

Assignment-5

Code:

```
// Q11. Write a program in C to check whether a number can be
expressed as the sum of two prime numbers
#include <stdio.h>
int prime_number(int x)
{
    for (int i=2;i<x;i++)
    {
        if (x%i == 0)
            return 0;
        else if (i+1 == x)
            return x;
    }
}
int main()
{
    int n;
    printf("Enter Number: ");
    scanf("%d",&n);
    int a = n / 2;
    for (int i = 2; i <= a; i++)
    {
        if (prime_number(i) && prime_number(n - i))
            printf("%d = %d + %d\n", n, i, n - i);
    }
    return 0;
}
```

Output:

```
Enter Number: 26
26 = 3 + 23
26 = 7 + 19
26 = 13 + 13
```

Code:

```
// Q12. Write a C program to check whether a number is a
// Armstrong Number or not.
#include <stdio.h>
#include <math.h>
int length(int x)
{
    int len =0;
    while (x>0)
    {
        x=x/10;
        len++;
    }
    return len;
}
void Armstrong_Number(int x)
{
    int l=length(x);
    int A[l],s=0,a=x;
    while (a>0)
    {
        int r=a%10;
        s=s+round(pow(r,l));
        a/=10;
    }
    if (s==x)
        printf("Number %d is an Armstrong Number.\n",x);
    else
        printf("Number %d is NOT an Armstrong Number.\n",x);
}
int main()
{
    int n;
    printf("Enter Number: ");
    scanf("%d",&n);
    Armstrong_Number(n);
    return 0;
}
```

Output:

```
Enter Number: 8208
Number 8208 is an Armstrong Number.
```

```
Enter Number: 562
Number 562 is NOT an Armstrong Number.
```

Code:

```
// Q13. Write C program to convert binary to hexadecimal &
vice versa
#include <stdio.h>
#include <math.h>
int Binary_to_Decimal(int x)
{
    int A[32],i=0,s=0;
    while (x>0)
    {
        A[i]=x%10;
        x/=10;
        if (A[i]==1)
            s+=A[i]*pow(2,i);
        i++;
    }
    return s;
}
int Decimal_to_Binary(int x)
{
    int A[64],i=0;
    if (x==0)
        return 0;
    while (x>0)
    {
        A[i]=x%2;
        x/=2;
        i++;
    }
    for (int j=i-1;j>=0;j--)
        printf("%d",A[j]);
}
```

```

int main()
{
    int c,n;
    printf("Enter\n1. for Binary to Hexadecimal, or\n2.
    Hexadecimal to Binary: ");
    scanf("%d",&c);
    switch(c)
    {
        case 1:
        {
            printf("Enter Binary Number: ");
            scanf("%d",&n);
            int d=Binary_to_Decimal(n);
            printf("Binary: %d --> Hexadecimal: %X\n",d,d);
            break;
        }
        case 2:
        {
            char A[16],B[32];
            printf("Enter Hexadecimal Number: ");
            scanf("%s",&A);
            sscanf(A,"%X",&n);
            printf("Hexadecimal: %s --> Binary: ",A);
            Decimal_to_Binary(n);
            break;
        }
    }
}

```

Output:

```

Enter
1. for Binary to Hexadecimal, or
2. Hexadecimal to Binary: 1
Enter Binary Number: 11011
Binary: 27 --> Hexadecimal: 1B

```

```

Enter
1. for Binary to Hexadecimal, or
2. Hexadecimal to Binary: 2
Enter Hexadecimal Number: 11E
Hexadecimal: 11E --> Binary: 100011110

```

Code:

```
// Q14. Write a program in C to find the number and sum of
all integers between 100 and 200 which are divisible by 9.
#include <stdio.h>
int main()
{
    int s=0;
    printf("Number divisible by 9 are: \n");
    for (int i=100;i<=200;i++)
    {
        if (i%9 == 0)
        {
            printf("%d ",i);
            s+=i;
        }
    }
    printf("\nSum: %d",s);
}
```

Output:

```
Number divisible by 9 are:
108 117 126 135 144 153 162 171 180 189 198
Sum: 1683
```

Code:

```
// Q15. Write a program to find out the second largest factor of a
number. Eg. Input->100, Output-> 50.
#include <stdio.h>
int main()
{
    int n,index=0;
    printf("Enter Number: ");
    scanf("%d",&n);
    int A[n/2];
    for (int i=1;i<=n;i++)
    {
        if (n%i == 0)
        {
            A[index]=i;
            index++;
        }
    }
    printf("Second largest Factor of %d is %d.",n,A[index-2]);
}
```

Output:

```
Enter Number: 75
Second largest Factor of 75 is 25.
```