Week-05-Nested Loops - while and for, Jumps in Loops

Week-05-02-Practice Session Coding

Question 1
Correct
Marked out of 3.00

Flag question

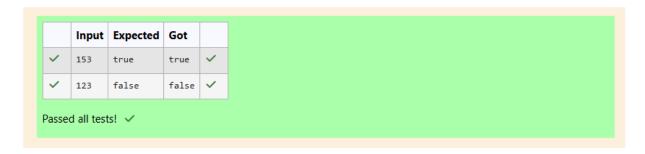
The k-digit number N is an Armstrong number if and only if the k-th power of each digit sums to N.

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

Source Code

```
Answer: (penalty regime: 0 %)
   1 #include<stdio.h>
      #include<math.h>
      int main()
   4 🔻 {
          int n,copy,sum=0,count,c;
          scanf("%d",&n);
   6
          copy =n;
   8
   9
          while(copy>0)
  10 🔻
              copy =copy/10;
  11
  12
              count++;
          }
  13
  14
          copy =n;
  15
          while(copy>0)
  17 -
  18
              c=copy%10;
              sum=sum+pow((c),count);
  19
  20
              copy = copy/10;
  21
           if(sum ==n)
  22
  23 v
           {
              printf("true");
  24
  25
  26
           else
  27 1
              printf("false");
  28
  29
  30
           return 0;
  31 }
```

Result



Question **2** Correct

Take a number, reverse it and add it to the original number until the obtained number is a palindrome. Constraints 1<=num<=99999999 Sample Input 1 32 Sample Output 1 55 Sample Input 2 789 Sample Output 2 66066

Source Code

```
Answer: (penalty regime: 0 %)
   1 #include<stdio.h>
   2
      int main()
   3 √ {
   4
           long long int num, sum, revnum, tempnum, tempsum;
   5
           scanf("%lld",&num);
   6
          while(1)
   7 .
   8
               revnum = 0;
               tempnum =num;
   9
  10
               while(num)
  11 •
               {
                  revnum = revnum *10+(num%10);
  12
  13
                  num = num/10;
  14
  15
               sum = tempnum + revnum;
               tempsum = sum;
  16
  17
               revnum =0;
  18
               while(sum)
  19 •
  20
                   revnum = revnum*10+(sum%10);
  21
                   sum = sum/10;
  22
               if(tempsum == revnum)
  23
  24 v
               {
  25
                   break;
  26
  27
               num = tempsum;
  28
           printf("%1ld",tempsum);
  29
           return 0;
  30
  31 }
```

Result

	Input	Expected	Got	
~	32	55	55	~
~	789	66066	66066	~
assed all tests! 🗸				

Question **3**Correct
Marked out of 7.00

Flag 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.

Source Code

```
Answer: (penalty regime: 0 %)
      #include<stdio.h>
      int main()
   3 ₹ {
   4
           long int i,j;
           int rem,n,count=0,flag;
   5
    6
           scanf("%d",&n);
   7
           for(i=1;count<=n;i++)</pre>
   8 ,
   9
               flag=0;
   10
               j=i;
   11
               while(j>0)
   12 ,
   13
                   rem = j\%10;
                   if(rem == 3 || rem == 4)
  14
   15
                   j = j/10;
   16
                   else
   17
                    {
                        flag=1;
   18
   19
                        break;
   20
   21
               if(flag==0)
   22
   23
   24
                    count++;
   25
                    if(count==n)
   26
                    break;
   27
   28
           printf("%ld",i);
   29
   30
           return 0;
   31 }
```

Result

