1. Prints the value of a variable and it's memory address

```
#include <stdio.h>
main()
{
   float xval=8.5;
   char usrans='Q';
   printf("\naddress of xval is %p", &xval);
   printf("\nvalue of xval is %f", xval);
   printf("\naddress of usrans is %p", &usrans);
   printf("\nvalue of usrans is %c\n", usrans);
   system("PAUSE");
   return 1;
}
```

2. Checks a given number if prime

```
#include <stdio.h>

int main()
     {
         int r, i, flag=0, n;
         printf("Enter any number to check if prime:\n");
         scanf("%d",&n);
         for(i=2; i<n; i++)
                 {
                 r=n%i;
                 if(r==0)
                   {
                    flag=1;
                   }/*end if*/
                 } /*end of for*/
             if(flag==0)
               {  printf("tne number is prime\n");  }
             else
               {  printf("the number is not prime\n");  }
         return 0;
     }
```

3. Converts a temperature given in cel into far.

```
#include <stdio.h>
int main()
    {
          float f, c;
          printf("Enter the temp in c: \n");
```

```
        scanf("%f", &c);
        f=32 + (9*c)/5;
        printf("The temperature in f=%f/n", f);
       return 0;
    }
```

4. Programe checks, wheather a number is divisible by 2 and 3

```
#include <stdio.h>
int main()
   {
        int num,x,y;
        printf("Enter a number :\n");
        scanf("%d",&num);

        x = num % 2;
        y = num % 3;

        if((x==0) && (y==0))
            printf("%d is divisible by both 2 and 3\n",num);
        else
            printf("%d is not divisible by both 2 and 3\n",num);

        return(0);
   }
```

5. Adds the digits of a given number.

```
#include <stdio.h>

int main()
   {
        int sum=0, r, x,y;
        printf("Enter a number: \n");
        scanf("%d", &x);
        y=x;
        while (x>0)
             {
                  r=x%10;
                  sum=sum+r;
                  x=x/10;
             }
        printf("The sum of the degits of %d is: %d\n", y,sum);
        return 0;
   }
```

6. Find the sum of the given series : 2+5+8+.....

```
#include <stdio.h>

int main()
    {
        int sum=0,x=2,num,i;
        printf("Enter a number to sum upto :\n");
        scanf("%d",&num);

        for(i=0; i<num; i++)
            {
                sum += x;
                x +=3;
            }

        printf("The sum is : %d\n",sum);
        return(0);
   }
```

7. ADD 1+1/2+1/3+1/4.....+1/n (n will be given by the user)

```
#include <stdio.h>
int main()
{
        int num, i, x=2;
        float sum=1.0;
        printf("Enter a number to get the sum upto:\n");
        scanf("%d",&num);
        for(i=1; i<num; i++)
            {
                sum +=1.0/x;
                x++;
           }

        printf("The sum is : %f\n",sum);
        return(0);
}
```

8. Write a c program to convert given number of days to a measure of time given in years, weeks and days. For example 375 days is equal to 1 year 1 week and 3 days (ignore leap year)

```
#include <stdio.h>
#define DAYSINWEEK 7
```

```
void main()
    {
        int ndays, year, week, days;

        printf("Enter the number of days\n");
        scanf("%d",&ndays);

        year = ndays/365;
        week = (ndays % 365)/DAYSINWEEK;
        days = (ndays%365) % DAYSINWEEK;

        printf ("%d is equivalent to %d years, %d weeks and %d days\n",
                    ndays, year, week, days);
    }
```

9. Finds gcd of two given numbers

```
#include <stdio.h>
int main()
    {
        int num1, num2, r, t, gcd;
        printf("Enter the first number to get the GCD :\n");
        scanf("%d",&num1);
        printf("Enter the second number to get the GCD :\n");
        scanf("%d",&num2);

        printf("The GCD of %d and %d is : \n",num1,num2);
        if (num2>num1)
            {
                t=num1;
                num1=num2;
                num2=t;
            }
        while(num1%num2!=0)
            {
                 r=num1%num2;
                 num1=num2;
                 num2=r;
            }
        gcd=num2;
        printf("%d\n",gcd);

        return(1);
   }
```

10. Write a c program to swap the contents of two numbers using bitwise XOR operation. Don't use either the temporary variable or arithmetic operators

```
#include <stdio.h>
void main()
   {
        long i,k;

        printf("Enter two integers\n");
        scanf("%ld %ld",&i,&k);


        printf("\nBefore swapping i= %ld and k = %ld",i,k);

        i = i^k;
        k = i^k;
        i = i^k;

        printf("\nAfter swapping i= %ld and k = %ld",i,k);

   }
```

