

COMP 304 Project 1 Report

Batu Helvacıoğlu – Can Gözpınar

Q1)

We iterated over the paths using `strtok()` with delimiter “:” to find the correct path using `accept()`.

```
can@can-Inspiron-5521: ~/Desktop/comp304Project1
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ cal
      Nisan 2021
Pa Pz Sa Çr Pr Cu Ct
                1  2  3
4  5  6  7  8  9 10
11 12 13 14 15 16 17
18 19 20 21 22 23 24
25 26 27 28 29 30

can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ ls
amk                cronfile.txt      file3.txt          seashell.c        textFile.txt
chdirMem.txt       file1.txt         file4.txt          song.mp3
chimney            file2.tx4.txt     project1.pdf       tempFile.txt
chimneyAnimation.py file2.txt         seashell           textFile
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ date
Paz 04 Nis 2021 14:15:43 +03
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$
```

Q2)

A text file called `chdirMem.txt` store the alias-directory pairs split by \$. We write to the file to add new pairings and read from the file to get the correct path. Once the correct path has been found we use `execvp()` with “cd” and the correct path to change directory.

```
can@can-Inspiron-5521: ~/Desktop/comp304Project1
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ shortdir set home
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ cd ../
can@can-Inspiron-5521:/home/can/Desktop seashell$ shortdir set upDir
can@can-Inspiron-5521:/home/can/Desktop seashell$ cd ../
can@can-Inspiron-5521:/home/can seashell$ shortdir set upUpDir
can@can-Inspiron-5521:/home/can seashell$ shortdir jump home
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ shortdir jump
upUpDir
can@can-Inspiron-5521:/home/can seashell$
```

Listing and deleting saved directories.

```
chdirMem.txt
1 upDir$/home/can/Desktop
2 upUpDir$/home/can
3 home$/home/can/Desktop/comp304Project1
4
```

```
can@can-Inspiron-5521: ~/Desktop/bhelvacioglu18-cgozpina...  
can@can-Inspiron-5521:/home/can/Desktop/bhelvacioglu18-cgozpina18 seashell$ shortdir jump upUpDir  
can@can-Inspiron-5521:/home/can seashell$ shortdir list  
name: home directory: /home/can/Desktop/bhelvacioglu18-cgozpina18  
  
name: upDir directory: /home/can/Desktop  
  
name: upUpDir directory: /home/can  
  
can@can-Inspiron-5521:/home/can seashell$ shortdir del home  
can@can-Inspiron-5521:/home/can seashell$ shortdir list  
name: home directory: /home/can/Desktop/bhelvacioglu18-cgozpina18  
  
name: upDir directory: /home/can/Desktop  
  
name: upUpDir directory: /home/can  
  
name: upDir directory: /home/can/Desktop  
  
name: upUpDir directory: /home/can  
  
can@can-Inspiron-5521:/home/can seashell$
```

Clearing all saved directories.

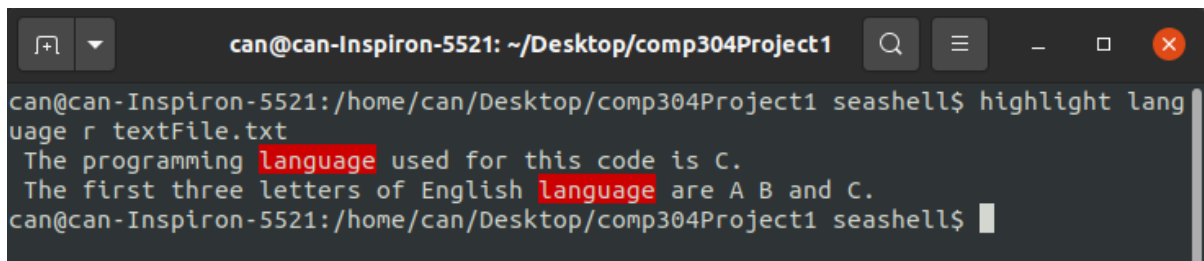
```
can@can-Inspiron-5521: ~/Desktop/comp304Project1  
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ shortdir list  
name: home directory: /home/can/Desktop/comp304Project1  
  
name: upDir directory: /home/can/Desktop  
  
name: upUpDir directory: /home/can  
  
name: upDir directory: /home/can/Desktop  
  
name: upUpDir directory: /home/can  
  
name:  
  directory:  
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ shortdir clear  
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ shortdir list  
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$
```

Alias-directory pairings remain over two sessions.

```
can@can-Inspiron-5521: ~/Desktop/comp304Project1  
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ shortdir list  
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ cd ../  
can@can-Inspiron-5521:/home/can/Desktop seashell$ shortdir set upDir  
can@can-Inspiron-5521:/home/can/Desktop seashell$  
  
can@can-Inspiron-5521: ~/Desktop/comp304Project1  
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ shortdir jump  
upDir  
can@can-Inspiron-5521:/home/can/Desktop seashell$
```

Q3)

