1)Exercise 1: Write a c program to convert English units to metric (i.e., miles to kilometers, gallons to liters, etc.). Include a specification and a code design.

```
#include<stdio.h>

int main()
{
   int choice;
   printf("Select the type of conversion\n");
   printf("1. Miles to Kilometers\n");
   printf("2. Gallons to Liters\n");
   printf("3. Pounds to Kilograms\n");
   printf("4. Inches to Centimeters\n");
   printf("Enter your choice (1-4): ");
   scanf("%d",&choice);
   float temp;
   switch(choice)
   {
      case 1:

         printf("Enter distance in miles: ");
         scanf("%f",&temp);
         printf("Distance in km is %f",temp*1.60934);
         break;

      case 2:

         printf("Enter volume in gallons: ");
```

```
            scanf("%f",&temp);
            printf("Distance in km is %f",temp*3.78541);
            break;

        case 3:

            printf("Enter weight in pounds: ");
            scanf("%f",&temp);
            printf("Distance in km is %f",temp*0.453592);
            break;

        case 4:

            printf("Enter length in inches: ");
            scanf("%f",&temp);
            printf("Distance in km is %f",temp*2.54);
            break;

        default:
            printf("enter a valid choice");
            break;
    }

    return 0;
}
```

2) Exercise 2: Write a program to perform date arithmetic such as how many days there are between 6/6/90 and 4/3/92. Include a specification and a code design.

```c
#include <stdio.h>

int isLeapYear(int year);
int countDays(int day, int month, int year);
int daysBetween(int day1, int month1, int year1, int day2, int month2, int year2);

int main() {
    int day1, month1, year1;
    int day2, month2, year2;
    int days;

    printf("Enter the first date (dd mm yyyy): ");
    scanf("%d %d %d", &day1, &month1, &year1);

    printf("Enter the second date (dd mm yyyy): ");
    scanf("%d %d %d", &day2, &month2, &year2);

    days = daysBetween(day1, month1, year1, day2, month2, year2);
    printf("Number of days: %d\n", days);

    return 0;
}

int isLeapYear(int year)
{
    return (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0);
}
```

```
int countDays(int day, int month, int year)
{
    static int monthDays[12] = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};
    int days = year * 365 + day;

    for (int i = 0; i < month - 1; i++)
    {
        days += monthDays[i];
    }

    days += year / 4 - year / 100 + year / 400;

    if (month > 2 && isLeapYear(year)) {
        days++;
    }

    return days;
}

int daysBetween(int day1, int month1, int year1, int day2, int month2, int year2)
{
    return countDays(day2, month2, year2) - countDays(day1, month1, year1);
}
```

3) Exercise 3: A serial transmission line can transmit 960 characters each second. Write a program that will calculate the time required to send a file, given the file's size. Try the prog ram on a 400MB (419,430,400 -byte) file. Use appropriate units. (A 400MB file takes days.)

```
#include <stdio.h>

int main() {
   int transmission_speed = 960;
   int file_size;



   printf("Enter the file size: ");
   scanf("%d",&file_size);
   long bytes=file_size*1048576;



   long total_seconds = bytes / transmission_speed;





   long days = total_seconds / (24 * 3600);
   long hours = (total_seconds % (24 * 3600)) / 3600;
   long minutes = (total_seconds % 3600) / 60;
   long seconds = total_seconds % 60;



   printf("Time required to transmit the file of size %d mb:\n", file_size);
   printf("%ld days, %ld hours, %ld minutes, and %ld seconds.\n", days, hours, minutes, seconds);

   return 0;
}
```

4) Exercise 4: Write a program to add an 8% sales tax to a given amount and round the result to the nearest penny.

```
#include<stdio.h>

int main()

{

   float amt;

   float tax;

   printf("Enter the amount: ");

   scanf("%f",&amt);

   tax=amt+amt*0.08;

   printf("The toatal amount is %0.1f",tax);

   return 0;


}
```