11. Calculates the factorial of a given number

```
#include <stdio.h>
int main()
   {
        int i,num,x=1;
        long fact=1;
        printf("Enter a number :\n");
        scanf("%d",&num);

        for(i=0; i<num; i++)
          {
          fact *= x;
          x++;
          }

        printf("The factorial of %d is : %ld\n",num,fact);
        return(0);
   }
```

12. Prints n numbers of fibonacci numbers

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

```
int main()
   {
        int num,i;
        long first=0, second=1, new;

        printf("How many fibonacci number to print :\n");
        scanf("%d",&num);
        printf("Fibonacci of %d numbers :\n",num);
        printf("Fibonacci [ 0] is : 0\n");
        printf("Fibonacci [ 1] is : 1\n");

        for(i=1; i<num; i++)
              {
                  new = first + second;
                  printf("Fibonacci [%2d] is : %ld\n",i+1,new);
                  first = second;
                  second = new;
              }
        return(1);
   }
```

13. Finds out the simple ratio of two given numbers.

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

int main()
   {
        int num1, num2, m, n;
        printf("Enter two numbers to get their simple ratio :\n");
        scanf("%d %d",&num1,&num2);
        printf("The two numbers are - %d : %d\n",num1,num2);
        m=num1;
        n=num2;
        while(m!=n)
             {
                 if(m>n)
                      m=m-n;
                 else
                      n=n-m;
              }
        num1 /=m;
        num2 /=m;
        printf("The simple ratio is - %d : %d\n",num1,num2);
        return 0;
```

```
   }
```

14. Converts a decimal number to it's binary equivalent.

```
#include <stdio.h>
int main()
{
        int num,sum=0,bs=1,r,n;
        printf("Enter a number :\n");
        scanf("%d",&num);
        n=num;
        while(num>1)
           {
                r = num%2;
                num /=2;
                sum += r * bs;
                bs *=10;
           }

        sum += num * bs;

        printf("The binary conversion of %d is : %d\n",n,sum);
        return(0);
}
```

15. Write a C program to convert the given binary number into decimal

```
#include <stdio.h>

void main()
    {
         int   num, bnum, dec = 0, base = 1, rem ;

         printf("Enter a binary number(1s and 0s)\n");
         scanf("%d", &num);          /*maximum five digits */

         bnum = num;

         while( num > 0)
         {
                rem = num % 10;
                dec = dec + rem * base;
                num = num / 10 ;
                base = base * 2;
         }
```

```
        printf("The Binary number is = %d\n", bnum);
        printf("Its decimal equivalent is =%d\n", dec);

   }
```

16. Area of the circle without using macro definition

```
#include <stdio.h>
int main()
    {
        float a,r,pi=3.142;
        printf("enter the value of radius r\n");
        scanf("%f",&r);
        a=pi*r*r;
        printf("area of the circle a=%f\n",a);
       return 1;
   }
```

17. Area of the circle using macro definitiom

```
#include<stdio.h>
# define pi 3.142

int main()
   {
        float a,r;
        printf("enter the value of radius of r:\n");
        scanf("%f",&r);
        a=pi*r*r;
        printf("area of the circle a=%f\n",a);
        return 1;
   }
```

18. Write a C program to read an English sentence and replace lowercase characters by uppercase and vice-versa. Output the given sentence as well as the case covrted sentence on two different lines.

```
#include <stdio.h>
#include <ctype.h>
#include <conio.h>

void main()
    {
         char sentence[100];
```

```
        int count, ch, i;

        clrscr();

        printf("Enter a sentence\n");
        for(i=0; (sentence[i] = getchar())!='\n'; i++)
           {
              ;
           }

        sentence[i]='\0';

        count = i; /*shows the number of chars accepted in a sentence*/

        printf("The given sentence is   : %s",sentence);

        printf("\nCase changed sentence is: ");
        for(i=0; i < count; i++)
           {
                ch = islower(sentence[i]) ? toupper(sentence[i]) : tolower(sentence[i]);
                putchar(ch);
           }

   }
```

19. Write a C program read a sentence and count the number of number of vowels and consonants in the given sentence. Output the results on two lines with suitable headings

```
#include <stdio.h>
#include <conio.h>

void main()
    {
        char sentence[80];
        int i, vowels=0, consonants=0, special = 0;

        clrscr();

        printf("Enter a sentence\n");
        gets(sentence);

        for(i=0; sentence[i] != '\0'; i++)
            {
                if((sentence[i] == 'a'||sentence[i] == 'e'||sentence[i] == 'i'||
```

