

|  |
| --- |
| C.P.N.M. LAB REPORT |
|  |
| ASSIGNMENT 7  BCSE FIRST YEAR FIRST SEMESTER  Authored by: SOHAM CHOWDHURY |



**CPNM LAB ASSIGNMENT REPORT**

BCSE FIRST YEAR FIRST SEMESTER 2021-2022

NAME-SOHAM CHOWDHURY

DEPARTMENT-COMPUTER SCIENCE AND ENGINEERING

SECTION-A3.

ROLL NO-002110501145.

# ASSIGNMENT 7

1. Create a structure to specify data on students given below: Roll number, Name, Department, Course, Year of joining, Marks of 6 Subjects Write a function to print the names of all students who joined in a particular year. a. Write a function to print the data of a student whose roll number is given.

b. Write a function to create and store this information in a text file.

c. Write a function which prints in ascending order the rank list (Roll, Name, Department, Average) based on the average of 6 subjects.

First I made a structure which contained details of students such as name roll number and marks in 6 subjects department year of joining and to deal with multiple students I made an array of structures

For printing details of a student whoes roll number has been given I searched for the roll number of the students by a loop fro accesing the array of structures, and for storing the data in a file I used the fprintf function to store the inputs, I just took the avg marks of all stdents sorted them and then printed the merit list.

Program:

#include<stdio.h>

#include<math.h>

#include<stdlib.h>

struct student

{

    int roll\_no;

    char name[30];

    char department[35];

    char course[36];

    int year\_of\_joining;

    int marks[6];

};

void input(struct student p[],int n)

{

    for(int i=0;i<n;i++)

    {

        printf("\nenter the details of the student no %d:-\n",i+1);

        fflush(stdin);

        printf("roll number=");

        scanf("%d",&p[i].roll\_no);

        fflush(stdin);

        printf("name=");

        gets(p[i].name);

        fflush(stdin);

        printf("department=");

        gets(p[i].department);

        fflush(stdin);

        printf("course=");

        gets(p[i].course);

        fflush(stdin);

        printf("year of joining=");

        scanf("%d",&p[i].year\_of\_joining);

        printf("marks in six subjects repectively");

        for(int j=0;j<6;j++)

        {

            scanf("%d",&p[i].marks[j]);

        }

    }

}

void data\_of\_student(int rollnum,struct student s[],int n)

{

    int c=0;

    for(int i=0;i<n;i++)

    {

        if(s[i].roll\_no==rollnum)

        {

            c=1;

            printf("the data of the student with roll number %d is:\n",rollnum);

            printf("roll number=");

            printf("%d",s[i].roll\_no);

            printf("\nname=");

            puts(s[i].name);

            printf("department=");

            puts(s[i].department);

            printf("course=");

            puts(s[i].course);

            printf("year of joining=");

            printf("%d",s[i].year\_of\_joining);

            printf("\nmarks in six subjects repectively:-\n");

            for(int j=0;j<6;j++)

            {

                printf("%d\t",s[i].marks[j]);

            }

        }

    }

    if(c==0)

    printf("\nthe student with this roll number %d could not be found in the database",rollnum);

}

void copy\_data\_to\_file(FILE \* ptr,struct student  s[],int n)

{

    ptr=NULL;

    ptr=fopen("file.txt","a+");

    for(int i=0;i<n;i++)

    {

        fprintf(ptr,"\nthe details of student number %d is:-\n",i+1);

        fprintf(ptr,"%s","roll no=");

        fprintf(ptr,"%d",s[i].roll\_no);

        fprintf(ptr,"%s","\nname=");

        fprintf(ptr,"%s",s[i].name);

        fprintf(ptr,"%s","\ndepartment=");

        fprintf(ptr,"%s",s[i].department);

        fprintf(ptr,"%s","\ncourse=");

        fprintf(ptr,"%s",s[i].course);

        fprintf(ptr,"\nyear of joining=%d",s[i].year\_of\_joining);

        fprintf(ptr,"%s","\nmarks in six subjects respectively:-\t");

        for(int j=0;j<6;j++)

        {

            fprintf(ptr,"\t%d",s[i].marks[j]);

        }

    }

    fclose(ptr);

}

void print\_ranked\_list(struct student s[],int n)

