

**WEEK:05-02**

**ROLL NO: 240801161**

**NAME: KEERTHANA S**

Status	Finished
Started	Monday, 23 December 2024, 5:33 PM
Completed	Friday, 20 December 2024, 10:02 PM
Duration	2 days 19 hours

**QUESTION:1**

## **ARMSTRONG NUMBER**

### **Problem Statement:**

The k-digit number N is an Armstrong number if and only if the k-th power of each digit sums to N.

Given a positive integer N, return true if and only if it is an Armstrong number.

Note:  $1 \leq N \leq 10^8$

Hint: 153 is a 3-digit number, and  $153 = 1^3 + 5^3 + 3^3$ .

### **Sample Input:**

153

### **Sample Output:**

true

### **Sample Input:**

123

### **Sample Output:**

false

### **Sample Input:**

1634

### **Sample Output:**

True

## PROGRAM:

Answer: (penalty regime: 0 %)

```
1  #include<stdio.h>
2  #include<math.h>
3  int main()
4  {
5      int n;
6      scanf("%d",&n);
7      int x=0,n2=n;
8      while (n2!=0)
9      {
10         x++;
11         n2= n2/10;
12     }
13     int sum =0;
14     int n3= n ,n4;
15     while (n3!=0)
16     {
17         n4=n3%10;
18         sum = sum + pow(n4,x);
19         n3= n3/10;
20     }
21     if (n == sum)
22     {
23         printf("true");
24     }
25     else{
26         printf("false");
27     }
28     return 0;
29 }
30 }
```

## OUTPUT:

	Input	Expected	Got	
✓	153	true	true	✓
✓	123	false	false	✓

Passed all tests! ✓

## **QUESTION: 2**

### **REVERSE AND ADD UNTIL GET A PALINDROME**

#### **Problem Statement:**

Take a number, reverse it and add it to the original number until the obtained number is a palindrome.

#### **Constraints**

$1 \leq \text{num} \leq 999999999$

#### **Sample Input 1**

32

#### **Sample Output 1**

55

#### **Sample Input 2**

789

#### **Sample Output 2**

66066

## PROGRAM:

Answer: (penalty regime: 0 %)

```
1 #include<stdio.h>
2 int main()
3 {
4     int rev, n, nt=0,i=0;
5     scanf("%d",&n);
6     do{
7         nt = n;
8         rev = 0;
9         while(n!=0)
10        {
11            rev= rev*10+n%10;
12            n=n/10;
13        }
14        n = nt+rev;
15        i++;
16    }
17    while(rev!=nt || i==1);
18    printf("%d",rev);
19    return 0;
20 }
21
```

## OUTPUT:

	Input	Expected	Got	
✓	32	55	55	✓
✓	789	66066	66066	✓

Passed all tests! ✓

**QUESTION:3****LUCKY NUMBER****Problem Statement:**

A number is considered lucky if it contains either 3 or 4 or 3 and 4 both in it. Write a program to print the nth lucky number. Example, 1st lucky number is 3, and 2nd lucky number is 4 and 3rd lucky number is 33 and 4th lucky number is 34 and so on. Note that 13, 40 etc., are not lucky as they have other numbers in it.

The program should accept a number 'n' as input and display the nth lucky number as output.

**Sample Input 1:**

3

**Sample Output 1:**

33

## PROGRAM:

Answer: (penalty regime: 0 %)

```
1  #include<stdio.h>
2  int main()
3  {
4      int n=1,i=0,nt,co,e;
5      scanf("%d",&e);
6      while (i<e)
7      {
8          nt= n;
9          while (nt!=0)
10         {
11             co=0;
12             if (nt%10!=3 && nt%10!=4)
13             {
14                 co=1;
15                 break;
16             }
17             nt=nt/10;
18         }
19         if(co==0)
20         {
21             i++;
22         }
23         n++;
24     }
25     printf("%d",--n);
26     return 0;
27 }
28
```

## OUTPUT:

	Input	Expected	Got	
✓	34	33344	33344	✓

Passed all tests! ✓