```
                    sentence[i] == 'o'||sentence[i] == 'u') ||(sentence[i] == 'A'||
                    sentence[i] == 'E'||sentence[i] == 'I'|| sentence[i] == 'O'||
                    sentence[i] == 'U'))
                      {
                            vowels = vowels + 1;
                      }
                else
                      {
                            consonants = consonants + 1;
                      }
                if (sentence[i] =='\t' ||sentence[i] =='\0' || sentence[i] ==' ')
                     {
                            special = special + 1;
                     }
          }

       consonants = consonants - special;
       printf("No. of vowels in %s = %d\n", sentence, vowels);
       printf("No. of consonants in %s = %d\n", sentence, consonants);

    }
```

20. Takes input as : INDIA
outputs as :

I
I N
I N D
I N D I
I N D I A

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

int main()
   {
      char ip[20];
      int i,j,l;
      printf("Enter a word : \n");
      gets(ip);
      l=strlen(ip);

      for (i=0; i<l; i++)
            {
               for(j=0; j<=i; j++)
```

```
                    {
                       printf("%c ", ip[j]);

                    }
                printf("\n");
             }
        return(1);
    }
```

21. Sums the elements of an array.

```
#include <stdio.h>

int main()
     {
          int a[10], i, sum=0;
          printf("Enter 10 elements:\n");
          for(i=0; i<10; i++)
                {
                     scanf("%d", &a[i]);
                     sum=sum+a[i];
                }
          printf("sum of the array=%d\n", sum);
          return 0;
    }
```

22. Prints the reverse of a given string

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

int main()
    {
          char name[15], rev[15];
          int n,i=0, l;

          printf("Enter a word:\n");
          gets(name);
          n=strlen(name);
          printf("The reverse of the word :");
          for(i=n-1; i>=0; i--)
                {
                     printf("%c",name[i]);
                }
          printf("\n");
```

```
        return(1);
    }
```

23. Program for concatanation, copy and length of string

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

int main()
    {
        char fname[15], lname[15], name[30];
        int n, x;
        printf("Enter ur first name :\n");
        gets(fname);
        printf("Enter ur last name :\n");
        gets(lname);
        strcpy(name,fname);
        printf("Your name : %s\n",name);
        strcat(name,lname);
        printf("Your full name : %s\n",name);
        n=strlen(name);
        printf("The length of ur name : %d\n",n);
        x = strncmp(fname,lname,1);
        if(x==0)
            printf("The first character of ur first and last name is same.\n");
        else
            printf("The first character of ur first and last name is not same.\n");

        return(1);
    }
```

24. Takes the values of an array and prints the max and next max values

```
#include <stdio.h>
int main()
    {
        int nums[10];
        int max1, max2, n, i, x=1;
        printf("How many numbers (less than 10) u want to compare :\n");
        scanf("%d",&n);
        for(i=0; i<n; i++)
            {
                printf("number %d :\n",x);
                x++;
```

```
            scanf("%d",&nums[i]);
        }

    printf("The numbers are:\n");
    for(i=0; i<n; i++)
        {
            printf("%d ",nums[i]);
        }
    printf("\n");

    x=0;
    max1=nums[0];
    for(i=1; i<n; i++)
       {
            if(max1<nums[i])
                {
                    max2=max1;
                    max1=nums[i];
                }
       }
    printf("The maximum values are: %d %d\n",max1, max2);
    return(1);
  }
```

25. To find Max and Min in an array

```
#include<stdio.h>
main()
    {
        float a[20],max,min;
        int i;
        for(i=0;i<20;i++)
           {
                printf("Enter value:\n");
                scanf("%f",&a[i]);
           }
        max=min=a[0];
        for(i=1;i<20;i++)
          {
                if(max<a[i])
                max=a[i];
                if(min>a[i])
                min=a[i];
           }
        printf("In the given array max=%f & min=%f\n",max,min);
    }
```

26. For input: National Institute Science Tetcnology , output: N. I. S. T.

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

int main()
   {
        char name[25];
        int n, i=0, x;
        printf("Enter your name : \n");
        gets(name);
        n=strlen(name);

        if(name[0]!='\0')
                x=toupper(name[0]);
        printf("%c",x);
        for(i=1; i<n; i++)
           {
                if(name[i]==' ')
                     {
                          x=toupper(name[i+1]);
                          printf(". %c",x);
                     }
           }
        printf("\n");

        return(0);
   }
```

27. Checks a string if polidrome and and print the string in reverse.