{

    int avg\_marks[30];

    int unsorted\_array[30];

    int temp,c=0,sum1=0,sum2=0;

    for(int i=0;i<n;i++)

    {

        for(int j=0;j<6;j++)

        {

            sum1=sum1+s[i].marks[j];

            sum2=sum2+s[i].marks[j];

        }

        avg\_marks[i]=sum1/6;

        unsorted\_array[i]=sum2/6;

        sum1=0;

        sum2=0;

    }

    for(int i=0;i<n-1;i++)

    {

        for(int i=0;i<n-1;i++)

        {

            if(avg\_marks[i]>avg\_marks[i+1])

            {

                temp=avg\_marks[i+1];

                avg\_marks[i+1]=avg\_marks[i];

                avg\_marks[i]=temp;

            }

        }

    }

    for(int i=0;i<n;i++)

    {

        for(int j=0;j<n;j++)

        {

            if(avg\_marks[i]==unsorted\_array[j])

            {

                printf("\nrank %d, details are as follows:-\n",++c);

                printf("roll number=");

                printf("%d",s[j].roll\_no);

                printf("\nname=");

                puts(s[j].name);

                printf("department=");

                puts(s[j].department);

                printf("course=");

                puts(s[j].course);

                printf("year of joining=");

                printf("%d",s[j].year\_of\_joining);

                printf("\nmarks in six subjects repectively:-\n");

                for(int k=0;k<6;k++)

                {

                    printf("%d\t",s[j].marks[k]);

                }

            }

        }

    }

}

int main()

{

    struct student s[100];

    int n;

    char choice;

    int roll\_num;

    FILE \*ptr=NULL;

    printf("enter the number of students=");

    scanf("%d",&n);

    fflush(stdin);

    input(s,n);

    fflush(stdin);

    printf("enter the choice=");

    scanf("%c",&choice);

    switch(choice)

    {

        case 'a':

        printf("enter the roll number to be searched=");

        scanf("%d",&roll\_num);

        data\_of\_student(roll\_num,s,n);

        break;

        case 'b':

        copy\_data\_to\_file(ptr,s,n);

        break;

        case 'c':

        printf("RANK LIST");

        print\_ranked\_list(s,n);

        break;

        default:

        printf("invalid choice");

