

 Doll Show

✓ Done

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

 Week-05-01-Practice Session-Coding

✓ Done

 Week-05-02-Practice Session-Coding

✓ Done

Quiz navigation

1 2 3

Show one page at a time

Back Review

Status

Finished

Started

Monday, 23 December 2024, 5:53 PM

Completed

Thursday, 21 November 2024, 9:46 PM

Duration

31 days, 19 hours

Question 1

Correct

Marked out of 100

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.

Example 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

Explanation:

123 is a 3-digit number, and $123 \neq 1^3 + 2^3 + 3^3 = 36$.

Example 3:

Input:

1634

Output:

true

Note:

$1 \leq N \leq 10^8$

Answer: (penalty regime: 0 %)

```
1 #include <stdio.h>
2 #include <math.h>
3 int main() {
4     int N;
5     scanf("%d", &N);
6     int sum = 0;
7     while(N > 0) {
8         int digit = N % 10;
9         sum += pow(digit, 3);
10        N /= 10;
11    }
12    int sum2 = sum;
13    while(N > 0) {
14        int digit = N % 10;
15        sum += pow(digit, 3);
16        N /= 10;
17    }
18    if (sum == sum2) {
19        printf("true");
20    } else {
21        printf("false");
22    }
23 }
```

Input	Expected	Got	
153	true	true	✓
123	false	false	✓

Passed all tests! ✓

Question 2

Correct

Marked out of 100

Flag question

Take a number, reverse it and add it to the original number until the obtained number is a palindrome. Constraints: $1 \leq \text{num} \leq 99999999$. Sample Input 1: 32 Sample Output: 1 55 Sample Input 2: 789 Sample Output 2: 66066

Answer: (penalty regime: 0 %)

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

Input	Expected	Got	
32	55	55	✓
789	66066	66066	✓

Passed all tests! ✓

Question 3

Correct

Marked out of 700

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, 43 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:

34

Sample Output 2:

33344

Answer: (penalty regime: 0 %)

```
1 #include <stdio.h>
2 int main()
3 {
4     int n = 1, i, j, k, l, m, count = 0;
5     scanf("%d", &n);
6     while (count < n)
7     {
8         int num = 0;
9         int temp = n;
10        while (temp > 0)
11        {
12            if (temp % 10 != 3 && temp % 10 != 4)
13                break;
14            num = num * 10 + temp % 10;
15            temp /= 10;
16        }
17        if (count < n)
18            count++;
19        n = num;
20    }
21    printf("%d", n);
22 }
```

Input	Expected	Got	
34	33344	33344	✓

Passed all tests! ✓

Finish Review