1. C Program to Check if a given Integer is Odd or Even

```
#include <stdio.h>
void main()
{
        int a;
        printf("Enter integer :");
        scanf ("%d",&a);
        if (a%2==0)
        printf(":%dis Even Number\n",a);
        else
                printf("%d:is Odd Number\n",a);
}
```

2. C Program to Calculate the Sum of Odd & Even Numbers

```
#include <stdio.h>
void main()
{
int count,sum,end;
printf("Enter No:");
scanf("%d",&end);
sum=0;
if (end%2==0)
        for(count=0; count<=end; count+=2)
        {
        sum+=count;
        printf("%d\n",sum);
        }
        printf("Total : %d\n",sum);
}
```

3. C Program to Check if a given Integer is Positive or Negative

```
#include <stdio.h>
void main()
{
        int a;
        printf("Enter integer :");
        scanf("%d",&a);
if(a>=0)
        printf("%d is positive\n",a);
else
        printf("%d is negative\n",a);
}
```

4. C Program to Find the Number of Integers Divisible by 5

```
#include <stdio.h>
void main()
{
        int a;
        printf("Enter number :");
        scanf("%d",&a);
if (a%5==0)
        printf("is divisible 5\n");
else
        printf("is can't divisibe 5\n");
}
```

5. C Program to Read Two Integers M and N & Swap their Values

```
#include <stdio.h>
void main()
{
        int M,N;
        printf("Enter M: ");
        scanf("%d",&N);

        printf("Enter N: ");
        scanf("%d",&M);
        temp= N;
        N= M;
        M= temp;

        printf("M is: %d\n",M);
        printf("N is: %d\n",N);
}
```

6. C Program to Accept two Integers and Check if they are Equal

```
#include <stdio.h>
void main()
{
        int a,b;
        printf("Enter 1st no: ");
        scanf("%d",&a);
        printf("Enter 2nd no: ");
        scanf("%d",&b);
        if(a==b)
                printf("Equal");
        else
                printf("not Equal");
}
```

7. C Program to Compute the Sum of Digits in a given Integer

```
#include <stdio.h>
void main()
{
        int a,balance,sum=0;
        printf("Enter 1st no: ");
        scanf("%d",&a);

        while (a != 0)
```

```
  {
    balance = a % 10;
    sum= sum + balance;

    a = a / 10;
  }

        printf("Sum: %d",a);
}
```

8. C Program to Convert the given Binary Number into Decimal
9. C Program to Convert a Decimal Number to Binary & Count the Number of 1s
10. C Program to Convert a Given Number of Days in terms of Years, Weeks & Days

```
#include <stdio.h>
void main()
{
  int days, year, week, days;

  printf("Enter the number of days\n");
  scanf("%d", &days);
  year = days / 365;
  week =(days % 365) / 7;
  days =(days % 365) % 7;
  printf (" %d years, %d weeks & %d days",
      days, year, week, days);
}
```

11. C Program to Convert Binary to Octal

```
#include <stdio.h>

int main()
{
    int b_num, o_num = 0, i = 1, balance;

    printf("Enter the binary number: ");
    scanf("%d", &b_num);
    while (b_num != 0)
    {
        balance = b_num % 10;
        o_num = o_num + balance * i;
        i = i * 2;
        b_num = b_num / 10;
    }
    printf("%o",o_num);
    return 0;
}
```

12. C Program to Convert Binary to Hexadecimal

```
#include <stdio.h>

int main()
{
    int b_num, h_num = 0, i = 1, balance;

    printf("Enter the binary number: ");
    scanf("%d", &b_num);
    while (b_num != 0)
    {
        balance = b_num % 10;
        h_num = h_num + balance * i;
        i = i * 2;
        b_num = b_num / 10;
    }
    printf("%X", h_num);
    return 0;
}
```

13. C Program to Convert Decimal to Octal

```
#include<stdio.h>
int main() {
    long int d_num;
    printf("Enter a decimal number: ");
    scanf("%d",&d_num);
    printf("Equivalent Ocatal number is: %o",d_num);
    return 0;
}
```

14. C program to Convert Decimal to Hexadecimal

```
#include<stdio.h>
int main() {
    long int d_num;
    printf("Enter a decimal number: ");
    scanf("%d",&d_num);
    printf("Equivalent hexadecimal number is: %X",d_num);
    return 0;
}
```

15. C Program to Convert Roman Number to Decimal Number

16. C Program to Convert Octal to Binary

```
#include <stdio.h>
#define y 1000

int main()
{
    char o_num[y];
    int i = 0;

    printf("Enter a octal number: ");
    scanf("%s", o_num);
    while (o_num[i])
    {
        switch (o_num[i])
        {
        case '0':
            printf("000"); break;
        case '1':
            printf("001"); break;
        case '2':
            printf("010"); break;
        case '3':
            printf("011"); break;
        case '4':
            printf("100"); break;
        case '5':
            printf("101"); break;
        case '6':
            printf("110"); break;
        case '7':
            printf("111"); break;
        default:
            printf("\n Invalid octal digit ");
            return 0;
        }
        i++;
    }
    return 0;
}
```