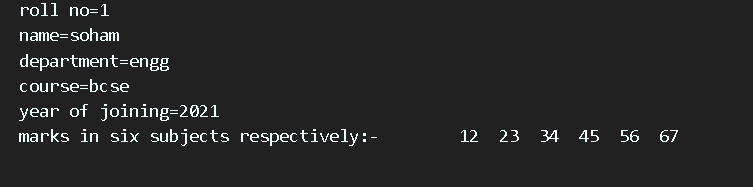
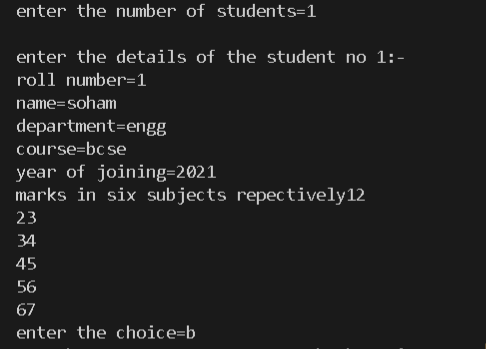
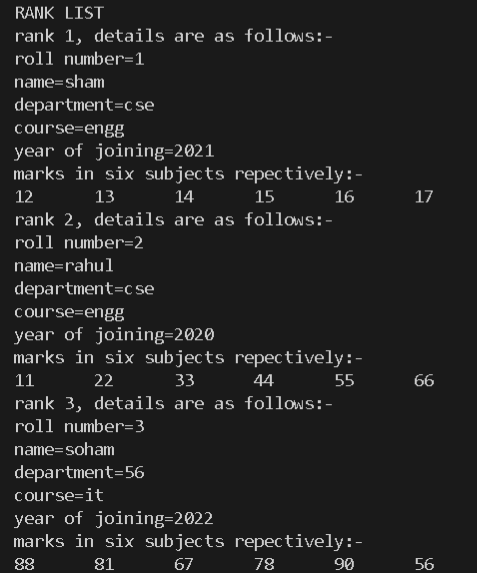
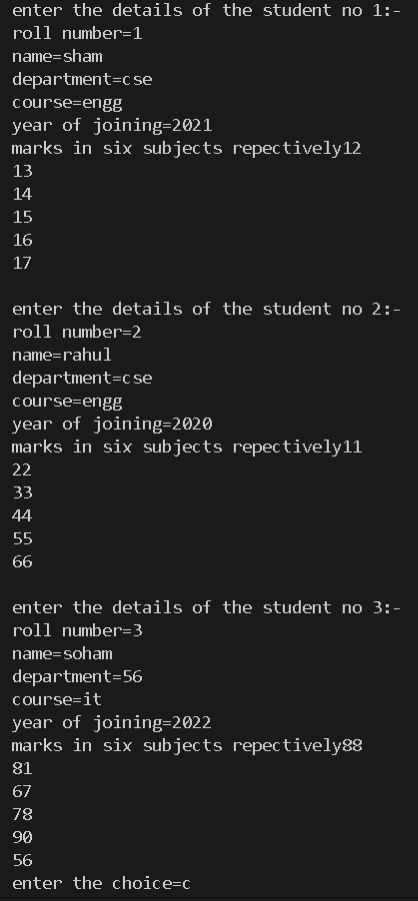
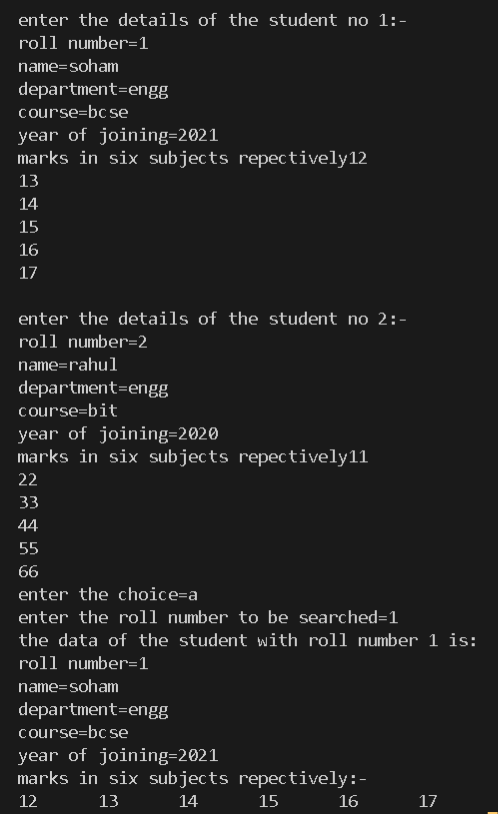
        break;

    }

    return 0;

}

Output:



2. Write a program in C for following tasks

a. Output of the code is the program itself.

b. Read a C source file having comments (between /\* and \*/) and copies it to another file leaving the comments.

c. Reads a C source file and determines the percentage of characters which are part of comments.

d. Reads a C source file and determines no of variables defined of each built in data types.

e. Copy one file to another where files names are passed as command line arguments.

I used fgetc to read the source file and printed these characters till the end of file using loops then terminated comments and stored the data into a text file by traversing the source file and if there were consecutive \ or \* then terminated the part enclose by them. for counting number of variables I got each word by reading the file then if they were having variables such as “int” or ”float” or “char” then incease counter by one also for commas.

Program:

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

int main(int argc,const char \*argv[])

{

    FILE \*ptr1=NULL;/\*file pointers\*/

    FILE \*ptr2=NULL;/\*file pointers\*/

    char str,str2,str3;

    int i=0,k=0,j=0;

    double percent;

    ptr1=fopen(argv[1],"a+");/\*opening the 1st file that is the c source file\*/

    while(!feof(ptr1))

    {

        str=fgetc(ptr1);

        if(str!=' ')

        i++;/\*counting the total number of character except spaces

        in the entire code for percentage calculation\*/

        printf("%c",str);

    }

    FILE \*ptr3=NULL;

    ptr3=fopen(argv[1],"a+");/\*opening the 1st file that is the c source file\*/

    ptr2=fopen(argv[2],"a+");

    while(!feof(ptr3))

