**Practical 10**

**Demonstration**

1. Write a C++ program to perform the following tasks:

1. Prompt user to input any 5 integers and write them into the file rawfile.dat. The file will store the integers in a single line where each integer is separated by tab space (“\t”).

E.g.

10 20 30 40 50

**rawfile.dat**

1. Read the 5 integers from rawfile.dat and output them into the scorefile.dat as below (example):

Score Sheet

---------------

Team 1 : 10

Team 2 : 20

Team 3 : 30

Team 4 : 40

Team 5 : 50

-------------------

Total : 150

Average : 30

**scorefile.dat**

Note: Your program should include the necessary file error checking and file closing.

**Exercise**

1. A text file, sales.dat**,** stores the sales data by quarters for the salesmen of PTA Sdn. Bhd. in the following format:

Sample data in **sales.dat**:

Jimmy Lim

100000.00 155000.00 200000.00 125000.00

Peter Chin

132000.00 145000.00 195000.00 165000.00

Use notepad to create the **sales.dat** above with its content. Then write a program to open the file and check for any file opening error, read the data from this file**,** calculate the total sales by quarters and the overall sales total and then display a report on screen as follows:

Salesman Q1 Q2 Q3 Q4

========================================================

Jimmy Lim 100000.00 155000.00 200000.00 125000.00

Peter Chin 132000.00 145000.00 195000.00 165000.00

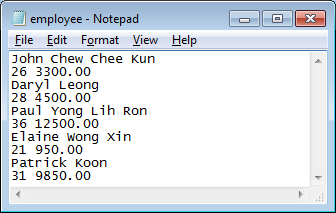
==============================================================

Quarter Total xxxxxx.xx xxxxxx.xx xxxxxx.xx xxxxxx.xx

==============================================================

Grand Total xxxxxx.xx

1. A small company has stored its records of employees in a file called employee.txt as below:



The employee.txt file contains 5 employee records which consist of the *name*, *age* and *salary* of the employee.

Develop a C++ program which fulfills the following requirements:

* Define a structure called **Employee** for storing each employee record.
* Besides declaring the necessary variables, declare a 5-element array variable for the *Employee* structure above. Name the array as employeeList which will store the 5 employee records.

The following diagram illustrates the array:

employeeList

Name

Age

Salary

[ 0 ]

:

:

:

:

:

:

:

:

:

:

:

:

:

:

[ 1 ]

[ 2 ]

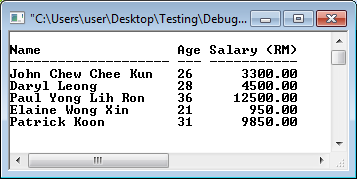
[ 3 ]

[ 4 ]

* Open the employee.txttext file for input. Use the file stream variable

EmpFile.

* Use a loop and appropriate function to read the five (5) records from the *employee.txt* file. Each of the record read will be stored into the employeeList array variable above.
* Close the employee.txt text file.
* Generate the output as per the following screenshot:



You are required to use an appropriate loop structure to extract the five (5) employee records stored in the employeeList array variable and generate the output as the above screenshot.

1. A survey was carried out to find out the acceptance level of a new product produced by a company. The results of the survey were recorded in a text file, **survey.dat**,
2. Prompt user to enter number of responds for each of the 3 acceptance levels as shown below:

E.g.:

Enter number of responds for the acceptance level:

Very Satisfied : xx

Satisfied : xx

Not Satisfied : xx

1. Calculate the percentage of the response for each acceptance level, the total responds and the total percentage of responds
2. Produce and display the following table on screen and save a copy (same format) of this table to the report.dat file

Acceptance Level Number of Responds Percentage

------------------------------------------------------

Very satisfied xx xx.xx

Satisfied xx xx.xx

Not satisfied xx xx.xx

------------------------------------------------------

Total xxx xxx.xx

Note: You should do the necessary file opening error checking and file closing.

1. Sales at Hewlett-Packard are recorded in the text file "sales.txt". Each line of the file specifies a sale with the salesperson's last name and the dollar amount of the sale. The file is sorted alphabetically by name, so that each person's sales appear consecutively. Write a full program (starting from #include) that reads in the file "sales.txt" and prints to the console the total sales amount for each salesperson. An example is shown below. Note the first total is shown as RM21.50, not RM21.5.

|  |  |  |
| --- | --- | --- |
| **Sales.txt** |  | **Console Output** |
| Bakar 11.50  Bakar 10.00  Yuen 4.10  Yuen 3.80  Yuen 1.00  Sharmin 4.88  Jessy 1.00  Jessy 2.00  Jessy 3.00 |  | Bakar earned RM 21.50  Yuen earned RM 8.90  Sharmin earned RM 4.88  Jessy earned RM 6.00 |

1. Write a C++ program that gets the name of a text file from the user. The program then reports the longest and shortest word in the file. In the case when there is a tie for the shortest/longest word, report the word that occurs first. You may assume that punctuation contributes to the length of the word, so that "hi!!!!!!" is a longer word than "hello". Your output should show double quotes " " around the word. Your answer should not require arrays. A sample run is shown below for the given input file **input.txt** at right.

|  |  |
| --- | --- |
| Enter file name: input.txt  The longest word is "elephants!!!".  The shortest word is "out". | **input.txt** |
| Hello Frodo.  Watch out for those elephants!!! |

1. Write a full program, starting from #include, that gets the name of a text file from the user. The program then counts the number of words and lines in the file. You may assume the user's file exists and contains only words, no numbers or isolated punctuation marks. You may also assume that all line breaks occur immediately after a word and there are no blank lines. A sample run is shown below for the input file at right.

|  |  |
| --- | --- |
| Enter file name: input.txt  # words = 6  # lines = 3 | **input.txt** |
| Hello Gandalf.  How are you?  Good? |