

## **Assignment 2**

**ELP - 718 Telecommunication Software Laboratory**

**Varun Gupta**

**2019BSY7505**

**2019-2021**

A report presented for the assignment on  
Basic Programming in C and use of GIT for version control



**Bharti School Of  
Telecommunication Technology and Management**

**IIT Delhi**

**India**

**August 6, 2019**

# Contents

<b>1</b>	<b>Problem Statement-1</b>	<b>4</b>
1.1	Problem Statement . . . . .	4
1.2	Command and Algorithm . . . . .	4
1.3	Screenshot . . . . .	7
<b>2</b>	<b>Problem Statement-2</b>	<b>8</b>
2.1	Problem Statement . . . . .	8
2.2	Command and Algorithm . . . . .	9
2.3	Screenshot . . . . .	10

## List of Figures

1	Screenshot1 . . . . .	7
2	Screenshot1 . . . . .	10

# 1 Problem Statement-1

## 1.1 Problem Statement

Consider a square matrix of size  $N(N \times N \text{ matrix})$ . Each element is of type unsigned integer. Let  $r_i$  denote the minimum value in the  $i$ -th row, and  $c_i$  denote the maximum value in  $i$ -th column. You have to obtain all such  $r_i$ s and  $c_i$ s. From now on, let's call the set of all such  $r_i$ s  $S_1$ , set of all  $c_i$ s  $S_2$  and set containing both  $r_i$ s and  $c_i$ s as  $S$ .  $i = 0$  to  $N-1$

(20 min)

You have to do the following operations:

- 1) From the set  $S$ , find the prime numbers and display those values on the terminal (10 min)
- 2) Consider string  $D$  as a concatenation of all the  $r_i$ s and  $c_i$ s e.g. If  $S$  contains elements 13,10,5,7,10,7, string  $D$  will become 131057107. Now substring will be entered by the user and the program should be able to find the position of the first occurrence of that substring in  $D$ . Display that position along with string  $D$  on the terminal. If the substring is not found then display the message stating no substring matched.

(30 min)

- 3) Let local minimum and local maximum be defined as follows:

4) The element at an index  $i$  is said to be a local minimum if the  $S[i-1] > S[i] < S[i+1]$

5) The element at an Index  $i$  is said to be a local maximum if the  $S[i-1] < S[i] > S[i+1]$

Find the local minima and local maxima and display the index at which they occur in the set  $S$ . If no minima and maxima are found, display the message accordingly.

(30 min)

## 1.2 Command and Algorithm

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
```

```
int main(int argc, char* argv[])
{
    int N,i,j,min,max;
    char d[14][100];
    N= atoi(argv[1]);
    int a[N][N]; int S1[N]; int S2[N]; int S[(2*N)];

    for(i=0;i<N;i++)
```

```

{
    for(j=0;j<N;j++)
    {
        scanf("%d", &(a[i][j]));
    }
}

```

```

for(i=0;i<N;i++)
{
    for(j=0;j<N;j++)
    {
        if(min>a[i][j])
            min = a[i][j];
    }
    S1[i] = min;
    min = 100;
}

```

```

for(j=0;j<N;j++)
{
    for(i=0;i<N;i++)
    {
        if(max < a[i][j])
            max = a[i][j];
    }
    S1[i] = min;
    max = 0;
}

```

```

for (i=0;i<(2*N);i++)
{
    if(i<N)
        S[i] = S1[i];
    if(i>=N)
        S[i] = S2[i-N];
}

```

```

for(i=0;i<(2*N);i++)
{
    for(j=2;j<S[i];j++)

```

```

        {
            if(S[i]%j==0)
            {
                printf("%d is a Prime No.", S[i]);
                break;
            }
        }
    }
for(i=0;i<(2*N);i++)
{
    sprintf(&(d[i][0]), "%d", S[i]);
}
for(i=1;i<(2*N);i++)
{
    strcat(&(d[0][0]), &(d[i][0]));
}

char sub[20];
printf("Enter the substring\n");
scanf("%s", sub);

for(i=0;i<100;i++)
{
    if(d[0][i] == sub[0])
    for(j=0;j<(strlen(sub));j++)
    {
        if(d[0][i+j] != sub[j])
            break;
    }
    if(j == (strlen(sub)))
    {
        printf(" First index for occurence of substring is %d" , j);
        break;
    }
}

printf("Local Minima Indices:");
for(i=1;i<((2*N)-1); i++)
{
    if((S[i]<S[i-1]) && (S[i]<S[i+1]))
        printf(" %d", i);
}

```