    {

        str2=fgetc(ptr3);

        if(str2=='/'&&(fgetc(ptr3)=='\*'))

        {

            while(!feof(ptr3))

            {

                str2=fgetc(ptr3);

                if(str2!=' ')

                k++;

                if(str2=='/')

                break;

            }

        }

        else

        {

            if(str2==EOF)

            {

                break;

            }

            fprintf(ptr2,"%c",str2);

        }

    }

    percent=(double)k/(double)i\*100;

    printf("the percentage of characters in the comments is=%0.3lf",percent);

    printf("\n");

    fclose(ptr2);

    fclose(ptr1);

    FILE \*ptr4=NULL;

    ptr4=fopen(argv[1],"a+");

    char str\_data\_type[100]="";

    unsigned counter=0,var\_counter=0,f=0;

    str3=fgetc(ptr4);

    while(str3!=EOF)

    {

        str\_data\_type[counter]=str3;

        str3=fgetc(ptr4);

        if(str3==' '||str3=='\n')

        {

            if(strcmp(str\_data\_type, "int") == 0 || strcmp(str\_data\_type, "float") == 0 || strcmp(str\_data\_type, "char")==0||strcmp(str\_data\_type,"double")==0||strcmp(str\_data\_type,"unsigned")==0)

            f=1;

            else if(f&&(str3=='\n'))

            f=0;

            int n=strlen(str\_data\_type);

            for(int j=0;j<n+1;j++)

            str\_data\_type[j]='\0';

            counter=-1;

            str3=fgetc(ptr4);

        }

        if(f&&(str3==','||str3==';'))

        var\_counter++;

        counter++;