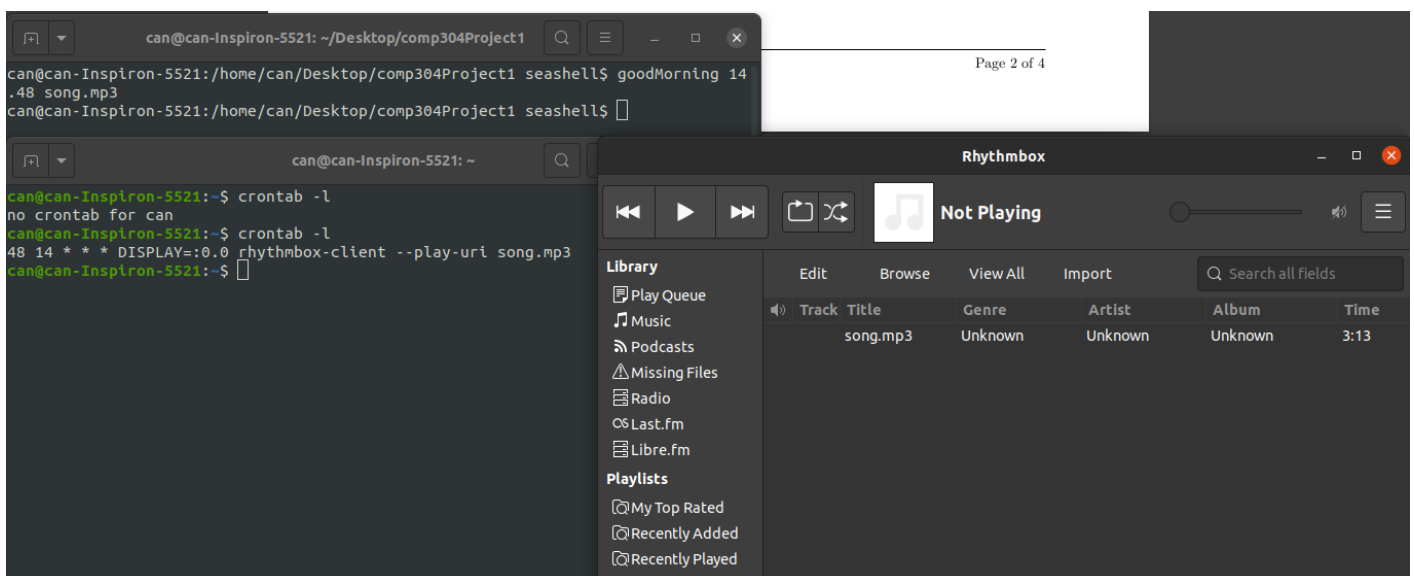
Opens the text file supplied as parameter for reading. On successful opening continues to read the text file line by line. At each line parses the words and checks for the key word to be highlighted. If a match is found then, prints that word in the color {red, green blue} that was supplied as parameter. At the end of each line prints the output and proceeds to do the same operations on the next line until, the text file is fully read.



```
can@can-Inspiron-5521: ~/Desktop/comp304Project1
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ highlight language r textFile.txt
The programming language used for this code is C.
The first three letters of English language are A B and C.
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$
```

Q4)

We take the required arguments from the user and create the appropriate crontab schedule. We use a text file to store the crontab schedule that will be used. Then we pass the text file to crontab using `execvp()`.



The top screenshot shows a terminal window with the following commands and output:

```
can@can-Inspiron-5521: ~/Desktop/comp304Project1
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ goodMorning 14
.48 song.mp3
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$
```

The bottom screenshot shows the Rhythmbox music player interface. The track 'song.mp3' is loaded and is currently 'Not Playing'. The interface includes a library sidebar, a main track display area, and a playlist area.

Track	Title	Genre	Artist	Album	Time
1)	song.mp3	Unknown	Unknown	Unknown	3:13

Q5)

