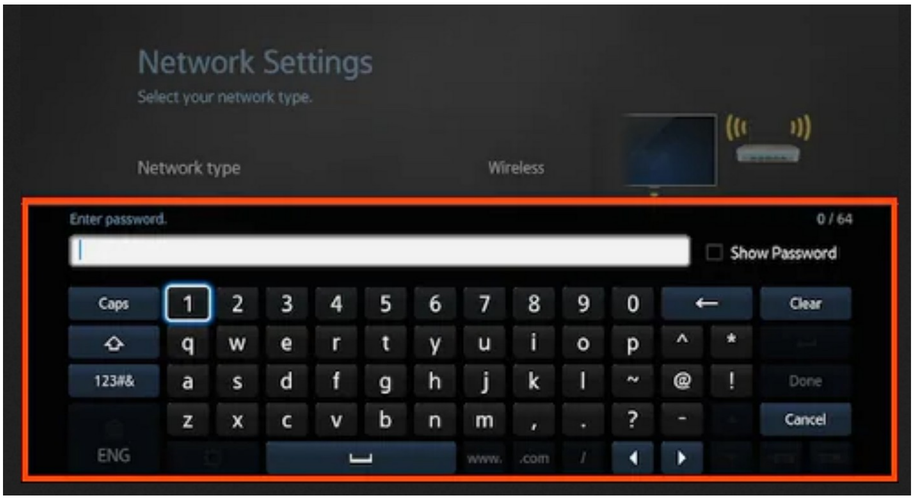
ECON381 Fall 2024

Homework Assignment 3

Passwords are an integral part of our life. There are many rules about choosing you password based on its strength. There are also many tools to suggest strong passwords. However, they usually miss an important point in passwords: Can you actually type it?

This becomes even more important when the devices you use do not have keyboards attached. A very typical example is a smart TV which requires you to use the remote controller's arrow keys to navigate through a virtual keyboard on screen. Here is a sample from a Samsung smart TV.



Network Settings

Select your network type.

Network type Wireless

Enter password. 0 / 64

□ Show Password

Caps 1 2 3 4 5 6 7 8 9 0 ← Clear
⇧ q w e r t y u i o p ^ *
123#& a s d f g h j k l ~ @ ! Done
z x c v b n m , . ? - Cancel
ENG www. .com / ◀ ▶

Figure 1: Trying to type a wifi password on a smart TV

If only a password was **both** strong and easy to type on this setting.

In this assignment, you are required to develop an algorithm to suggest an **8 character** alphanumeric (just the 10 digits and the 26 letters of the English alphabet) password based on the following rules.

1. The user will select the first character to type. You will then suggest the remaining 7.
2. For every character selected, you will **find a short list of characters** that are at a minimum of two but maximum of three moves away from the selected character. For example from the character g, d is 2 moves, s is 3 moves away.
3. You will then **select a random character from this short list**. This will be your next character.
4. Stop selecting characters when you have a password of size 8.

Example.

- Start with **d**, move to **s**. (3 moves)
- Continue with s, move to **e**. (2 moves)
- Continue with e, move to **c** (2 moves).
- Continue with c, move to **b** (2 moves)
- Continue with b, move to **x** (3 moves)
- Continue with x, move to **w** (2 moves)
- Continue with w, move to **4** (3 moves)
- So that the password is **dsecbxw4** which is impossible to guess but relatively fast to type.

You can disregard capital letters (therefore shift and caps lock keys) as well.

Questions

1. Which distance metric is usable for distances between keys?
2. Would you need a particular data structure to represent the keyboard layout? Would this structure be needed permanent or once we calculate the distances between keys, could it be replaced by another structure?
3. Suppose we decided to map each key to a list of valid moves (ie. other keys with 2 to 3 distance). What kind of Java data structure be the best suited for this?
4. Write pseudocode (or Java code) for creating an 8 character password using the data structure you suggested.
5. Compute the list of valid moves for for the following keys: a, f, h, 8, 0, and p.

For Question 4, you need not code with Java, but you might like to do so.

Please prepare a report for these questions and submit as a PDF file through Github.