5) Exercise 5: Write a program to tell if a number is prime.

```
#include<stdio.h>


void checkprime( int num);


int main()

{

   int  num;

   printf("Enter a number: ");

   scanf("%d",&num);
```

```
    checkprime(num);

    return 0;

}

void checkprime( int num)
{
    int a=0;
    for(int i=2;i<num;i++)
    {
        if(num%i==0)
        {
            a=1;
            break;
        }
    }

    if(a==1)
    {
        printf("number is not prime");

    }
    else{
        printf("number is  prime");
    }
}
```

6) Exercise 6: Write a program that takes a series of numbers and counts the number of positive and negative values.

```
#include<stdio.h>
int main()
{
   int n,num;
   int neg=0,pos=0;
   printf("Enter the limit: ");
   scanf("%d",&n);
   printf("Enter the numbers ");
   for(int i=1;i<=n;i++)
   {
      scanf("%d",&num);
      if(num<0)
      {
         neg=neg+1;
      }
      else
      {
         pos=pos+1;
      }
   }
   printf("number of positives are %d\n",pos);
   printf("number of positives are %d",neg);

   return 0;
}
```

7) C program to find the HCF of given numbers using recursion

```
#include<stdio.h>
int main()
{
   int x,y,result;
   printf("Enter 1st number: ");
   scanf("%d",&x);
   printf("Enter 2nd number: ");
   scanf("%d",&y);
   result =hcf(x,y);
   printf("%d",result);
   return 0;

}

int hcf(int x,int y)
{
   if(y==0)
   {
      return x;
   }
   else
   {
      return hcf(y,x%y);
   }
}
```

8) C program to find the LCM of given numbers using recursion

```
#include <stdio.h>

int lcm(int a, int b, int multiple) {
    if (multiple % a == 0 && multiple % b == 0)
        return multiple;
    else
    {
        return lcm(a, b, multiple + 1);
    }

}

#include<stdio.h>
int main()
{
    int x,y,result;
    printf("Enter 1st number: ");
    scanf("%d",&x);
    printf("Enter 2nd number: ");
    scanf("%d",&y);
    int m=0;
    if(x>y)
    {
        m=x;
    }
    else
```

```
  {
    m=y;
  }

  result = lcm(x, y, m);
  printf("%d",result);

  return 0;
}
```

10) C program to convert a Decimal number to Binary using Recursion

```
#include <stdio.h>

void decimalToBinary(int n)
{
  if (n == 0)
    return;
  else

    decimalToBinary(n / 2);
    printf("%d", n % 2);
}

int main() {
  int num;
```

```
    printf("Enter a decimal number: ");
    scanf("%d", &num);
    decimalToBinary(num);

    return 0;
}
```

11) C program to convert Binary Number to Gray Code

```
#include <stdio.h>
int binaryToGray (int num);
int main()
{
    int binary;
    printf("Enter a binary number: ");
    scanf("%d", &binary);
    int gray = binaryToGray (binary);
    printf("Binary %d to Gray Code = %d\n", binary, gray);
    return 0;
}
int binaryToGray (int num)
{
    return num ^ (num >> 1);
}
```

12) C program to convert Binary Number to Gray Code using Recursion

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

```
#include <math.h>

int bintogray(int bin)

{

  if (bin == 0 || bin == 1)

  {

  return bin;

  }

  int last_digit = bin % 10;

  int remaining_bin = bin / 10;

  int gray = bintogray(remaining_bin);

  return (last_digit ^ (gray % 10)) * pow(10, (int)log10(gray) + 1) + gray / 10;

}

int main()

{

  int bin, gray;

  printf("Enter a binary number: ");

  scanf("%d", &bin);

  gray = bintogray(bin);

  printf("The Gray code of %d is %d\n", bin, gray);

  return 0;

}
```

13) Print the pattern

```
**********

****  ****

***    ***

**      **

*        *
```

```
#include<stdio.h>
int main()
{
   int n=5;
   for(int i=n;i>=1;i--)
   {
      for(int j=1;j<=i;j++)
      {
         printf("*");
      }

      for(int j=5;j>i;j--)
      {
         printf(" ");
      }

      for(int j=5;j>i;j--)
      {
         printf(" ");
      }

      for(int j=1;j<=i;j++)
      {
         printf("*");
      }
      printf("\n");
   }
}
```

14) C program to find the sum of Natural Number/Factorial of Number of all natural number from 1 to N.

