Department of Information Systems and Technologies

CTIS151 – Introduction to Programming

Fall 2024 - 2025

Lab Guide #12 – Week 9 – 1

OBJECTIVES : File Operations and Functions
Instructors : Serpil TIN **Assistants :** Berk ÖNDER, Efe Mert ŞAHİNKOÇ, Hatice Zehra YILMAZ

Q1.a) Write a C program that reads **10** numbers from a text file named **"data.txt"**. The program then displays these values on the screen and calculates their totals.

Project Name: LG12_Q1a
File Name: Q1a.cpp

data.txt

```
12 67 45 18 33 74 56 89 41 27 63 51 75 36 57 4 53 71 58 9 65 42
```

Example Run:

```
Numbers read from the file are: 12 67 45 18 33 74 56 89 41 27
The total of these numbers is: 462
```

b) Modify the program **Q1a.cpp** to read all numbers from the file until the **end of the file** is reached. The program displays the ***odd*** numbers, calculates their average, and also counts the total number of integers in the file.

Finally, a summary will be displayed on the screen, including the total count of integers in the file, the number of odd numbers, and the average of these odd numbers on the screen.

Example Run:

```
 67 is odd!
 45 is odd!
 33 is odd!
 89 is odd!
 41 is odd!
 27 is odd!
 63 is odd!
 51 is odd!
 75 is odd!
 57 is odd!
 53 is odd!
 71 is odd!
  9 is odd!
 65 is odd!
-------------------

The file contains 22 integer numbers.
14 of them are ODD numbers, and average of these odd numbers is: 53.29
```

Project Name: LG12_Q1b
File Name: Q1b.cpp

Q2. Write a C program that reads the grades of students from the file **grades.txt**, and performs the following operations;

- finds the **total number of students**,
- finds the **minimum** of these grades,
- calculates the average of the grades excluding the ones below 40.
- displays the total number of students, minimum grade, and average grade as in the following example run.

grades.txt

```
95 75 64 42 50 14 67 2 12 89
```

Example Run:

```
Number of students is 10
Minimum grade is 2
Number of students with grades >= 40 is: 7
Average of the grades >= 40 is 68.86
```

Project Name: LG12_2
File Name: Q2.cpp

Q3. Write a C program that reads lines from a text file named **"choco.txt".** Each line displays the total number of uppercase, and lowercase letters, and digits on the screen.

Write the following functions;

- **isUpper** that gets a character and returns 1 if it is an uppercase letter, 0 otherwise.
- **isLower** that gets a character and returns 1 if it is a lowercase letter, 0 otherwise.
- **isDigit** that gets a character and returns 1 if it is a digit, 0 otherwise.

Project Name: LG12_Q3
File Name: Q3.cpp

choco.txt

```
Ingredients for classic homemade Chocolate Chip Cookies:
1 cup (2 sticks or 226g) unsalted butter, softened
1 cup Granulated Sugar
2 Large Eggs
1 teaspoon Pure Vanilla extract
2 cups all-purpose Flour
1 teaspoon Baking Soda
1/2 teaspoon salt
2 cups semi-sweet chocolate chips
```

Example Run:

```
1. line contains:
4 uppercase letter(s), 45 lowercase letter(s), and 0 digit(s).

2. line contains:
0 uppercase letter(s), 34 lowercase letter(s), and 5 digit(s).

3. line contains:
2 uppercase letter(s), 16 lowercase letter(s), and 1 digit(s).

4. line contains:
2 uppercase letter(s), 7 lowercase letter(s), and 1 digit(s).

5. line contains:
2 uppercase letter(s), 24 lowercase letter(s), and 1 digit(s).

6. line contains:
1 uppercase letter(s), 18 lowercase letter(s), and 1 digit(s).

7. line contains:
2 uppercase letter(s), 16 lowercase letter(s), and 1 digit(s).

8. line contains:
0 uppercase letter(s), 12 lowercase letter(s), and 2 digit(s).

9. line contains:
0 uppercase letter(s), 27 lowercase letter(s), and 1 digit(s).
```

Q4. Write a C program that reads lines from a file named **"mytext.txt"** and writes the lines in an expander manner to a new file named **"final.txt"**. The idea is to display the characters according to the information next to the character. If the input is "**j1a5w1y9k2c3b1d2**", the program should write "**jaaaaaw999999999kkcccbdd**". Assume that the input only consists of lowercase letters.

Project Name: LG12_Q4
File Name: Q4.cpp

mytext.txt

```
j1a5w1y9k2c3b1d2
```

final.txt

```
j
aaaaa
w
yyyyyyyyy
kk
ccc
b
dd
```

Additional Questions

AQ1.

Write a C program that reads numbers from a file **"nums.txt",** finds the four-digit numbers, and writes them and their separated digits into a new file **"fourdigit.txt"**. The program also checks which of the four digit numbers provides the equation; **abcd = (ab + cd)²** and displays these numbers as in the example run and writes them into a new file **"fourdigit.txt"**, and puts an asterisk (*) next to such numbers in the file **"fourdigit.txt"**.

Write the following functions;

- **isFourDigit** that gets an integer number and checks whether it is a four-digit number.
- **separate** that gets a four-digit integer number and separates the first two digits from the last two digits and returns them separately. For example; if the number is **abcd**, it returns **ab** and **cd**.

Project Name: LG12_AQ1
File Name: AQ1.cpp

Example Run:

```
Numbers providing the equation are
------------------------------
3025
```

nums.txt

```
55
23698
1236
1233
101203
933
8260
874
3025
4728
5564
3
1987
```

fourdigit.txt

```
1236 12 36
1233 12 33
8260 82 60
3025 30 25 *
4728 47 28
5564 55 64
1987 19 87
```

AQ2.

Modify the program **Q4.cpp** so that the program reads lines from a file named **"mytext.txt"** and writes the lines in a compressed manner to a new file named **"compressed.txt"**. The idea is to display only one character and its number of occurrences for repeated characters. If the input is "**jddddyggrclnnfvvvwwwwwkd**", the program should write "**jd4yg2rcln2fv3w5kd**". Assume that the input only consists of lowercase letters.

Project Name: LG12_AQ2
File Name: AQ2.cpp

mytext.txt

```
jddddyggrclnnfvvvwwwwwkd
cccctiiiisbllllll
mmmmmyyiiiipoodiisbbbbbll
lllllooookaaaaatyyyyoooou
```

compressed.txt

```
jd4yg2rcln2fv3w5kd
c4ti4sb2l6
m5y2i4po2di2sb5l2
l5o4ka5ty4o4u
```