Write a program to promote the user to input 3-digit no. & print these value in forward & reversed order.

```
# include <stdio.h>

void main() {

        int num, rem, rev=0, fo;
        printf("Enter a three digit no: ");
        scanf("%d", &num);
        fo = num;
        rem = num%10;
        rev = rev*10+rem;
        num = num/10;
        rem = num%10;
        rev = rev*10+rem;
        num = num/10;
        rem = num%10;
        rev = rev*10+rem;
        num = num/10;
    printf("Forword order = %d\n", fo);
    printf("Reversed order = %d", rev);
    getch();
}
```

Enter a three digit no. 653
Forword order = 653
Reversed order = 356

Write a program to swap two variable value with & without using third variable

```
#include <stdio.h>
#include <conio.h>
int main()
  {
    int  x, y;
    printf ("Enter the value of x & y ?");
    scanf ("%d %d", &x, &y);
    printf ("Before swapping numbers: %d.%d \n", x, y);
     x = x+y;
     y = x-y;
     x = x-y;
     printf(" After swapping : %d %d", x, y);
     return 0;
  }
```

Output

```
Enter the value of x & y ?
13
23
```

Output

```
Before swapping number:
        13  22
After swaping :
        22  13
```

Write a program to find given no. is palindrome or not.

```
# include <stdio.h>
# include <conio.h>
 int main()
 {
    int n, reversedN = 0, rem, originalN;
    printf("Enter an integer: ");
    scanf("%d", &n);
     originalN = n;
     while (n != 0)
         {
            rem = n % 10;
            reversedN = reversedN * 10 + rem;
             n /= 10;
         }
       if (originalN == reversedN)
        printf("%d is a palindrome.", originalN);
       else
        printf("%d is not palindrome.", originalN);
       return 0;
    }
```

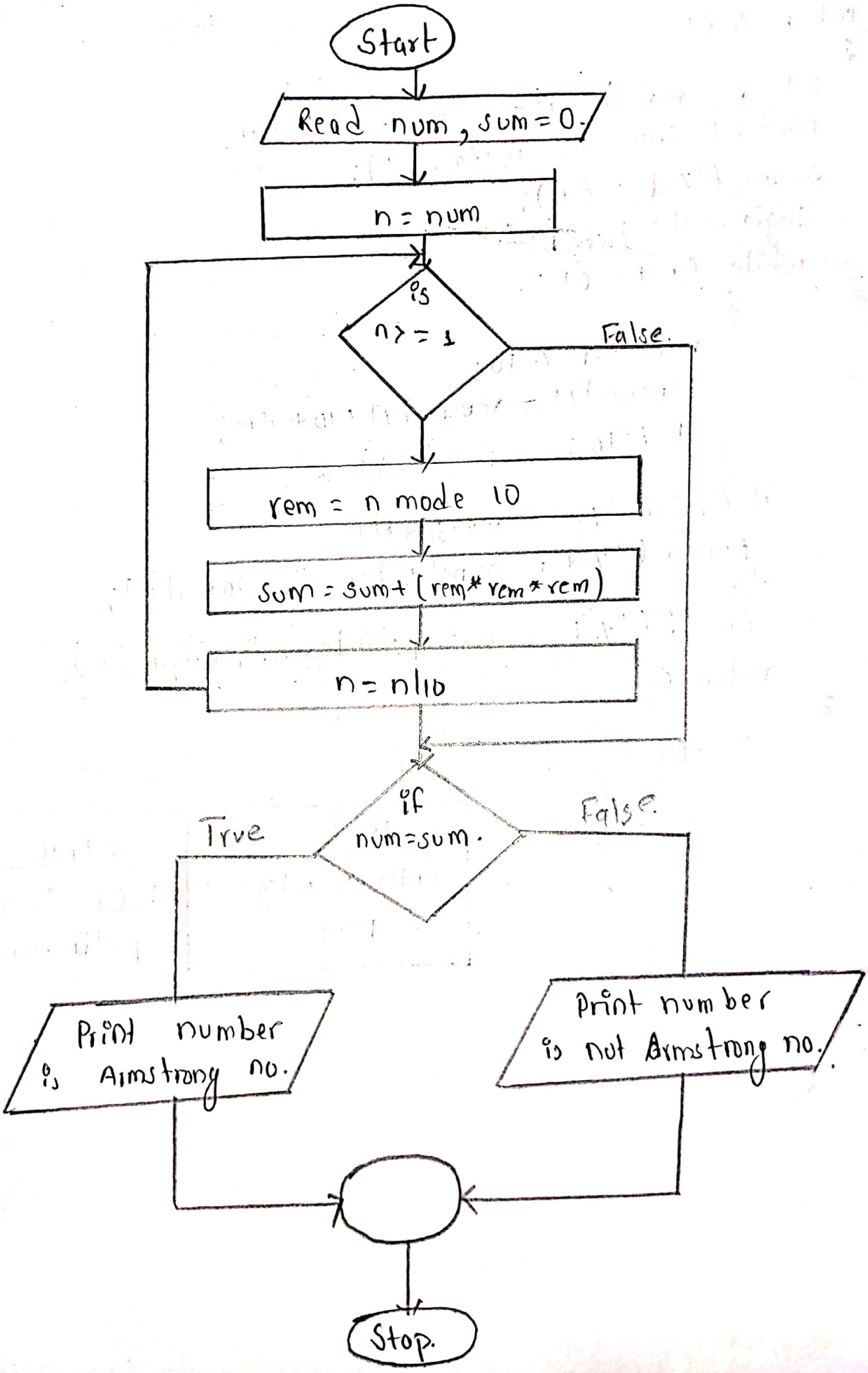
input	output
Enter an integer : 1001	1001 is a palindrome.