```
#include <stdio.h>
#include <string.h>
int main()
   {
        char name[15];
        int n,i=0, flag;
        printf("Enter a word:\n");
        gets(name);
        n=strlen(name);
        while(i<=n/2)
            {
                if(name[i]==name[n-1])
```

```
                {
                    flag=1;
                    i++;
                    n--;
                }
            else
                {
                    flag=0;
                    break;
                }
        }
    if (flag==1)
        printf("This word is a polidrome\n");
    else    printf("This word is not a polidrome\n");

    printf("The reverse of the word :");
    n=strlen(name);
    for(i=n-1; i>=0; i--)
        {
            printf("%c",name[i]);
        }
    printf("\n");

    return(1);
}
```

28. Write a C Program to accept a grade and declare the equivalent descrption *
if code is S, then print SUPER
if code is A, then print VERY GOOD
if code is B, then print FAIR
if code is Y, then print ABSENT
if code is F, then print FAILS
***/**

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

void main()
    {
        char remark[15];
        char grade;

        printf("Enter the grade\n");
        scanf("%c",&grade);

        grade=toupper(grade);  /* lower case letter to upper case */
```

```
            switch(grade)
                {
                    case 'S': strcpy(remark," SUPER");
                                        break;

                    case 'A': strcpy(remark," VERY GOOD");
                                        break;

                    case 'B': strcpy(remark," FAIR");
                                        break;

                    case 'Y': strcpy(remark," ABSENT");
                                        break;

                    case 'F': strcpy(remark," FAILS");
                                        break;

                    default : strcpy(remark, "ERROR IN GRADE\n");
                                        break;

                }               /* End of switch */

            printf("RESULT  : %s\n",remark);

        }
```

29. Write a C program to accept a string and a substring and check if the substring is present in the given string

```
#include<stdio.h>

void main()
    {
        char str[80],search[10];
        int count1=0,count2=0,i,j,flag;
        puts("Enter a string:");
        gets(str);

        puts("Enter search substring:");
        gets(search);

        while (str[count1]!='\0')
                count1++;
```

```
        while (search[count2]!='\0')
                count2++;

        for(i=0;i<=count1-count2;i++)
            {
                for(j=i;j<i+count2;j++)
                    {
                        flag=1;
                        if (str[j]!=search[j-i])
                            {
                                flag=0;
                                break;
                            }
                    }
                if (flag==1)  break;
            }
        if (flag==1)
                puts("SEARCH SUCCESSFUL!");
        else
                puts("SEARCH UNSUCCESSFUL!");
        getch();
}
```

30. Adds the digits of a number using a user defined function.

```
#include <stdio.h>

int sumdigits(int x);

int main()
    {
        int sum, x;
        printf("Enter a number: \n");
        scanf("%d", &x);
        sum=sumdigits(x);
        printf("The sum of the degits of %d is: %d\n",x,sum);
        return 0;
    }

int sumdigits(int x)
     {
        int r, sum=0;
        while (x>0)
            {
                r=x%10;
```

```
                sum=sum+r;
                x=x/10;
            }  /* end of while*/
        return sum;
    }   /* end of fun */
```

**31. Swaps the value of two variables with function **

```
#include <stdio.h>
void swap(int *x,int *y);

int main()
    {
        int x, y;
        printf("Enter two values of x and y:\n");
        scanf("%d%d",&x,&y);
        printf("Before swapping X=%d and Y=%d\n", x,y);
        swap(&x,&y);
        printf("After swapping X=%d and Y=%d\n", x,y);
        return 0;
    }

void swap(int *x, int *y)
    {
        int t=*x;
        *x=*y;
        *y=t;
    }
```

32. Write a C program to compute the value of X ^ N given X and N as inputs using recursive function

```
#include <stdio.h>
#include <math.h>

long int power(int x, int n);
void main()
    {
        long int x,n,xpown;
        long int power(int x, int n);

        printf("Enter the values of X and N\n");
        scanf("%ld %ld",&x,&n);

        xpown = power (x,n);
```

```
            printf("X to the power N = %ld\n");
      }

 /*Recursive function to computer the X to power N*/

 long int power(int x, int n)
      {
            if (n==1)
                 return(x);
            else if ( n%2 == 0)
                 return (pow(power(x,n/2),2));  /*if n is even*/
            else
                 return (x*power(x, n-1));     /* if n is odd*/
      }
```

