

## IT351- HCI Laboratory Exercise-1

**Shonali K S**

**181IT244**

1) Prepare a list of 8 to 10 animals. Read this list sequentially and slowly to your friends within a stipulated time. Then ask them to recall the items in the list freely. Record the recall frequencies of these words. Analyze whether the recall % varies with the position of each item within the list? Does the behaviour change for each individual? Does it have something to do with the individual's favourite animal? What is the general trend?

### **Solution:**

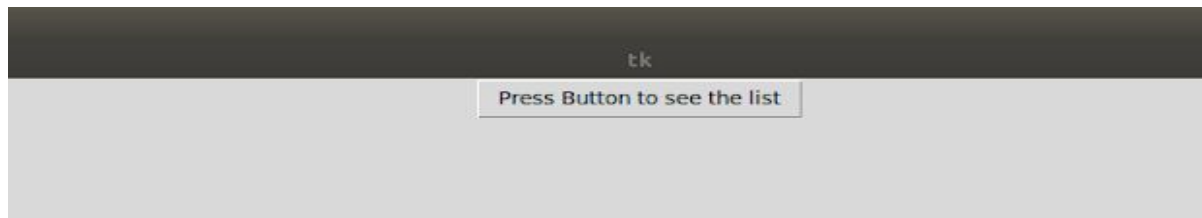
For the following experiment I have chosen 9 animals for equal distribution

- 1) Dog
- 2) Cat
- 3) Cow
- 4) Horse
- 5) Monkey
- 6) Lion
- 7) Tiger
- 8) Bear
- 9) Gorilla

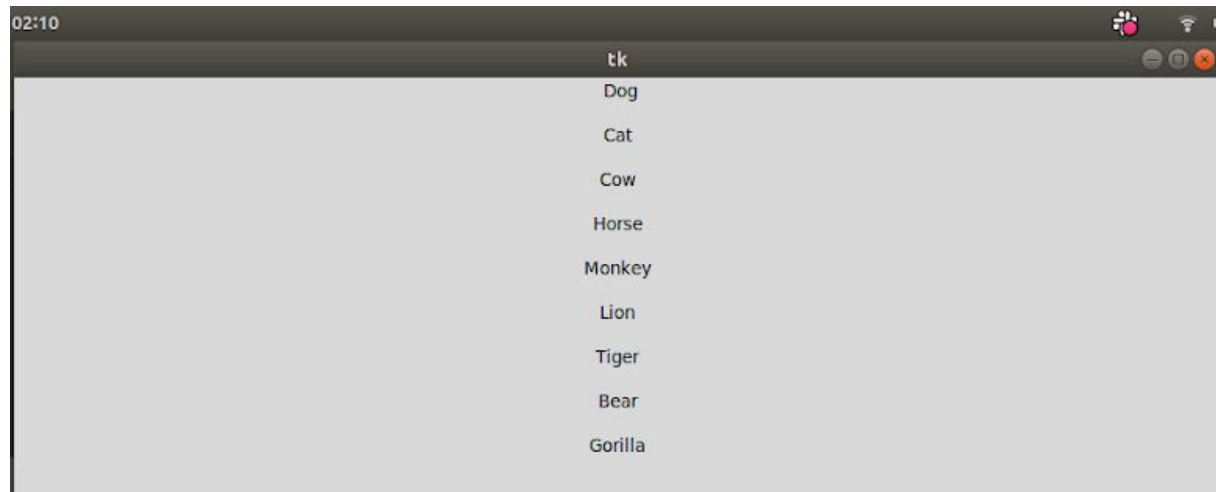
The exercise showed that when users are presented with a list of words, they tend to remember the first few and last few words and are more likely to forget those in the middle of the list.

