

Project 1

Name: Bibek Dhungana

CODE:

```
/*
AUTHOR: Bibek Dhungana
FILENAME: Project1.c
SPECIFICATION: This program takes input of Customer ID and their internet
                used hours as input and calculate the total charge and average
                cost and output these information in the output file.
FOR: CS 1412 Programming Principles 2 Section 504
*/

/*importing all the required library*/
#include <stdio.h>
#include <math.h>

/*Defining macro constant for the flat rate and additional rate*/
#define FLAT_RATE 7.99
#define ADDITINAL_HOURS_RATE 1.99

/*storing the base hour 10 for flat rate in variable*/
const int baseHours = 10;

/*function prototype for total charge*/
void totalCharge(double hoursUsed, double *chargePerHour, double *averageCost);
```

```

/*function prototype for round_money function*/
double round_money(double averageCost);

int main(void){

    /*intializing all the required variables*/
    int month;
    int year;
    int customerId;
    double hoursUsed;
    double charge;
    double averageCost;

    //creating file pointer for reading file
    FILE *inputFilePointer;
    inputFilePointer = fopen("usage.txt","r");

    //creating file pointer for writing in file
    FILE *outputFilePointer;
    outputFilePointer = fopen("charges.txt","w");

    //read hoursUsed and chargePerHour
    fscanf(inputFilePointer,"%d %d", &month, &year);

    //write the Charges for given month on the file
    fprintf(outputFilePointer,"Charges for %d/%d\n\n",month, year);

    //printing the header of the file
    fprintf(outputFilePointer,"Customer      Hours      Charge      Average \n");

    /*Running the while loop reading a data, processing it by
    calling function and writing it into the file*/

    while (fscanf(inputFilePointer, "%d", &customerId) != EOF){
        fscanf(inputFilePointer,"%lf", &hoursUsed);
        totalCharge(hoursUsed, &charge, &averageCost);
        fprintf(outputFilePointer,"%d      %.1f \t %.2f \t %.2f\n",customerId,
hoursUsed, charge, averageCost);
    }
}

```

```

    }

    //closing the pointer to the files.
    fclose(inputFilePointer);
    fclose(outputFilePointer);

    return 0;
}

/*
NAME: totalCharge
INPUT PARAMETERS: hoursUsed(double)
OUTPUT PARAMETRS: charge(double) and averageCost(double)
RETURN TYPE: void
PRECONDITION: charge and averageCost must be paased by reference.
SPECIFICATION: This function takes total hours used as input and address of charg
e and
                average cost. Then it calculate the total charge as per the given
condition.

*/

void totalCharge(double hoursUsed, double *charge, double *averageCost){
    double roundedHoursUsed = ceil(hoursUsed);
    /*checking the condition*/
    if (hoursUsed <= baseHours){
        *charge = FLAT_RATE;

    }
    else{
        *charge = FLAT_RATE + (roundedHoursUsed - baseHours) * ADDITINAL_HOURS_RA
TE;
    }

    /*calculating the average cost*/
    *averageCost = *charge/hoursUsed;

    /*rounding average cost and charge into two decimal place by calling round_mo
ney function*/
    *averageCost = round_money(*averageCost);
    *charge = round_money(*charge);

```

```

}

/*
NAME: round_money
INPUT PARAMETERS: cost (double)
OUTPUT PARAMETERS: NONE
RETURN TYPE: DOUBLE
PRECONDITION: cost must be floating point number.
POSTCONDITION: return value must be rounded to two decimal place.
SPECIFICATION: This function takes total hours used as input and address of charge and
                average cost. Then it calculate the total charge as per the given
                condition.


*/
double round_money(double cost){
    double roundedCost;

    /*creating rounded integer value by multiplying by 100*/
    roundedCost = (int) (cost * 100 + 0.5);

    /*returning the rounded double cost*/
    return (double) (roundedCost/100);
}

```

OUTPUT FILE:

 charges - Notepad

File Edit Format View Help

Charges for 10/2009

Customer	Hours	Charge	Average
15362	4.2	7.99	1.90
42768	11.1	11.97	1.08
11111	9.9	7.99	0.81