17. C Program to Convert Hexadecimal to Binary

```
#include <stdio.h>
#define y 1000
```

```
int main()
{
    char b_num[y], hexa[y];
    long int i = 0;

    printf("Enter a hexadecimal num ");
    scanf("%s", hexa);
    while (hexa[i])
    {
        switch (hexa[i])
        {
        case '0':
            printf("0000"); break;
        case '1':
            printf("0001"); break;
        case '2':
            printf("0010"); break;
        case '3':
            printf("0011"); break;
        case '4':
            printf("0100"); break;
        case '5':
            printf("0101"); break;
        case '6':
            printf("0110"); break;
        case '7':
            printf("0111"); break;
        case '8':
            printf("1000"); break;
        case '9':
            printf("1001"); break;
        case 'A':
```

```
      printf("1010"); break;
    case 'B':
      printf("1011"); break;
    case 'C':
      printf("1100"); break;
    case 'D':
      printf("1101"); break;
    case 'E':
      printf("1110"); break;
    case 'F':
      printf("1111"); break;
    case 'a':
      printf("1010"); break;
    case 'b':
      printf("1011"); break;
    case 'c':
      printf("1100"); break;
    case 'd':
      printf("1101"); break;
    case 'e':
      printf("1110"); break;
    case 'f':
      printf("1111"); break;
    default:
      printf("\n Invalid hexa num ");
      return 0;
    }
    i++;
  }
  return 0;
}
```

18. C Program to Convert Numbers to Roman Numerals

19. C Program to Convert Octal to Decimal

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

int main()
{

     int octal, decimal = 0;
    int i = 0;

    printf("Enter a octal number: ");
    scanf("%d", &octal);
    while (octal != 0)
    {
        decimal =  decimal +(octal % 10)* pow(8, i++);
        octal = octal / 10;
    }
    printf("%d",decimal);
    return 0;
}
```

20. C Program to Convert a Number Decimal System to Binary System using Recursion

```
#include <stdio.h>
int find(int dnum)
{
    if (dnum == 0)
        return 0;
    else
        return (dnum % 2 + 10 *
                find(dnum / 2));
}

int main()
```

```
{
   int dnum;
   scanf(“%d”,&dnum);
   printf("%d", find(dnum));
   return 0;
}
```

21. C Program to Convert Binary Code of a Number into its Equivalent Gray’s Code without using Recursion
22. C Program to Convert Binary Code of a Number into its Equivalent Gray’s Code using Recursion
23. C Program to find Sum of Digits of a Number using Recursion

```
#include <stdio.h>
int Total(int num)
{
   if (num == 0)
      return 0;
   return (num % 10 + Total(num / 10));
}
int main()
{
   int num ;
   scanf("%d",&num);
   printf("Sum is %d\n",Total(num));
   return 0;
}
```

24. C Program to find Reverse of a Number using Recursion

```
#include<stdio.h>
int sum=0,balance;

int main(){
  int num,num2;
  printf("Enter a number:\n");
  scanf("%d",&num);


  num2=rev_num(num);
  printf("%d",num2);
  return 0;
}

int rev_num(int num){
  if(num){
    balance=num%10;
    sum=sum*10+balance;
    rev_num(num/10);
  }
  else{
    return sum;
  }
  return sum;
}
```

25. C Program to find Sum of N Numbers using Recursion

```
#include <stdio.h>
int Tot_num(int n)
{
   if(n != 0)
      return n + Tot_num(n-1);
   else
      return n;
}

int main()
```

```
{
  int num;
  printf("Enter a number: ");
  scanf("%d", &num);
  printf("%d",Tot_num(num));
  return 0;
}
```

26. C Program to find whether a Number is Prime or Not using Recursion

```
#include <stdio.h>

int pnum(int num, int num2)
{
  if (num2 == 1)
  {
    return 1;
  }
  else
  {
   if (num % num2 == 0)
   {
    return 0;
   }
   else
   {
    return pnum(num, num2 - 1);
   }
  }
}

int main()
{
  int num, check;
```

```
    printf("Enter a number: ");
    scanf("%d", &num);
    check = pnum(num, num / 2);
    if (check == 1)
    {
        printf("%d is a prime number\n", num);
    }
    else
    {
        printf("%d is not a prime number\n", num);
    }
    return 0;
}
```

27. C Program to Print Binary Equivalent of an Integer using Recursion

```
int convert(int num)
{
    if (num == 0)
    {
        return 0;
    }
    else
    {
        return (num % 2) + 10 * convert(num / 2);
    }
}

int main()
{
   int num, Res;

   printf("Enter a  number: ");
```

```
  scanf("%d", &num);
  Res = convert(num);
  printf("%d\n", Res);
}
```

28. C Program to find Product of 2 Numbers using Recursion

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



int product(int, int);

int main()
{
   int a, b, result;

   printf("Enter two numbers : ");
   scanf("%d %d", &a, &b);
   result = product(a, b);
   printf("%d\n", a, b, result);
   return 0;
}

int product(int a, int b)
{
   if (a < b)
   {
      return product(b, a);
   }
   else if (b != 0)
   {
      return (a + product(a, b - 1));
```

```
    }
    else
    {
        return 0;
    }
}
```