### **Analysis:**

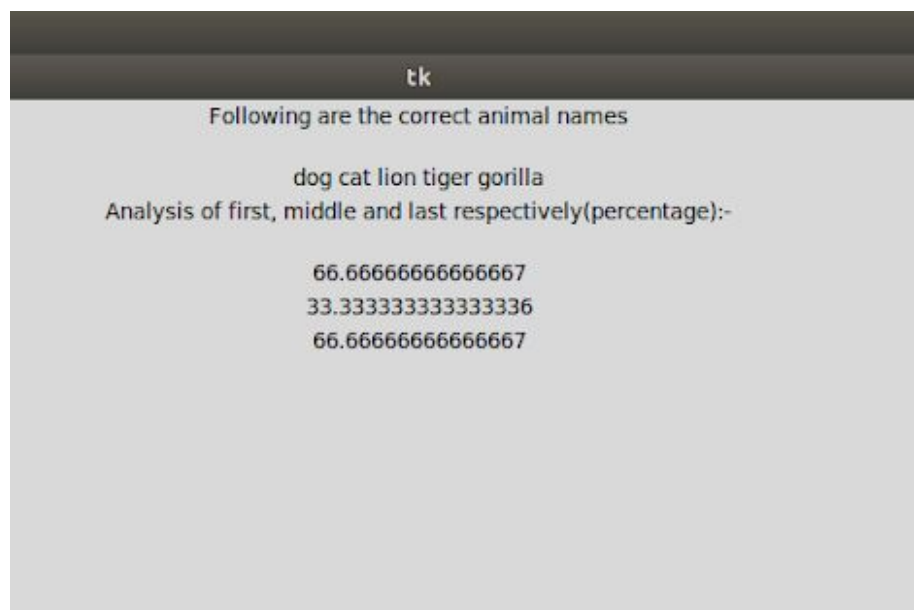
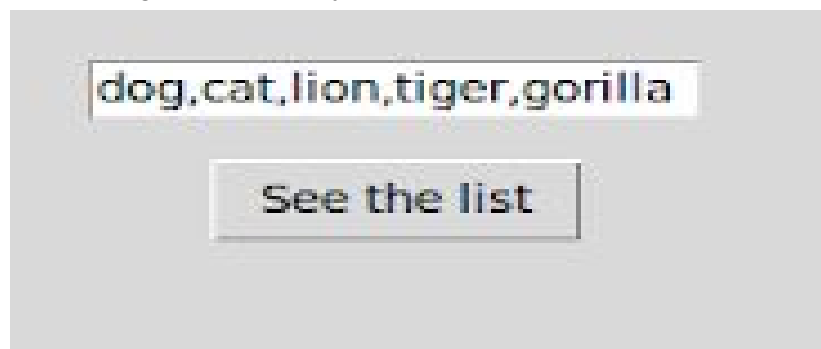
One such example is given below:-



As soon as the user clicks the link he/she will be able to see a list of animals and will have 8 secs to remember it.



After 8 sec the user can write all the animal names he/she remembers and click on the button which will give basic analysis of the user input.



In our case the user could remember **66.67%** of the number of earlier words and recent words and only **33.33%** of middle words.

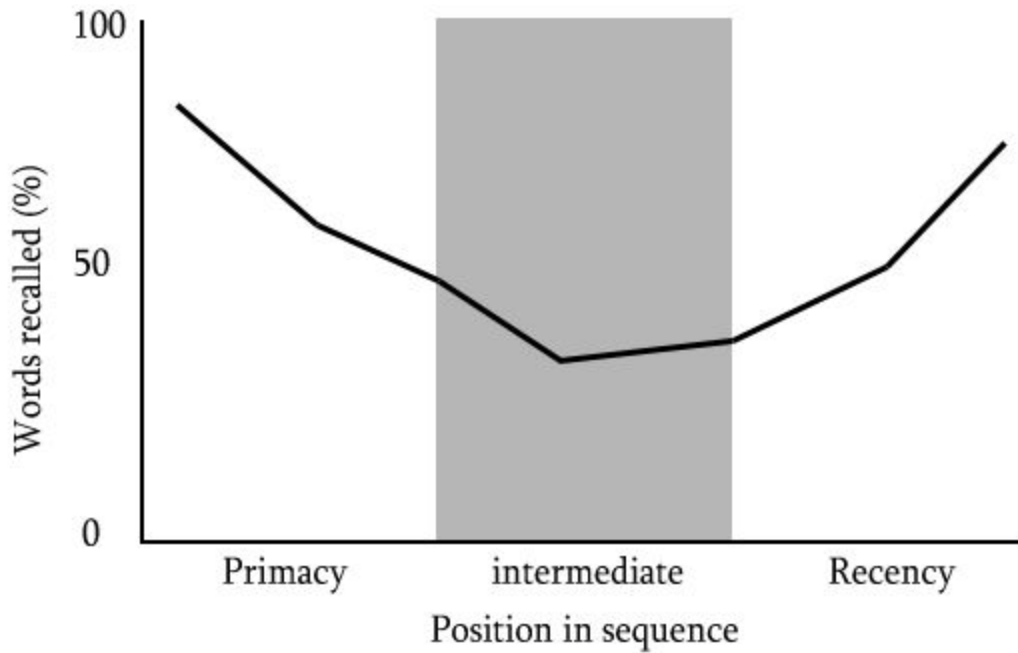
The experiment was done on 10 people, following are the analysis :-