33. Calculates the GCD of two numbers with function

```
#include <stdio.h>
int gcd(int x, int y);

int main()
    {
          int x, y, g;
          printf("Enter two numbers:\n");
          scanf("%d %d", &x, &y);
          g=gcd(x, y);
          printf("Ultimate value of GCD= %d\n", g);
          return 0;
    }

int gcd(int x, int y)
    {
          int min, max, r;
          if(x>y)
               {
                    max=x;
                    min=y;
               }
          else
               {
                    max=y;
                    min=x;
               }
         r=max%min;
         while(r!=0)
```

```
            {
                max=min;
                min=r;
                r=max%min;
             }   /* end of while */
        return min;
    } /* end of function */
```

34. Function to concatenate two string

```
#include <stdio.h>

void str_concat(char *source, char *destination);

int main()
    {
        char firststr[15], secondstr[15];
        int i=0;
        printf("enter one string:\n");
        scanf("%s", firststr);
        printf("enter the second string:\n");
        scanf("%s", secondstr);
        str_concat(firststr, secondstr);
        printf("the new string is:\n%s\n", firststr);
        return 0;
    }

void str_concat(char *source, char *destination)
    {
        int s=0, d=0;
        while(source[s]!='\0')
            {
                s++;
            } /* end of while loop */
        while(destination[d]!='\0')
            {
                 source[s]=destination[d];
                s++;
                d++;
            }  /* end of  2nd while */
        source[s]='\0';

   } /* end of function */
```

35. Function to copy string

```
#include <stdio.h>
void str_copy(char *source, char *destination);
int leg_str(char *source);

int main()
   {
        char firststr[15], secondstr[15];
        int i=0;
        printf("enter one string:\n");
        scanf("%s", firststr);
        str_copy(firststr, secondstr);
        printf("the copied string is:\n%s\n", secondstr);
        i=leg_str(firststr);
        printf("the length of the string is :%d\n", i);
        return 0;
  }

void str_copy(char *source, char *destination)
    {
          int i=0;
          while(source[i]!='\0')
                {
                     destination[i]= source[i];
                     i++;
                } /* end of while loop */
          destination[i]='\0';

    }  /* end of function */

int leg_str(char *source)
      {
            int c=0;
            while(source[c]!='\0')
                {
                   c++;
                } /* end of while */
            return(c);
      } /*end of function */
```

36. Userdefined function to find a string's length.

```
#include <stdio.h>
int strlength(char a[]);
```

```
int main()
    {
          char a[10];
          int l;
          printf("enter a string:\n");
          scanf("%s", a);
          l=strlength(a);
          printf("length of the string=%d", l);
          system("PAUSE");
          return 0;
    }

int strlength(char a[])
   {
         int l=0;
         while (a[l]!='\0')
             {
                l++;
             } /* end of while loop */
          return l;
   } /* end of function */
```

37. Calculates the factorial of a number by using recursive function call.

```
#include <stdio.h>
long fact(long x);
int main()
    {
        long factorial;
        long num;
        printf("Enter a number :\n");
        scanf("%ld",&num);
        factorial = fact(num);
        printf("The factorial of %ld is : %ld\n",num,factorial);
        return(0);
   }

long fact(long x)
  {
        if(x==0)
             return (1);
        else
             return(x * fact(x-1));
  }
```

38. Write a C program to accept an array of 10 elements and swap 3rd

element with 4th element using pointers. And display the results

```
#include <stdio.h>

void main()
     {

          float x[10];
          int i,n;
          void swap34(float *ptr1, float  *ptr2 ); /* Function Declaration */

          printf("How many Elements...\n");

          scanf("%d", &n);
          printf("Enter Elements one by one\n");
          for(i=0;i<n;i++)
               {
                    scanf("%f",x+i);
               }

          swap34(x+2, x+3);      /* Function call:Interchanging 3rd element by 4th */

          printf("\nResultant Array...\n");
          for(i=0;i<n;i++)
             {
                    printf("X[%d] = %f\n",i,x[i]);
             }
   }
```

39. Write a C program to find the sum of all elements of an array using pointers as arguments

```
#include <stdio.h>

void main()
    {
          static int array[5]={ 200,400,600,800,1000 };
          int sum;
          int addnum(int *ptr);     /* function prototype */

          sum = addnum(array);

          printf("Sum of all array elements = %5d\n", sum);

    }     /* End of main() */
```

```
int addnum(int *ptr)
   {
        int index, total=0;

        for(index = 0; index < 5; index++)
              {
                   total += *(ptr+index);
              }
        return(total);
   }
```