    }

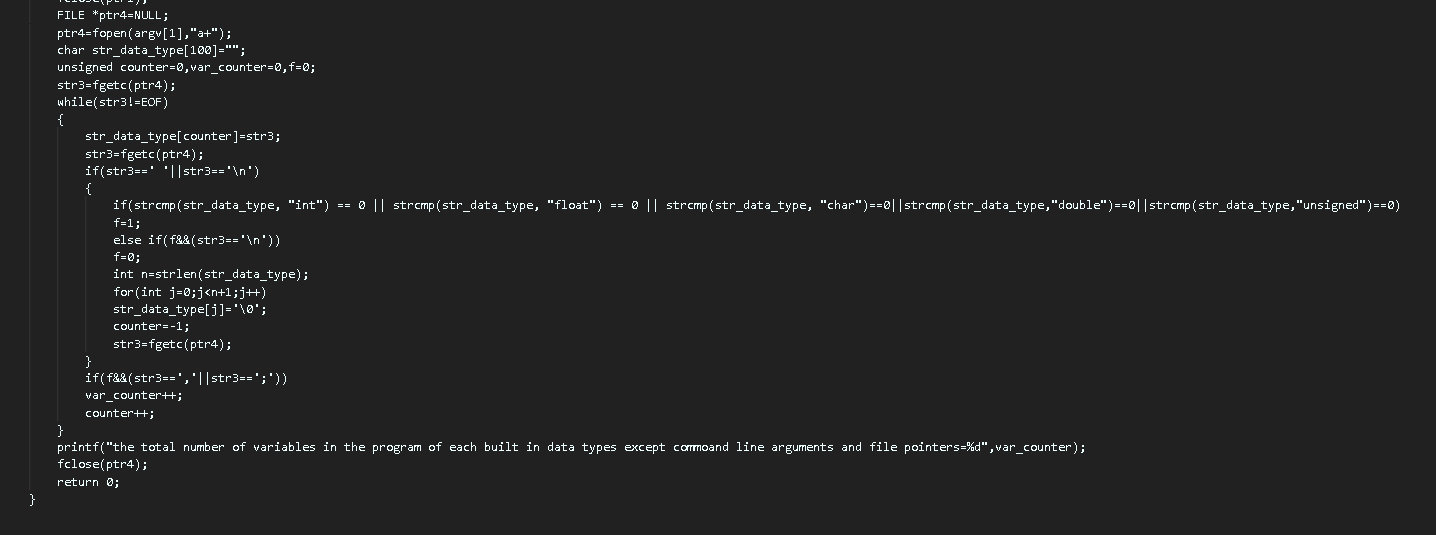
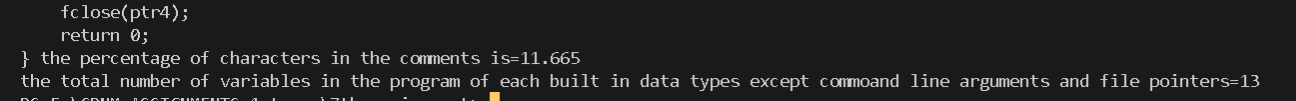
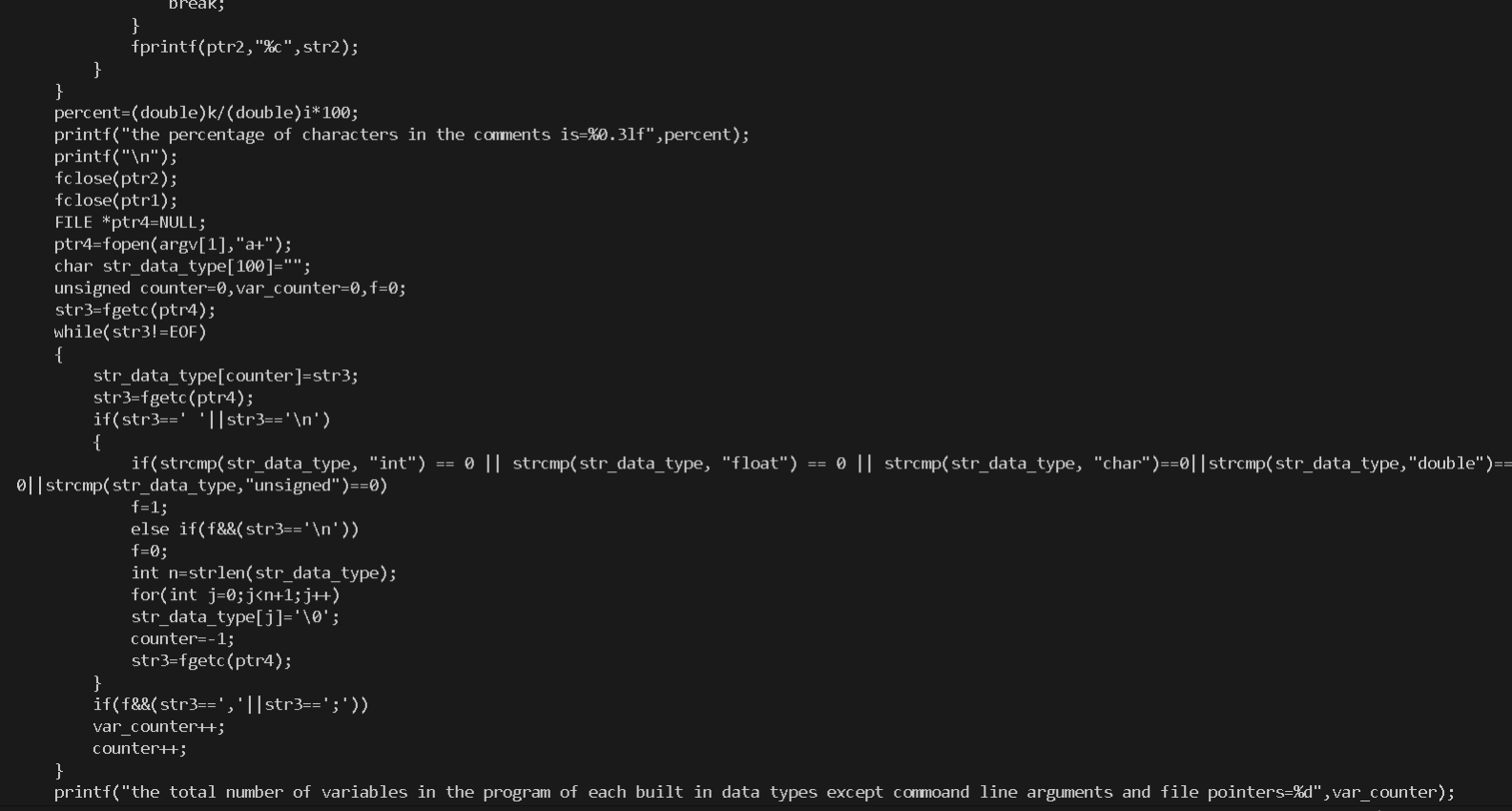
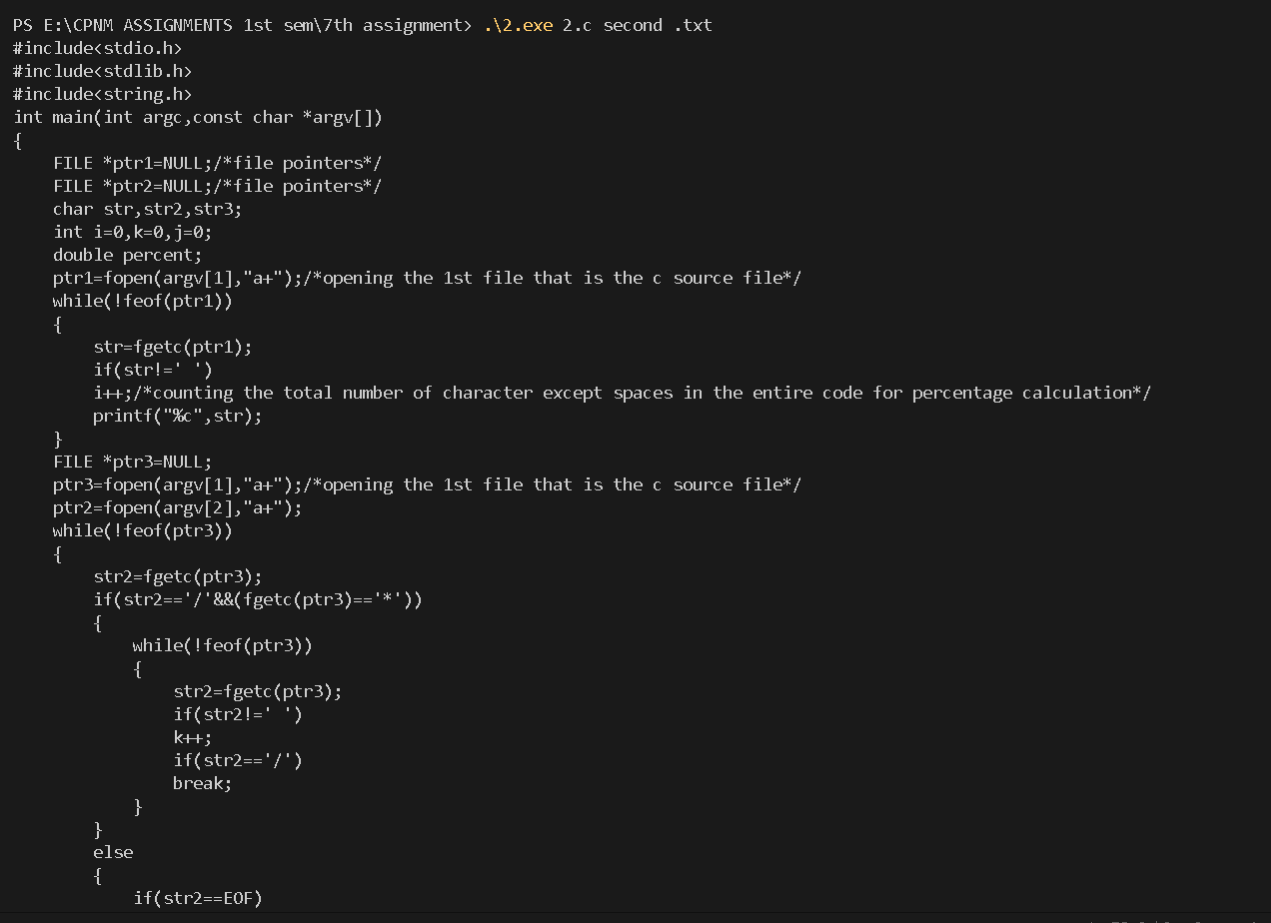
    printf("the total number of variables in the program of each built in data types except commoand line arguments and file pointers=%d",var\_counter);

