MIDDLE EAST TECHNICAL UNIVERSITY, NORTHERN CYPRUS CAMPUS
CNG140 C Programming – Worksheet 22

Purpose:
In this work sheet, you will get hands-on experience with arrays and files.

In this worksheet, you can use string.h library.

Tasks:
a. Assume that the following array contains a week's worth of daily temperature readings

int temperatures [7] = {37, 42, 48, 43, 38, 32, 30};

i) Write a function that takes this array and writes the data to a file called "out.txt".
ii) Write another function that takes this file (out.txt) and reads the data back to this array and computes the average.

A Sample Run would be as follows:

```
out.txt is created!
out.txt is read back again!
Average is 38.6
```

b) Write a program that includes the following steps.
i) Create a file called "payroll.txt"containing the following names, social security numbers, hourly rate and hours worked:

B. Caldwall	555-98-4182	7.32	37
D. Memcheck	555-53-2147	8.32	40
R. Potter	555-32-9826	6.54	40
W. Rosen	555-09-4263	9.80	35

ii) Write a C program that reads this data file and computes and displays a payroll schedule. The output should list the Social Security number, name and gross pay for each individual. Gross pay is computed by multiplying the hourly rate with hours worked.

A Sample Run would be as follows:

```
payroll.txt is read successfully:
Social Security Number:555-98-4182 Name:B.Caldwall Gross Pay:270.84
Social Security Number:555-53-2147 Name:D.Memcheck Gross Pay:332.80
Social Security Number:555-32-9826 Name:R.Potter Gross Pay:261.60
Social Security Number:555-09-4263 Name:W.Rosen Gross Pay:343.00
```

c) Write a program that analyzes an input data file named as "analyze.c". The input file contains only integer values and white spaces. A sample file (sample.txt) is as follows:

```
3 56 45 67
4 5 6 7
2 3
6
```

The program should print the number of lines in the file and the number of integer values. The program and the input file shall be specified on the command line as follows:

```
>analyze sample.txt
```

The output would then be:
```
Your file has 4 lines and 11 integers.
```

d) Write a program in a file called "studentgrades.c" that will process a txt file given on a command line and parses the content of it.

Create a file containing the following student ids and grades of students:

1245646	80
1254668	90
1456248	55
1784563	95
1325468	42

Write a C program that reads the name of the data file from the command line and finds and displays the highest grade and the second highest grade.

A sample run would be as follows:

```
>studentgrades grades.txt

5 students processed!

Highest grade is 95

Second highest grade is 90
```

e) Write a program that analyzes an input data file named as "cars.txt" by following steps;

i) Create a file with a name "cars.txt" containing the car name, space, year, space and price as follows:

```
BMW 2012 35000
AUDI 2015 45000
FORD 2011 18000
BMW 2015 39000
FORD 2012 15000
AUDI 2008 43000
```

ii) Write a program in a file called "analyzeCars.c" that will process this "cars.txt" file given on a command line, parses the content of it and displays the content. Assume that the file willl always have less than 10 cars. Please note that your program will read the file name from the command line so if a different file name is given, for example, "mycars.txt" then your program will process that file.

iii) In your program, create a file name called "FavoriteCars.txt", ask user to choose his/her favorite 2 cars and then add the data of those cars into that file.

A sample run would be as follows:

```
>analyzeCars cars.txt

Available cars are as follows:
```

```
Car 1: BMW 2012 35000
Car 2: AUDI 2015 45000
Car 3: FORD 2011 18000
Car 4: BMW 2015 39000
Car 5: FORD 2012 15000
Car 6: AUDI 2008 43000

Choose your first favorite car: 2
Choose your second favorite car: 4

Your favorite cars are added to FavoriteCars.txt!
```

Please note that based on the above sample run "FavoriteCars.txt" should cantain the following data:

```
AUDI 2015 45000
BMW 2015 39000
```

Recommended Reading: Chapter 10 (pp. 482-536)
Recommended Exercises: Programming Exercises given in the following pages.