40. Dynamic allocation of array and getting the average of that array.

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

int main()
     {
        float *a,sum=0.0,avg;
        int i,l;
        printf("How many real numbers u want to put in to array:\n");
        scanf("%d",&l);
        a=(float*)malloc(l*sizeof(float));
        for(i=0;i<l;i++)
           {
              printf("Enter no %d value:\n",i+1);
              scanf("%f",&a[i]);
              sum+=a[i];
              avg=sum/(float)l;
           }
        printf("The avarage of the elements in array is : %f\n",avg);
      return 1;
    }
```

41. Write a C program to read N names, store them in the form of an array and sort them in alphabetical order. Output the give names and the sorted names in two columns side by side with suitable heading

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

void main()
     {
          char name[10][8], Tname[10][8], temp[8];
```

```
        int i, j, N;

        clrscr();

        printf("Enter the value of N\n");
        scanf("%d", &N);

        printf("Enter %d names\n", N);
        for(i=0; i< N ; i++)
            {
                scanf("%s",name[i]);
                strcpy (Tname[i], name[i]);
            }
        for(i=0; i < N-1 ; i++)
           {
                for(j=i+1; j< N; j++)
                    {
                        if(strcmpi(name[i],name[j]) > 0)
                            {
                                strcpy(temp,name[i]);
                                strcpy(name[i],name[j]);
                                strcpy(name[j],temp);
                            }
                    }
            }

        printf("\n----------------------------------------\n");
        printf("Input Names\tSorted names\n");
        printf("----------------------------------------\n");
        for(i=0; i< N ; i++)
            {
                printf("%s\t\t%s\n",Tname[i], name[i]);
            }
        printf("----------------------------------------\n");

    }
```

42. Prints the sum of each row and column of a matrix

```
#include<stdio.h>
main()
    {
        int i,j,rsum=0,csum=0,d1sum=0,d2sum=0,vals[4][4];
        for(i=0;i<4;i++)
            {
                for(j=0;j<4;j++)
```

```
                {
                    printf("\n vals[%d][%d]:",i,j);
                    scanf("%d",&vals[i][j]);
                }
            }
        for(i=0;i<4;i++)
           {
            for(j=0;j<4;j++)
                    printf("\t vals[%d][%d]%d",i,j,vals[i][j]);
            printf("\n");
           }
        for(i=0;i<4;i++)
          {
            rsum=0;
            csum=0;
            d1sum=0;
            d2sum=0;
            for(j=0;j<4;j++)
                {
                  rsum+=vals[i][j];
                  if(i==j)
                     d1sum+=vals[i][j];
                  if((i+j)==3)
                     d2sum+=vals[i][j];
                  csum+=vals[j][i];
                }
           printf("row sum for row %d is %d",i+1,rsum);
           printf("\n");
           printf("col sum for row %d is %d",i+1,csum);
           printf("\n");
          }
    }
```

43. Takes the elements from the user and puts into a matrix

```
#include <stdio.h>
#define m 3
#define n 3

int main()
   {
        int a[m][n];
        int i,j;
        printf("enter %d elements:\n", m*n);
        for(i=0; i<m; i++)
             {
```

```
            for(j=0; j<n; j++)
                {
                    scanf("%d", &a[i][j]);
                }
        }
    for(i=0; i<m; i++)
        {
            for(j=0; j<n; j++)
                {
                    printf("%d ", a[i][j]);
                }
            printf("\n");
        }

    system("PAUSE");
    return 0;
}
```

44. Addition of two matrices.

```
#include <stdio.h>
#define m 3
#define n 3
int main()
    {
        int a[m][n], b[m][n], s[m][n];
        int i,j;
        printf("enter %d elements for the first matrix:\n", m*n);
        for(i=0; i<m; i++)
            {
                for(j=0; j<n; j++)
                    {
                        scanf("%d", &a[i][j]);
                    }
            }
        printf("enter %d elements for the second matrix:\n", m*n);
        for(i=0; i<m; i++)
            {
                for(j=0; j<n; j++)
                    {
                        scanf("%d", &b[i][j]);
                    }
            }
        printf("The resultant matrix is:\n");
        for(i=0; i<m; i++)
            {
```

```
            for(j=0; j<n; j++)
                {
                    s[i][j]=a[i][j]+b[i][j];
                    printf("%d ", s[i][j]);
                }
            printf("\n");
        }
    return 0;
}
```