We assume that the files that the user enters are in the current working directory. If argument “-b” is not supplied, we check the file extensions by looping over characters and finding “.” (It is assumed that there will be no “.” in the file name.) to check for text files. We compare the files line by line and print different lines until a file reaches the end. We count the remaining lines in the other file and add them to the different line count. If argument “-b” is supplied, we read the bytes instead of lines.

```
can@can-Inspiron-5521: ~/Desktop/comp304Project1
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ kdiff -a file1.txt file1.txt
The two files are identical
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ kdiff file1.txt file1.txt
The two files are identical
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ kdiff -b file1.txt file1.txt
The two files are identical
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ kdiff -a file2.txt file4.txt
file2.txt : Line: 3 : ali veli 4950
file4.txt : Line: 3 : Veli didn't want to go to school.

3 different lines found
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ kdiff -b file2.txt file4.txt
The two files are different in 68 bytes
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$
```

```
can@can-Inspiron-5521: ~/Desktop/comp304Project1
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ kdiff -a file1.txt file4.txt
file1.txt : Line: 1 : ali was very hungry today.

file4.txt : Line: 1 : veli was very full today.

file1.txt : Line: 2 : he drank a gallon of water today.

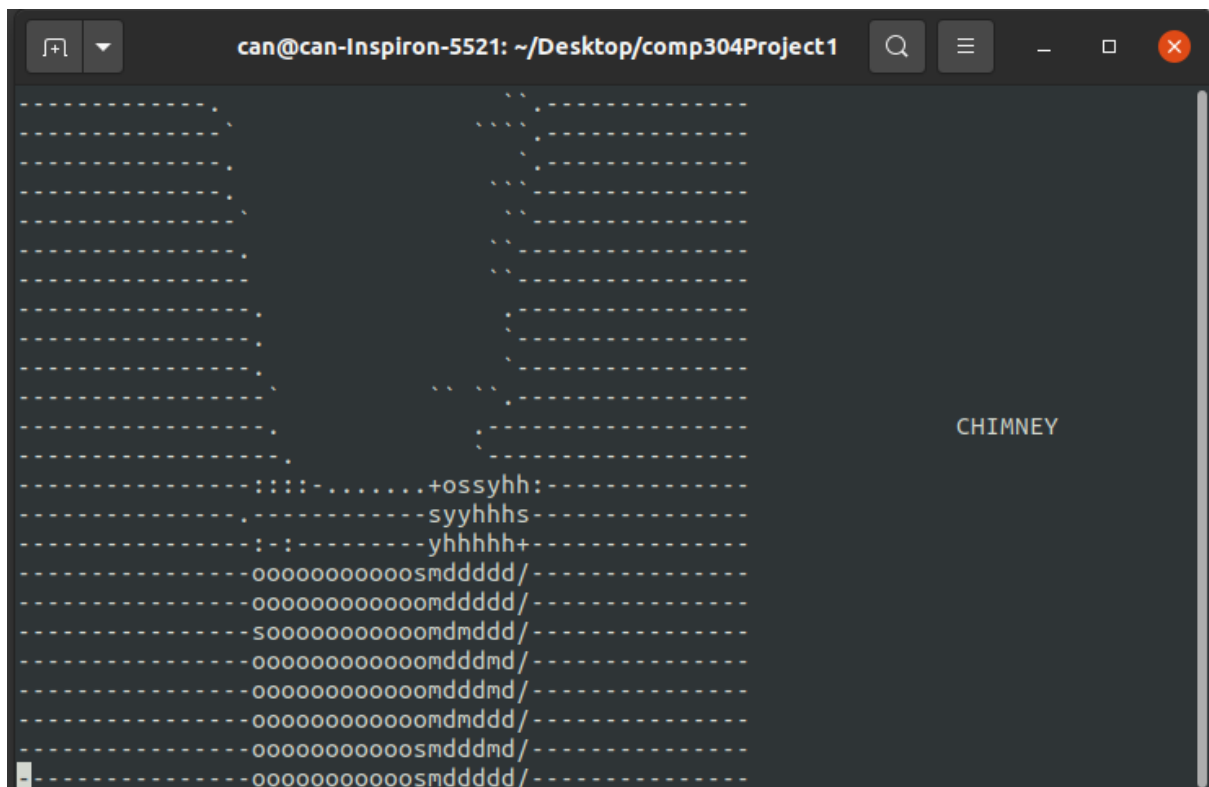
file4.txt : Line: 2 : he devoured a cow yesterday.

file1.txt : Line: 3 : ali veli 4950
file4.txt : Line: 3 : Veli didn't want to go to school.

5 different lines found
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$ kdiff -b file1.txt file4.txt
The two files are different in 73 bytes
can@can-Inspiron-5521:/home/can/Desktop/comp304Project1 seashell$
```

Q6)

We implemented a Baca command. Baca stands for Batu and Can. Baca mean chimney in Turkish. This command displays an animated chimney made out of ASCII characters. We used `execvp` to call a python script that is in the same directory as the `seashell.c`. We used `getcwd()` when `seashell` first initializes to get the directory that the `seashell.c` was in. ASCII frames are saved in `chimney` folder in the same directory as `seashell.c` script. Using a for loop each frame is loaded into a python dictionary. Then, each of these frames are printed on the same line using carriage return character and calling `os` module's `clear` function which prevents scrolling down in the terminal on each successive print operation. One thing to note is that there are 44 frames which means there are 44 text files that each contains one frame of the chimney ASCII animation. file's names are in the form of `frame{i}` in which, `i` is in between 2 and 45 which we use to load these frames using a for loop in the python script. ASCII frames are saved in `chimney` folder in the same directory as `seashell.c` script. Using a for loop each frame is loaded into a python dictionary. Then, each of these frames are printed on the same line using carriage return character and calling `os` module's `clear` function which prevents scrolling down in the terminal on each successive print operation. One thing to note is that there are 44 frames which means there are 44 text files that each contains one frame of the chimney ASCII animation. file's names are in the form of `frame{i}` in which, `i` is in between 2 and 45 which we use to load these frames using a for loop in the python script.



The screenshot shows a terminal window titled "can@can-Inspiron-5521: ~/Desktop/comp304Project1". The terminal displays an ASCII art chimney animation. The chimney is constructed from various characters including dots, dashes, and letters like 'o', 's', 'm', 'd', 'h'. To the right of the chimney, the word "CHIMNEY" is printed in all caps. The animation is shown as a single frame, likely due to the terminal's scrollback buffer or the speed of the animation.

[illegible]