29. C Program to Find the Biggest of 3 Numbers

```
#include <stdio.h>

void main()
{
    int num1, num2, num3;

    printf("Enter the 3 numbers \n");
    scanf("%d %d %d", &num1, &num2, &num3);
    printf("A = %d\tB = %d\tC = %d\n", num1, num2, num3);

        if (num1 > num3 && num1>num2)
        {
            printf("A is the greatest among three \n");
        }
        else if (num2 > num3 && num2>num1)
        {
            printf("B is the greatest among three \n");
        }
        else if (num3 > num2 && num3>num1)
        {
            printf("C is the greatest among three \n");
        }

}
```

30. C Program to Reverse a Given Number

```
#include <stdio.h>

void main()
{
   int  num, reverse = 0, balance;

   printf("Enter the number\n");
   scanf("%d", &num);
   while (num > 0)
   {
      balance = num % 10;  //2563  3
      reverse = reverse * 10 + balance;
      num /= 10;
   }

   printf("%d\n", reverse);
}
```

31. C Program to Reverse a Number & Check if it is a Palindrome

```
#include <stdio.h>

void main()
{
   int num, temp, balance, R_num = 0;

   printf("Enter a Number \n");
   scanf("%d", &num);

   temp = num;
   while (num > 0)
   {
      balance = num % 10;
      R_num = R_num * 10 + balance;
```

```
        num /= 10;
    }

    printf("%d\n", R_num);
    if (temp == R_num)
        printf("Number is a palindrome \n");
    else
        printf("Number is not a palindrome \n");
}
```

32. C Program to Find the Sum of two Binary Numbers
33. C Program to Find Multiplication of two Binary Numbers
34. C Program to find Product of 2 Numbers without using Recursion

```
#include <stdio.h>

int product(int a, int b)
{
    int temp = 0;

    while (b != 0)
    {
        temp += a;
        b--;
    }
    return temp;
}

int main()
{
    int a, b, result;

    printf("Enter two numbers: ");
    scanf("%d %d", &a, &b);
```

```
  result = product(a, b);
  printf("%d\n", result);
  return 0;
}
```

35. C Program to Check whether a given Number is Armstrong
36. C Program to Check whether a given Number is Perfect Number

```
#include <stdio.h>

int main()
{
  int num, balance, sum = 0, i;

  printf("Enter a Number\n");
  scanf("%d", &num);
  for (i = 1; i <= (num - 1); i++)
  {
    balance = num % i;
  if (balance == 0)
    {
      sum = sum + i;
    }
  }
  if (sum == num)
    printf("Entered Number is perfect number");
  else
    printf("Entered Number is not a perfect number");
  return 0;
}
```

37. C Program to Print Armstrong Number from 1 to 1000
38. C Program to Add two Complex Numbers
39. C Program to Generate Fibonacci Series of N Numbers using Command-Line Argument

40. C Program to Compute First N Fibonacci Numbers using Command Line Arguments

41. C Program to Find the Sum of first 50 Natural Numbers using For Loop

```
#include <stdio.h>

void main()
{
   int  num, sum = 0;

   for (num = 1; num <= 50; num++)
   {
      sum = sum + num;
   }
   printf("Sum = %4d\n", sum);
}
```

42. C Program to Swap the Contents of two Numbers using Bitwise XOR Operation

```
#include <stdio.h>

void main()
{
   int i, k,temp;

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

   temp= i;
   i = k;
   k = temp;
   printf("%d %d", i, k);
}
```

43. C Program to Multiply given Number by 4 using Bitwise Operators

44. C Program to Illustrate how User Authentication is Done

45. C Program to Display the IP Address of the System

46. C Program to Shutdown or Turn Off the Computer in Linux

47. C Program to Find if a given Year is a Leap Year

```
void main()
{
    int year;

    printf("Enter a year \n");
    scanf("%d", &year);
    if ((year % 400) == 0)
        printf("%d is a leap year \n", year);
    else if ((year % 100) == 0)
        printf("%d is a not leap year \n", year);
    else if ((year % 4) == 0)
        printf("%d is a leap year \n", year);
    else
        printf("%d is not a leap year \n", year);
}
```

48. C Program to Extract Last two Digits of a given Year

49. C Program to Display the Inventory of Items in a Store

50. C Program to Display the ATM Transaction

51. C Program to Accept the Height of a Person & Categorize as Taller, Dwarf & Average

52. C Program to Read a Grade & Display the Equivalent Description

53. C Program to Illustrate the Concept of Unions

54. C Program to Find the Size of a Union

55. C Program to Display Function without using the Main Function

56. C Program to Print a Semicolon without using a Semicolon anywhere in the Code

```
#include <stdio.h>

int main(void)
{
    if (printf("%c ", 59))
    {
    }
    return 0;
}
```

57. C program to Increase 1 to all of the given Integer Digit
58. C Program to Print Diamond Pattern
59. C Program to Print any Print Statement without using Semicolon
60. C Program to Display its own Source Code as its Output