Series: 1/1! + 2/2! + 3/3! + 4/4! +... N/N!

```
#include<stdio.h>
int factorial(int num);

int main()
{
   int n;
   printf("Enter n: ");
   scanf("%d",&n);
   int fact;
   float s=0.0;

   for(int i=1;i<=n;i++)
   {
      fact=factorial(i);
      s=s+((float)i/fact);
   }

   printf("sum of series is %f",s);

   return 0;
}

int factorial(int num)
{
   int fact=1;
   for(int i=1;i<=num;i++)
   {
      fact=fact*i;
   }

   return fact;
}
```

15) C program to find sum of following series:

1 + 3^2/3^3 + 5^2/5^3 + 7^2/7^3 + ... till N terms

```
#include<stdio.h>
int factorial(int num);

int main()
{
   int n,a,b;
   float s=0;
   printf("Enter n: ");
   scanf("%d",&n);

   for(int i=1;i<=n;i=i+2)
   {
      a=i*i;
      b=i*i*i;
      s=s+(float)a/b;
   }
```

```
    printf("sum of series is %f",s);

    return 0;
}
```

16) C program to replace all EVEN elements by 0 and odd by 1 in one dimensional array

```
#include<stdio.h>

int main()
{
    int n;
    printf("enter the number of elements: ");
    scanf("%d",&n);
    int arr[n];

    printf("enter the elements");
    for(int i=0;i<n;i++)
    {
        scanf("%d",&arr[i]);
    }

    for(int i=0;i<n;i++)
    {
        if(arr[i]%2==0)
        {
            arr[i]=0;
        }
        else
        {
            arr[i]=1;
        }
    }

    printf("new array is");
    for(int i=0;i<n;i++)
    {
        printf("%d ",arr[i]);
    }
}
```

17) C program to read a matrix and print diagonals

```
#include<stdio.h>

int main()
{
    int n;
    printf("enter matrix size n: ");
    scanf("%d",&n);


    int arr[n][n];

    printf("enter the elements");
```

```
for(int i=0;i<n;i++)
{
   for(int j=0;j<n;j++)
   {
      scanf("%d",&arr[i][j]);
   }
}

printf("the matrix");
printf("\n");
for(int i=0;i<n;i++)
{
   for(int j=0;j<n;j++)
   {
      printf("%d ",arr[i][j]);
   }
   printf("\n");

}

printf("\n");
printf("the 1st diagonal elements are: \n");
for(int i=0;i<n;i++)
{
   for(int j=0;j<n;j++)
   {
      if(i==j)
      {
```

```
                printf("%d ",arr[i][j]);
            }
        }
    }
    printf("\n");
    printf("the 2nd diagonal elements are: \n");

    for(int i=0;i<n;i++)
    {
        for(int j=0;j<n;j++)
        {
            if(i+j==n-1)
            {
                printf("%d ",arr[i][j]);
            }
        }
    }

    return 0;
}
```

18) C program to print the upper triangular portion of a 3x3 matrix

```
#include <stdio.h>

int main() {
    int matrix[3][3];
    printf("Enter the elements of the 3x3 matrix:\n");
    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            scanf("%d", &matrix[i][j]);
        }
    }

    printf("Upper triangular portion of the matrix is:\n");
    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            if (i <= j) {
                printf("%d ", matrix[i][j]);
            }
            else {
                printf("  ");
            }
        }
        printf("\n");
    }

    return 0;
}
```