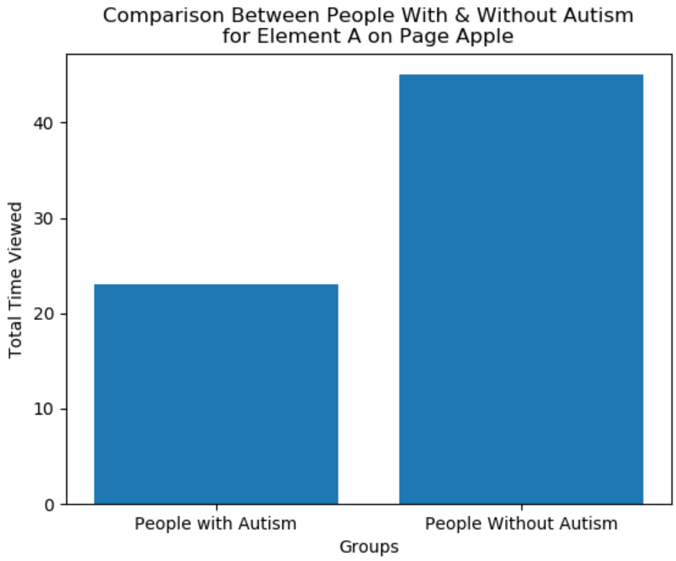
Once the data file is read and the dictionary is created, the application shows the following menu in a loop:

1. Compare the total time viewed, the total number of fixations or the total number of revisits for people with and without autism for a particular element on a specific web page
2. Compare the total time viewed, the total number of fixations or the total number of revisits for people with and without autism on a specific web page
3. Exit

Once a menu item is selected, the application should ask for the required details. Specifically, if the first item is selected, then the metric (the total time viewed, the total number of fixations or the total number of revisits), the element (such as, A) and the page (such as, Apple) should be provided. However, if the second item is selected, then the metric and the page should be provided.

The results should be shown as a bar chart, you can find an **example** piece of code below about how you can create a bar chart in Python.

```
import matplotlib.pyplot as plt
groups = ["People with Autism", "People Without Autism"]
values = [23, 45]
plt.bar(groups, values)
plt.xlabel('Groups')
plt.ylabel('Total Time Viewed')
plt.title('Comparison Between People With & Without Autism\nfor Element A
on Page Apple')
plt.show()
```



Rules

- You need to write your program by using **Python 3.x**.
- You can **only** use all built-in functions and modules.
- You also need to create a file called ReadMe.txt which contains the following items. Please note that **if you do not submit ReadMe.txt, your submission will not be evaluated.**
  - Team members
  - Which version of Python 3.x you have used
  - Which operating system you have used
  - How you have worked as a team, especially how you have divided the tasks among the team members (who was responsible for what?), how you have communicated, how you have tested the program, etc.
- You need to put all your files into a folder which is named with your student id(s) and submit the compressed version of the folder in the **.zip** format.
- **Only one team member** should submit the assignment.
- **Code quality, modularity, efficiency, maintainability and appropriate comments** will be part of the grading.

Grading Policy

The assignment will be graded as follows:

Grading Item	**Mark (out of 100)**
Data Representation	10
Read from a file	10
Create a dictionary	20
Take a command line argument and manage menu operations	10
Compare for a specific element on a particular page with a bar chart	20
Compare for a specific page with a bar chart	20

References:

Sukru Eraslan, Victoria Yaneva, Yeliz Yesilada & Simon Harper (2019) Web users with autism: eye tracking evidence for differences, Behaviour & Information Technology, 38:7, 678-700.