45. Multiplication of two matrices

```
#include <stdio.h>
#define m 3

int main()
    {
        int a[m][m], b[m][m], mul[m][m];
        int i,j, k;
        printf("enter %d elements for the first matrix:\n", m*m);
        for(i=0; i<m; i++)
            {
                for(j=0; j<m; j++)
                    {
                        scanf("%d", &a[i][j]);
                    }
            }
        printf("enter %d elements for the second matrix:\n", m*m);
        for(i=0; i<m; i++)
            {
                for(j=0; j<m; j++)
                    {
                        scanf("%d", &b[i][j]);
                    }
            }

        for(i=0; i<m; i++)
            {
                for(j=0; j<m; j++)
                   {
                        mul[i][j]=0;
                        for(k=0; k<m; k++)
                            {
                                mul[i][j]=mul[i][j]+a[i][k]*b[k][j];
                            }
                   }
```

```
        }
    printf("The resultant matrix is:\n");
    for(i=0; i<m; i++)
        {
            for(j=0; j<m; j++)
              {
                  printf("%d ", mul[i][j]);
              }
            printf("\n");
        }
    return 0;
}
```

46. Write a C program to read A (MxN), find the transpose of a given matrix and output both the input matrix and the transposed matrix.

```
#include <stdio.h>

void main()
    {
        int i,j,M,N;
        int A[10][10], B[10][10];
        int transpose(int A[][10], int r, int c);  /*Function prototype*/

        printf("Enter the order of matrix A\n");
        scanf("%d %d", &M, &N);

        printf("Enter the elements of matrix\n");
        for(i=0;i<M;i++)
            {
                for(j=0;j<N;j++)
                    {
                        scanf("%d",&A[i][j]);
                    }
            }

        printf("Matrix A is\n");
        for(i=0;i<M;i++)
            {
                for(j=0;j<N;j++)
                    {
                        printf("%3d",A[i][j]);
                    }
                printf("\n");
            }
```

```
        /* Finding Transpose of matrix*/
        for(i=0;i<M;i++)
            {
                for(j=0;j<N;j++)
                    {
                        B[i][j] =  A[j][i];
                    }
            }

        printf("Its Transpose is\n");
        for(i=0;i<M;i++)
            {
                for(j=0;j<N;j++)
                    {
                        printf("%3d",B[i][j]);
                    }
                printf("\n");
            }

  }
```

47. Write a C Program to check if a given matrix is an identity matrix

```
#include <stdio.h>

void main()
   {
        int A[10][10];
        int i, j, R, C, flag =1;

        printf("Enter the order of the matrix A\n");
        scanf("%d %d", &R, &C);

        printf("Enter the elements of matrix A\n");
        for(i=0; i<R; i++)
            {
                for(j=0; j<C; j++)
                    {
                        scanf("%d",&A[i][j]);
                    }
            }
        printf("MATRIX A is\n");
        for(i=0; i<R; i++)
            {
                for(j=0; j<C; j++)
```

```
                    {
                        printf("%3d",A[i][j]);
                    }
                printf("\n");
            }

        /* Check for unit (or identity) matrix */

            for(i=0; i<R; i++)
                {
                    for(j=0; j<C; j++)
                        {
                            if(A[i][j] != 1 && A[j][i] !=0)
                                {
                                    flag = 0;
                                    break;
                                }
                        }
                }

            if(flag == 1 )
                printf("It is identity matrix\n");
            else
                printf("It is not a identity matrix\n");

    }
```

48. Writes the Rs amount in words.

```
#include <stdio.h>

void fi(int x);
void f2(int x);
void f3(int x);
void f4(int x);
void f5(int x);
void f6(int x);

static char *a1[]={"zero", "One", "Two", "Three", "Four", "Five",
                "Six", "Seven", "Eight", "Nine", "Ten", "Eleven",
                "Twelve", "Thirteen", "Furteen", "Fifteen", "Sixteen",
                "Seventeen", "Eighteen", "Ninteen"};

static char *a2[]={"Zero", "Ten", "Twenty", "Thirty", "Fourty", "Fifty",
                "Sixty", "Seventy", "Eighty", "Ninty"};
int i;
```

```
int main()
   {
        double num, num1;
        long rs;
        printf("Enter the amount in Rs :\n");
        scanf("%lf", &num);
        rs=num;
        printf("Rupees ");
        if(rs<20)  f1(rs);
        else if((rs>19) && (rs<100))  f2(rs);
        else if((rs>99) && (rs<1000))  f3(rs);
        else if((rs>999) && (rs<100000))  f4(rs);
        else if((rs>99999)&&(rs<10000000))  f5(rs);
        else if((rs>9999999)&&(rs<1000000000))  f6(rs);

        num1=num-rs;
        if(num1!=0)
            {
                printf("and ");
                num1 *=100;
                rs=num1;
                if (rs>19) f2(rs);
                else f1(rs);
                printf("Paise ");
            }

        printf("Only....\n");
        return 0;
   }

void f1(int x)
     {
          printf("%s ", a1[x]);
      }

void f2(int x)
     {
          i=x/10;
          printf("%s ",a2[i]);
          i=x%10;
          if(i!=0)  printf("%s ", a1[i]);
     }

void f3(int x)
    {
```

```
            i=x/100;
            printf("%s Hundred ", a1[i]);
            i=x%100;
            if((i!=0)&&(i<20))     f1(i);
           else if((i!=0)&&(i>19))   f2(i);
    }

void f4(int x)
    {
          i=x/1000;
          if(i<20)   f1(i);
         else   f2(i);
         printf("Thousand ");
         i=x%1000;
         if((i!=0)&&(i<20))   f1(i);
         else if ((i!=0)&&(i<100))   f2(i);
         else if (i!=0)   f3(i);
   }

void f5(int x)
   {
         i=x/100000;
         if(i>19)   f2(i);
         else   f1(i);
         printf("Lakh ");
         i=x%100000;
         if((i!=0)&&(i<20))   f1(i);
         else if((i!=0)&&(i<100))   f2(i);
         else if((i!=0)&&(i<1000))   f3(i);
         else if((i!=0)&&(i<100000))  f4(i);
   }

void f6(int x)
   {
         i=x/10000000;
         if(i>19) f2(i);
         else f1(i);
         printf("Crore ");
         i=x%10000000;
         if((i!=0)&&(i<20))   f1(i);
         else if((i!=0)&&(i<100))  f2(i);
         else if((i!=0)&&(i<1000))   f3(i);
         else if((i!=0)&&(i<100000))   f4(i);
         else if((i!=0)&&(i<10000000))   f5(i);
   }
```

