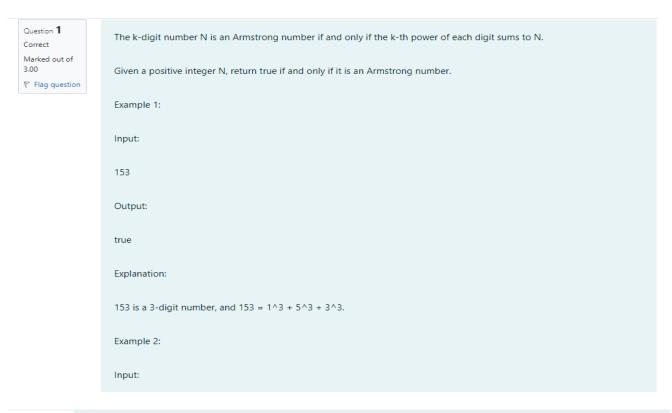
Name: Kirithick Raja

Roll no.: 240701627

Week-05-S02

Question 1:



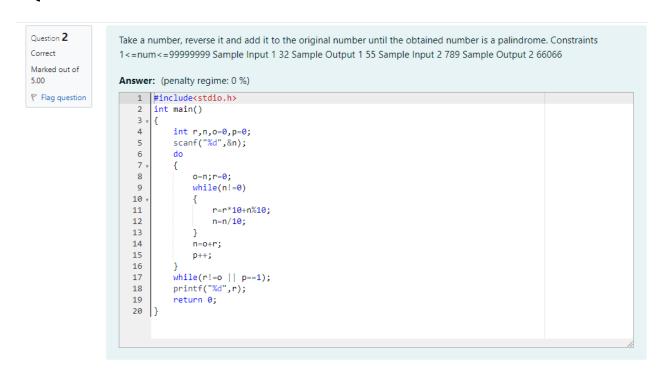
Input:
153
Output:
true
Explanation:
153 is a 3-digit number, and 153 = 1^3 + 5^3 + 3^3.
Example 2:
Input:
123
Output:
false

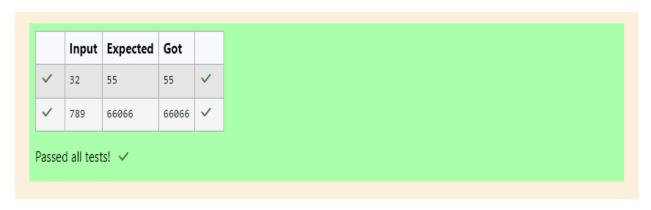
Source code:

```
Answer: (penalty regime: 0 %)
   1 #include<stdio.h>
       #include<math.h>
   3
      int main()
   4 * {
           int N;
scanf("%d",&N);
   5
   6
           int m=0,o=N;
   7
   8
           while(o!=0)
   9,
           {
  10
              m++;
  11
              o=o/10;
  12
           int sum=0;
  13
  14
           int o1=N,o4;
  15
           while(o1!=0)
  16 🔻
               04=01%10;
  17
  18
               sum=sum+pow(o4,m);
  19
               01=01/10;
  20
  21
           if(N==sum)
  22 ,
  23
               printf("true");
  24
  25
           else
  26 ,
  27
              printf("false");
  28
           return 0;
  29
  30 }
```

	Input	Expected	Got	
~	153	true	true	~
~	123	false	false	~
asse	d all test	s! 🗸		

Question 2 with source code:





Question 3:

```
Correct
Marked out of 7.00

P Rag question

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

Explanation:

Here the lucky numbers are 3, 4, 33, 34,, and the 3rd lucky number is 33.

Sample Input 2:
```

Source code:

```
Answer: (penalty regime: 0 %)
       #include<stdio.h>
        int main()
    3 ,
       {
            int m=1,n=0,o,p=0,q;
scanf("%d",&q);
    4
    5
    6
            while(n<q)
    8
                o=m;
    9
                while(o!=0)
   10
   11
                     p=0;
                     if(0%10!=3 && 0%10!=4)
   12
   13
   14
                         p=1;
   15
                         break;
   16
   17
                     o=o/10;
   18
                if(p==0)
   19
   20
   21
                     n++;
   22
   23
                m++;
   24
   25
            printf("%d",--m);
   26
            return 0;
   27 }
```

	Input	Expected	Got	
~	34	33344	33344	~

Passed all tests! 🗸