```

printf("Local Maxima Indices:");
for(i=1;i<((2*N)-1); i++)
{
    if((S[i]>S[i-1]) && (S[i]>S[i+1]))
        printf(" %d", i);
}
}

```

### 1.3 Screenshot

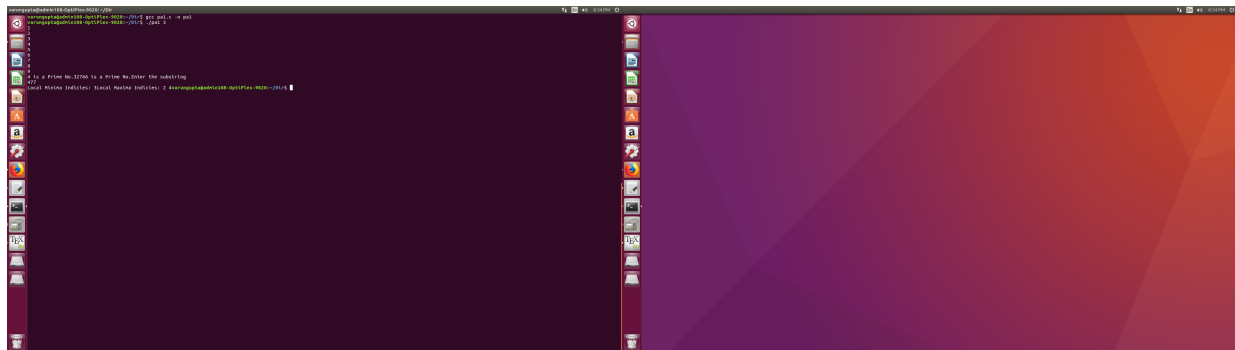


Figure 1: Screenshot1

## 2 Problem Statement-2

### 2.1 Problem Statement

Jeevan, Ramesh, Manisha, Parth and little chhaya went for a trip to Rishikesh. After returning from a trip, they are trying hard to split the expenses equally and settle their debts to each other. Help them splitting the expenses so that everyone gets to know how much money they owe to each other.

Note: Above statement is an example use case. Take your time to understand the whole problem and develop your logic first.

Use Command line argument to take the Number of members involved in the Trip, Lets call it N.

Now enter the names of N people who had gone for the trip.

In the next few lines, start adding the money paid by each of them in the format shown below:

If you want to stop adding expenses, type done and no more expenses can be added. (40 min)

Now the job is to get the calculations done and display the money each member will get back or has to pay. Ramesh pointed out that all the expenses of Chhaya were paid by her father, Jeevan. So, contribution of two shares should be taken for Jeevan and no share for Chhaya, all others will have the share of one each.

You have to make a generalised system for the distribution of expenses. Give the following options to the user and the user distributes expenses by choosing one option:

Split equally: (20 min)

Each member has to contribute equally to the total amount. Use proper calculations and find out how much extra money each member has to give or take back.

Split unequally: (30 min)

Here, user will enter the amount of each members contribution to the total amount and finally display how much remaining money each member has to pay or has to get back. If the sum of the individual amounts is not equal to the total amount (of all expenses), the error message should be displayed and the amounts have to be entered again

Input format:

Enter contributions to total amount in terms of amount that has to be paid by Jeevan Ramesh Manisha Parth Chhaya: xx xx xx xx xx

Split by percentages: (20 min)

Here, user will enter the percentage of each members contribution to the total amount and finally display how much remaining money each member has to pay or has to get back. If



the sum of the percentages does not match to 100, the error message should be displayed and the percentages have to be entered again

Input format:

Enter contributions to total amount in terms of percentage that has to be paid Jeevan Ramesh Manisha Parth Chhaya: 50 10 20 20 0

Split by shares: (10 min)

Here, user will enter the share of each members contribution to the total amount and finally display how much remaining money each member has to pay or has to get back.

Input format:

Enter contributions to total amount in terms of share for Jeevan Ramesh Manisha Parth Chhaya: 2 1 1 1 0

The output of all the above cases should be in the following form: (10 min)

Jeevan has to pay : Rs. XXXX

Ramesh has to pay : Rs. XXXX

Manisha will get back : Rs. XXXX

Parth has to pay : Rs. XXXX

Chhaya has to pay : Rs. XXXX (Here value will be zero in case of 0 share or no share)

Constraints

2 N 7

## 2.2 Command and Algorithm

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>

int main(int argc, char * argv[])
{
    int n; int i;
    n= atoi(argv[1]); /* no of people */
    char names[n][100];
    char expen[100][100];
    for(i=0;i<n;i++)
    {
        scanf("%s", &(names[i][0]));    /* Inputting the names of the members */
    }
    for(i=0;i<100;i++)
    {
```

}

## 2.3 Screenshot

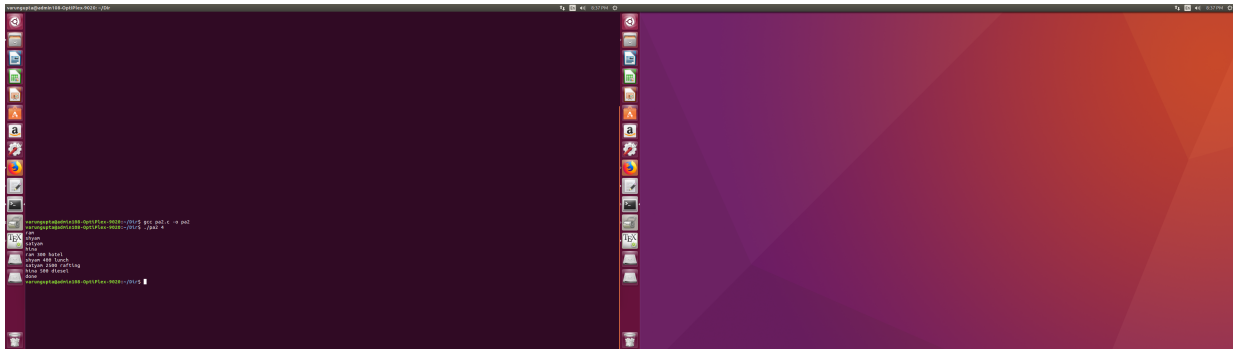


Figure 2: Screenshot1

## References

- [1] <https://www.geeksforgeeks.org/>
- [2] [www.overleaf.com](http://www.overleaf.com)
- [3] <https://stackoverflow.com/>