    fclose(ptr4);

    return 0;

}

Output:



4. Write a program which allows you to do addition and subtraction of two integers which can be upto 40 decimal digits long (call it as huge integers). Create your own representation of huge integers and it should use as minimum space as possible.

Here I just performed digit wise operation by accepting two numbers as string from the user then taking consideration of carry.

Program:

#include<stdio.h>

#include<stdlib.h>

#include<math.h>

#include<string.h>

void sum(char\*,char\*);

void difference(char\*,char\*);

void display(char\*);

void input(int choice)

{

    char str1[40],str2[40];

    printf("enter the 1st number:-");

    gets(str1);

    printf("enter the 2nd number:-");

    gets(str2);

    switch(choice)

    {

        case 1:

        sum(str1,str2);

        break;

        case 2:

        difference(str1,str2);

        break;

        default:

        printf("invalid choice");

    }

}

void sum(char str1[40],char str2[40])

{

    int carry=0,i,num;

    char str3[40];

    char str4[40];

    str1=strrev(str1);

    str2=strrev(str2);

    if(strlen(str1)<strlen(str2))

    {

        strcpy(str4,str2);

        strcpy(str2,str1);

        strcpy(str1,str4);

    }

        for(i=0;i<strlen(str2);i++)

        {

            num=(int)str1[i]+(int)str2[i]-96+carry;

            str3[i]=(char)(num%10+48);

            carry=num/10;

        }

        if(strlen(str1)!=strlen(str2))

        {

            for(i=strlen(str2);i<strlen(str1);i++)

            {

                num=(int)str1[i]-48+carry;

                str3[i]=(char)(num%10+48);

                carry=num/10;

                if((i==strlen(str1)-1))

                {

                    if(num>9)

                    str3[++i]=(char)(carry+48);

                    str3[++i]='\0';

                }

            }

        }

        else

        {

            if(num>9)

            str3[i]=(char)(carry+48);

            str3[++i]='\0';

        }

        display(str3);

}

void difference(char str1[40],char str2[40])

{

    char str3[40];

    char str4[40];

    int carry=0,i,num;

    if(strlen(str2)>strlen(str1))

    {

        strcpy(str4,str2);

        strcpy(str2,str1);

        strcpy(str1,str4);

    }

    if((strcmp(str2,str1)>0)&&(strlen(str1)==strlen(str2)))

    {

        strcpy(str4,str2);

        strcpy(str2,str1);

        strcpy(str1,str4);

    }

    str1=strrev(str1);

    str2=strrev(str2);

    for(i=0;i<strlen(str2);i++)

    {

        num=(int)str1[i]-(int)str2[i]-carry;

        if(num<0)

        {

            carry=1;

            num=num+10;

            str3[i]=(char)(num+48);

        }

        else

        {

            carry=0;

            str3[i]=(char)(num+48);

        }

    }

    if(strlen(str2)!=strlen(str1))

    {

        for(i=strlen(str2);i<strlen(str1);i++)

        {

            num=(int)str1[i]-48-carry;

            if(num<0)

            {

                carry=1;

                num=num+10;

                str3[i]=(char)(num+48);

            }

            else

            {

                carry=0;

                str3[i]=(char)(num+48);

            }

        }

        str3[i]='\0';

    }

    else

    {

        str3[i]='\0';

    }

    display(str3);

}

void display(char str[40])

{

    printf("the resultant is=");

    str=strrev(str);

    puts(str);

}

int main()

{

    int choice;

    printf("1->Sum\n2->difference\n");

    printf("enter choice:-");

    scanf("%d",&choice);

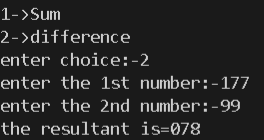
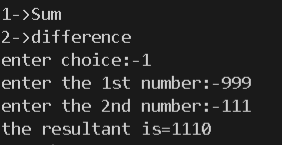
    fflush(stdin);

    input(choice);

    return 0;

}

Output:



5. Write a program to create a Telephone Directory application, which will have options for

a. Add Contact (Name and Telephone No.)

b. Delete Contact

c. Search (By Name)

d. Browse Contact Information should be stored in a text file in a readable format. Allow multiple telephone numbers against a single contact name..

here I used array of structures and inside a structure I kept the name of the phone number bearer the number of numbers he has and his phone numbers in an array where in each index his phone numbers were stored.

