

Laboratory 05 – loops

03.11.2020

1. [3points] Write a C program to input an integer number from user (`scanf`) and count number of digits in the given integer using `do-while` loop. To check the result, use the logarithm and the ternary operator (`? :`) to count the number of digits in a given integer.

- a) What arithmetic operation should be used to count digits in an integer?
- b) What happens to the value of a user-supplied variable after counting the digits?
- c) How many variables are needed?

Test data:

Enter any number: 123456789

Total digits: 9

Total digits (log10): 9

Enter any number: 0

Total digits: 1

Total digits (log10): 1

2. [4points] Write a C program to check if user-specified integer is a palindrome.

A palindrome is a word, number, phrase, or other sequence of characters which reads the same backward as forward, such as madam, racecar.

There are also numeric palindromes: 121, 12321, ...

The algorithm:

- a) Use the `scanf` function to input an integer (`n`).
- b) Make a copy of `n`.
- c) Find the reverse of `n` and assign it to `rev`.

Example: Reverse of 1234 is 4321.

The `rev` calculation should be performed in the `while` loop.

What will be the loop termination condition?

The body of the `while` loop consists of two lines.

- The first line is the calculation of `rev`.

- The second line is necessary for the loop to end.

To calculate the `rev` values use: `*`, `+`, `%`, `rev`, `n`, `10`.

- d) Check if `rev` is equal to copy of `n`. If so, the number is a palindrome, if not, it is not.

Test data:

Enter any number to check palindrome: 1234

1234 is not palindrome.

Enter any number to check palindrome: 121

121 is palindrome.

Enter any number to check palindrome: 12321

12321 is palindrome.

3. [2points] Write a C program to find power of a number using for loop.

- Use the scanf function to input base and exponent.
- Multiply base, exponent times.
- Print the result.
- Compare with the result returned by function pow.

The pow function takes two arguments (base value and power value) and, returns the power raised to the base number.

Test data:

```
Enter base: 2
Enter exponent: 5
for-loop power(2, 5) = 32
pow(2, 5) = 32.00
```

```
Enter base: 12
Enter exponent: 0
for-loop power(12, 0) = 1
pow(12, 0) = 1.00
```

4. [3points] Write a C program to input a number from user and find all factors of the given number using for loop.

- Use the scanf function to input an integer (n).
- If n is exactly divisible by i, then i is a factor of n.
- Use an additional variable to count the number of factors of n.

Test data:

```
Enter any number to find its factor: 12
All factors of 12 are:
1 2 3 4 6 12
12 has 6 factors.
```

```
All factors of 100 are:
1 2 4 5 10 20 25 50 100
100 has 9 factors.
```

```
All factors of 60 are:
1 2 3 4 5 6 10 12 15 20 30 60
60 has 12 factors.
```

5. [5points] Write a program in C to input a number and check whether the number is **prime** number or not using for loop.

An integer *greater than one* is called a **prime** number if its only positive divisors (factors) are one and itself.

The algorithm:

- Use the scanf function to input an integer (n).
- Declare and initialize another variable say isPrime = 1. isPrime is used as a notification or flag variable. Assigning 0 means number is composite and 1 means number is prime.

c) Run a loop from 2 to $n/2$.

d) Check the divisibility of the number n .

If the remainder is zero then, the number is not prime.

Set `isPrime = 0` indicating number is not prime and terminate the loop using **break**.

e) Outside the loop check the value of `isPrime`. If `isPrime` is equal to 1 then the number is prime otherwise composite.

https://en.wikipedia.org/wiki/List_of_prime_numbers

Test data:

Enter any number to check prime: 1234
1234 is composite number.

Enter any number to check prime: 1123
1123 is prime number.

6. [3points] Modify the previous program to calculate how many primes there are from 1 to 1001. Find out how many primes there are between 1 and 101, 101 and 201, etc.

Test data:

There are 25 primes between 1 and 101.
There are 21 primes between 101 and 201.
There are 16 primes between 201 and 301.
There are 16 primes between 301 and 401.
There are 17 primes between 401 and 501.
There are 14 primes between 501 and 601.
There are 16 primes between 601 and 701.
There are 14 primes between 701 and 801.
There are 15 primes between 801 and 901.
There are 14 primes between 901 and 1001.

Next time:

Laboratory 06 - Arrays