49. Write a C program to read N integers and store them in an array A, and so find the sum of all these elements using pointer. Output the given array and and the computed sum with suitable heading

```
#include <stdio.h>
#include <malloc.h>

void main()
    {
          int i,n,sum=0;
          int *a;

            printf("Enter the size of array A\n");
          scanf("%d", &n);

          a=(int *) malloc(n*sizeof(int));   /*Dynamix Memory Allocation */

          printf("Enter Elements of First List\n");
          for(i=0;i<n;i++)
               {
                    scanf("%d",a+i);
               }

          /*Compute the sum of all elements in the given array*/
          for(i=0;i<n;i++)
              {
                    sum = sum + *(a+i);
              }

          printf("Sum of all elements in array = %d\n", sum);

  }     /* End of main() */
```

50. Stores the employee dbase using structure, from the user and prints it

```
#include <stdio.h>
#include <string.h>
typedef struct
      {
           char fname[20];
           char lname[15];
           long emp_code;
      } EMP;
```

```
int main()
    {
        EMP nist[20];
        int i,n;
        printf("Enter the number of employees :\n");
        scanf("%d",&n);
        for(i=0; i<n; i++)
           {
                   printf("Enter the first name :\n");
                   scanf("%s",nist[i].fname);
                   printf("Enter the last name :\n");
                   scanf("%s",nist[i].lname);
                   printf("Enter the employee code :\n");
                   scanf("%ld",&nist[i].emp_code);
           }

        for(i=0; i<n; i++)
           {
                   printf("The name of the employee : %s %s and the employee code
                           :%ld\n", nist[i].fname,nist[i].lname,nist[i].emp_code);
           }

        return(0);
    }
```

51. Write a C program to illustrate as to how the data stored in the file is read

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

void main()
   {
        FILE *fptr;
        char filename[15];
        char ch;

        printf("Enter the filename to be opened\n");
        gets(filename);

        fptr = fopen (filename, "r");  /*open for reading*/

        if (fptr == NULL)
            {
                printf("Cannot open file\n");
                exit(0);
            }
```

```
        ch = fgetc(fptr);

        while (ch != EOF)
            {
                printf ("%c", ch);
                ch = fgetc(fptr);
            }

        fclose(fptr);
    }
```

52. Write a C program to create a file called emp.rec and store information about a person, in terms of his name, age and salary.

```
#include <stdio.h>

void main()
    {
        FILE *fptr;
        char name[20];
        int age;
        float salary;

        fptr = fopen ("emp.rec", "w"); /*open for writing*/

        if (fptr == NULL)
             {
                  printf("File does not exists\n");
                 return;
              }
        printf("Enter the name\n");
        scanf("%s", name);
        fprintf(fptr, "Name    = %s\n", name);

        printf("Enter the age\n");
        scanf("%d", &age);
        fprintf(fptr, "Age     = %d\n", age);

        printf("Enter the salary\n");
        scanf("%f", &salary);
        fprintf(fptr, "Salary  = %.2f\n", salary);

       fclose(fptr);
    }
```