Write a program to find given no. is Armstrong or not.

```
#include <stdio.h>
int main() {
    int num, originalNum, rem, r = 0;
    printf("Enter a 3 digit no: ");
    scanf("%d", &num);
    originalNum = num;
    while (originalNum != 0)
    {
        rem = originalNum % 10;
        r += rem * rem * rem;
        originalNum /= 10;
    }
    if (result == num)
        printf("%d is Armstrong no.", num);
    else.
        printf("%d is not Armstrong no.", num);
    return 0;
}
```

input	Output
Enter a 3 digit no.: 371	371 is an Armstrong no.

Flowchart for Armstrong number



Write a Algorithm to find given no. is Armstrong or not.

Step 1: Start

Step 2: Declear variable sum, temp, num.

Step 3: Read num from user.

Step 4: Initialize variable sum = 0 & temp = num.

Step 5: Repeat until num >= 0

5.1: sum = sum + cube of last digit ie. T(num%10) * (num%10) * (num%10) J.

5.2: num = num/10

Step 6: If sum == temp.

print "Armstrong Number".

Else.

Print "Not Armstrong Number".

Step 7: Stop

Write a program to check odd or even no.

a) Using Modulus operator.

```
# include<stdio.h>
   void main() {
        int num, rem;
        printf ("Enter a no.=");
        scanf ("%d", &num);
        rem = num %.2
         rem = 0 ?
         Printf ("Entered no. is even");
             Else
         Printf ("Entered no. is odd");
       getch ();
    }
```

Enter a no. = 6
Entered no. is even.

b) Using conditional operator.

```
# include <stdio.h>
  void main () {
       int num, rem;
        printf("Enter a no.=");
        scanf ("%d", &num);
         rem = num %. 2;
         if(rem == 0)
       {
         printf ("Entered no. is even");
       }
        else {
            Printf("Entered no. is odd");
        }
       getch();
  }
```

Enter a no. = 10
Entered no. is even.

Write an algorithm to check odd or even no.

i) Using modulus operator.

Step 1: Start

Step 2: Declare variable num, rem

Step 3: Input value of num.

Step 4: rem = num % 2

Step 5: If rem == 0 then goto step 6.
Else goto step 7.

Step 6: Display num is even no.

Step 7: Display num is odd no.

Step 8: stop.

ii) Using conditional operator.

Step 1: Start

Step 2: Declare variables num, rem.

Step 3: Input value of num

Step 4: rem = rem % 2

Step 5: rem == 0 ? Print("Even no.") : print("odd no.")

Step 6: Stop.

#Write an algorithm to print size of char, float double & long double data types in C.

Step 1: Start

Step 2: Declare variables of char as 'ch' float as 'b' double as 'c' & long double as 'd'.

Step 3: Display space occupied of char is size of(ch)

Step 3.1: Display space occupied of float is size of(b)

Step 3.2: Display space occupied of double is size of(c)

Step 3.3: Display space occupied of long double is size of(d).

Step 4: Stop.

```
# include<stdio.h>
void main() {
        char ch;
        float b; double c; long double d;
        printf("space occupied by char data types is %d\n",
                                        size of(ch));
        Printf("space occupied by float data type is %d\n",
                                        size of(b));
        Print f("space occupied by double data type is %d\n",
                                        size of(c));
        Print("space occupied by long double data type is %d\n",
                                        size of(d));
        getch();
    }
```

Output

```
Space occupied by char data type is 1
Space occupied by float data type is 4
Space occupied by double data type is 8
Space occupied by long double data type is 16
```

Conclusion & Discussion.

In this second lab of C Programming based or the focused objective to understand about C data type with formatted input / output function the additional lab exercise made me more confident towards the fulfillment of the objectives.