- 1)dog,cat,lion,tiger,giraffe
- 2)lion,tiger,cat
- 3)cat,dog,bear,tiger
- 4)dog,cat,bear,lion,tiger
- 5)gorilla,bear,tiger,lion,dog
- 6)dog,cat,lion,tiger
- 7)cow,tiger,lion,dog
- 8)cat,bear,lion,tiger
- 9)gorilla bear horse dog cat
- 10)tiger lion cat dog horse

Frequency percentage of words:-

- 1) Dog- 80%
- 2) Cat - 80%
- 3) Cow- 10%
- 4) Horse- 20%
- 5) Monkey-0%
- 6) Lion- 80%
- 7) Tiger - 90%
- 8) Bear - 50%
- 9) Gorilla - 20%

The distribution might have a certain trend. People seem to remember dog,cat,and lion tiger as the most common animals we can think of. Plus there is a possibility that since dog and cat were read first ,it's the first thing they will remember. Horse and monkey was very hard for people to recall. Gorilla, a bear which isn't very common for people to think of, was recalled better as it was the most recent animal they read. Lion had quite a remarkable remembrance probably due to favoritism in the group.



This is known as the serial position effect. The tendency to recall earlier words is called the primacy effect; the tendency to recall the later words is called the recency effect. One or two names can vary depending on whether the animal is user's favorite animal or a pet etc. But in general cases the principle of serial position effect remains.

#### CODE:-

```
import tkinter as tk

root = tk.Tk()

animals1=["dog","cat","cow"]

animals2=["horse","monkey","lion"]

animals3=["tiger","bear","gorilla"]

def button_pressed():
```

```

button.destroy()

label = tk.Label(root, text="
Dog\n\nCat\n\nCow\n\nHorse\n\nMonkey\n\nLion\n\nTiger\n\nBear\n\nGorilla\n
\n")

label.pack()

root.after(8000, label.destroy)

root.after(8000, canvas1.pack)

button = tk.Button(root, text="Press Button to see the list",
command=button_pressed)

button.pack()


canvas1 = tk.Canvas(root, width = 400, height = 300)


entry1 = tk.Entry (root)

canvas1.create_window(200, 140, window=entry1)


def getSquareRoot ():

    x1 = entry1.get()

    label = tk.Label(root, text= "Enter all the animals you remember in the
form(small cases)- animal1,animal2,..")

```

```
label.pack()

x=x1.split(",")

first = 0

middle = 0

last = 0

for i in x:

    if i in animals1:

        first=first+1

    elif i in animals2:

        middle=middle+1

    elif i in animals3:

        last=last+1

    else:

        x.remove(i)

label2 = tk.Label(root, text="Following are the correct animal
names\n")

label2.pack()

label1 = tk.Label(root, text=x)

label1.pack()

label3 = tk.Label(root, text="Analysis of first, middle and last
respectively(percentage):-\n ")

label3.pack()
```

```
label4 = tk.Label(root, text=first*100/3)

label4.pack()

label5 = tk.Label(root, text=middle*100/3)

label5.pack()

label6 = tk.Label(root, text=last*100/3)

label6.pack()

canvas1.destroy()

entry1.destroy()

button1.destroy()

button1 = tk.Button(text='See the list', command=getSquareRoot)

canvas1.create_window(200, 180, window=button1)

root.mainloop()
```