Program:

#include<stdio.h>

#include<math.h>

#include<string.h>

struct entries

{

    char name[30];

    int n;

    double ph\_nos[2];

};

void input(struct entries a[],int m)

{

    for(int i=0;i<m;i++)

    {

        printf("enter the name of the number bearer=");

        gets(a[i].name);

        fflush(stdin);

        printf("enter the number of numbers the bearer has=");

        scanf("%d",&a[i].n);

        fflush(stdin);

        for(int j=0;j<a[i].n;j++)

        {

            printf("enter the %dth number=",j+1);

            scanf("%lf",&(a[i].ph\_nos[j]));

            fflush(stdin);

        }

    }

}

int add\_contact(struct entries a[],int m)

{

    printf("enter the details:-\n");

    printf("enter the name of the number bearer=");

    fflush(stdin);

        gets(a[m].name);

        printf("enter the number of numbers the bearer has=");

        scanf("%d",&a[m].n);

        for(int j=0;j<a[m].n;j++)

        {

            printf("enter the %dth number=",j+1);

            scanf("%lf",&(a[m].ph\_nos[j]));

        }

        return ++m;

}

int delete\_contact(struct entries a[],int m)

{

    double temp1,temp2;

    char name\_del[30];

    printf("enter the name of the guy whoes number is to be deleted=");

    fflush(stdin);

    gets(name\_del);

    for(int i=0;i<m;i++)

    {

        if(strcmp(name\_del,a[i].name)==0)

        {

            printf("the contact to be deleted has been found\n");

            for(int j=i;j<m;j++)

            {

                strcpy(a[j].name,a[j+1].name);

                a[j].n=a[j+1].n;

                for(int k=0;k<a[j].n;k++)

                {

                    a[j].ph\_nos[k]=a[j+1].ph\_nos[k];

                }

            }

            m--;

            i--;

        }

    }

    return m;

}

void put\_directory\_to\_file(struct entries a[],int m)

{

    FILE \*ptr=NULL;

    ptr=fopen("telephone.txt","w+");

    for(int i=0;i<m;i++)

    {

        fprintf(ptr,"name=%s",a[i].name);

        fprintf(ptr,"\nnumber of numbers the bearer has=%d",a[i].n);

        fprintf(ptr,"\n%s\n","the phone numbers are:-");

        for(int j=0;j<a[i].n;j++)

        {

            fprintf(ptr,"%0.0lf\n",a[i].ph\_nos[j]);

        }

    }

    fclose(ptr);

}

void search\_contact(struct entries a[],int m)

{

    char name\_search[30];

    printf("enter the name to be searched=");

    fflush(stdin);

    gets(name\_search);

    for(int i=0;i<m;i++)

    {

        if(strcmp(name\_search,a[i].name)==0)

        {

           printf("the name has been found\n");

           printf("the phone numbers are:-\n");

           for(int j=0;j<a[i].n;j++)

           {

               printf("%.0lf\n",a[i].ph\_nos[j]);

           }

        }

    }

}

void browse\_contacts(struct entries a[],int m)

{

    printf("the entries are as follows:-\n");

    for(int i=0;i<m;i++)

    {

        printf("the phone numbers are:-\n");

        printf("name=");

        puts(a[i].name);

        for(int j=0;j<a[i].n;j++)

        {

            printf("%.0lf\n",a[i].ph\_nos[j]);

        }

    }

}

int main()

{

    struct entries a[100];

    int m;

    char c;

    printf("enter the number of entries you want");

    scanf("%d",&m);

    fflush(stdin);

    input(a,m);

    fflush(stdin);

    printf("enter the choice=");

    scanf("%c",&c);

    switch(c)

    {

        case 'a':

        browse\_contacts(a,add\_contact(a,m));

        break;

        case 'b':

        browse\_contacts(a,delete\_contact(a,m));

        break;

        case 'c':

        search\_contact(a,m);

        break;

        case 'd':

        browse\_contacts(a,m);

        break;

        default:

        printf("invalid inputs");

        break;

    }

    printf("the details has been succesfully stored in a file");

    put\_directory\_to\_file(a,m);

    return 0;

}

Output:

