

Лабораторная работа 3

Циклы

Цель этой лабораторной работы — изучить понятие цикла и научиться записывать различные циклы в языке Си.

ОБЩИЕ ЗАДАНИЯ

1. Напишите программу, которая вводит с клавиатуры 10 целых чисел и выводит их среднее арифметическое.
2. Напишите программу, которая вводит с клавиатуры число n от 1 до 20 включительно и печатает пирамиду из решеток и пробелов высоты n , например для $n = 3$ программа должна напечатать три строки:

```
#  
###  
#####
```

3. Напишите программу, которая вводит с клавиатуры целое число типа `long long` (в `scanf` указывать код формата `%lld`) и выводит его цифры в обратном порядке.
4. Для быстрой офлайновой (без связи с банком) проверки правильности написания номера кредитной карты используется алгоритм Х. П. Луна: умножим, двигаясь справа налево, каждую вторую цифру номера на 2. Сложим все цифры полученных чисел (внимание, не сами числа!). Теперь прибавим к ним сумму остальных цифр. Если последняя цифра полученной общей суммы не ноль, номер неправильный. Пример: номер 4372 2822 4431 0005 верный: удвоения подчеркнутых цифр равны 8, 14, 4, 4, 8, 6, 0, 0, их цифры в сумме дают $8 + (1 + 4) + 4 + 4 + 8 + 6 + 0 + 0 = 35$, сумма неподчеркнутых цифр номера равна $3 + 2 + 8 + 2 + 4 + 1 + 0 + 5 = 25$, а $35 + 25 = 60$.

Напишите программу, которая вводит номер карты как целое число (тип `long long`, форматный спецификатор `%lld`, **НЕ** как строку) и печатает `VALID` или `INVALID`, если номер, соответственно, верный или неверный по алгоритму Луна. Проверки будут производиться на числах, содержащих от 13 до 16 цифр.

5. К волшебному пределу $e = \lim_{n \rightarrow \infty} (1 + \frac{1}{n})^n$ («два и семь и дважды год рождения Тостого») можно приблизиться по-другому: ряд

$$1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^n}{n!} + \dots \quad (*)$$

сходится к e^x при любых x .

Напишите программу, которая вводит с клавиатуры вещественное x (используйте тип `double`), вычисляет для данного x частичную сумму ряда (*), в которой участвуют только те слагаемые ряда, которые не меньше 0.0001 (слагаемые монотонно убывают), и выводит модуль разности между вычисленной суммой и значением библиотечной функции $\exp(x)$ из заголовочного файла `cmath/math.h` (вывести не менее 6 знаков после десятичной точки).

Домашнее задание 3

Всего в этом домашнем задании 5 пунктов. Из каждого пункта надо сделать по одной задаче, в зависимости от вашего номера в списке группы. Считать по кругу: если в пункте 6 задач, то 7-й номер делает первую задачу, 8-й номер вторую и т.д.

Задачи

- 1.1. Напишите программу, которая вводит целое число n , $1 \leq n \leq 20$, и выводит текстовое поле размера $n \times n$ из точек и решеток в шахматном порядке, причем в левом верхнем углу находится решетка. Например, для $n = 5$ вывести
#.#.#
.#.#.
#.#.#
.#.#.
#.#.#
- 1.2. Напишите программу, которая вводит целое число n , $1 \leq n \leq 20$, и выводит текстовое поле размера $n \times n$ из точек и решеток в шахматном порядке, причем в левом верхнем углу находится точка. Например, для $n = 5$ вывести
.#.#.
#.#.#
.#.#.
#.#.#
.#.#.
- 1.3. Напишите программу, которая вводит целое число n , $2 \leq n \leq 20$, и выводит рамку размера $n \times n$ из решеток. Например, для $n = 4$ вывести

```
####
#  #
#  #
####
```

- 1.4. Напишите программу, которая вводит целое число n , $2 \leq n \leq 20$, и выводит контур треугольника высоты n с основанием ширины $2n$. Основание изображается подчеркиванием, стороны — прямыми и обратными косыми чертами. Например, для $n = 5$ вывести

```
  /\
 /  \
/    \
=====
```

-
- 2.1. Напишите программу, которая вводит 10 целых чисел и печатает через пробел количество положительных, отрицательных и нулей среди них. Например для ввода 1 -2 5 3 0 -3 -3 7 8 -1 вывести 5 4 1.
- 2.2. Напишите программу, которая вводит 10 неотрицательных чисел и печатает максимум среди четных из них, либо -1, если четных чисел нет. Например для ввода 1 2 5 3 0 3 0 7 8 9 вывести 8.
- 2.3. Напишите программу, которая вводит 10 неотрицательных чисел и печатает максимум среди нечетных из них, либо -1, если нечетных чисел нет. Например для ввода 1 2 5 3 0 3 0 8 4 7 вывести 7.
- 2.4. Напишите программу, которая вводит 10 неотрицательных чисел и печатает число таких, которые не меньше своих соседей. Например для ввода 1 2 5 3 0 4 0 7 8 9 вывести 3 (числа 5, 4, 9).
- 2.5. Напишите программу, которая вводит 10 целых чисел и печатает число с максимальным модулем. Например для ввода 1 -2 5 3 0 -3 -3 7 -8 -1 вывести -8.
-

- 3.1. Натуральное число из n цифр называется числом Армстронга, если сумма его цифр, возведенных в n -ю степень, равна самому числу (например $153 = 1^3 + 5^3 + 3^3$). Напечатать в возрастающем порядке все числа Армстронга, состоящие из четырех цифр.
- 3.2. Числа Фибоначчи определяются формулами: $F_0 = 0$; $F_1 = 1$; $F_n = F_{n-1} + F_{n-2}$, $n = 2, 3, \dots$. Ввести с клавиатуры целое число M и выве-

сти через пробел первое число Фибоначчи, большее M , и его номер k . Например, для $M = 5$ следует вывести 8 6.

- 3.3. Даны натуральные числа m и n . Найти наименьшее общее кратное этих чисел. Например, для входных данных 6 20 вывести 60.
 - 3.4. Ввести с клавиатуры целое положительное число k и напечатать k -е простое число. Например, для $k = 5$ вывести 11.
 - 3.5. Дано целое число $n \geq 0$. Вывести сумму его цифр.
 - 3.6. Найти и вывести через пробел в одной строке наибольшую и наименьшую цифры в записи данного натурального числа n .
-

- 4.1. Напишите программу, которая вводит с клавиатуры четыре числа d_1, m_1, d_2, m_2 — число и месяц для двух дат одного и того же невисокосного года, причем вторая дата не раньше первой. Программа должна выводить одно число — на сколько дней вторая дата позже первой. Например, для входных данных 1 3 8 3 вывести 7, для входных данных 1 1 31 12 вывести 364.
- 4.2. Напишите программу, которая вводит с клавиатуры три целых числа d, m, k — число и месяц, а также количество дней. Программа должна выводить два числа — число и месяц, наступающий через k дней после даты d/m . Гарантируется, что год невисокосный, и вторая дата в том же году, что и первая. Например, для входных данных 1 9 45 вывести 16 10, так как через 45 дней после 1 сентября наступает 16 октября.
- 4.3. Напишите программу, которая вводит с клавиатуры три целых числа d, m, k — число и месяц, а также количество дней. Программа должна выводить два числа — число и месяц бывший k дней назад относительно даты d/m . Гарантируется, что вторая дата в том же году, что и первая. Например, для входных данных 1 9 92 вывести 31 5, так как за 92 дня до 1 сентября было 31 мая.
- 4.4. Напишите программу, которая вводит через пробел целые числа K, s , и вещественное число p . K является первоначальным размером кредита, выданного под p % годовых. Кредит погашается равными платежами по s рублей каждый месяц. Из этой суммы сначала уплачивается проценты за пользование кредитом в размере $\frac{p}{12}$ процентов от текущей суммы долга, а затем оставшаяся часть суммы направляется на уменьшение суммы долга. Программа должна выводить по одному в строке оставшуюся сумму долга на конец каждого месяца до полного погашения кредита. Последним числом в выводе должен быть ноль. Округлять при

выводе до 2 знаков, при расчетах не округлять. Например, при сумме долга $K = 100000$, сумме ежемесячного платежа 10000 и ставке $p = 12\%$ должны быть распечатаны числа

91000.00

81910.00

72729.10

63456.39

54090.95

44631.86

35078.18

25428.96

15683.25

5840.09

0.00

- 4.5. Напишите программу, которая вводит с клавиатуры номер n дня в году (от 1 до 365) и выводит день и месяц в виде двух чисел d и m (например, для $n = 33$ вывести 3 2, так как 34-й день года — 3 февраля).

-
- 5.1. Выведите по одному в строке все трехзначные простые числа.
- 5.2. Введите с клавиатуры число $N > 5$. Натуральные числа a, b, c называются числами Пифагора, или пифагоровой тройкой, если выполняется условие $a^2 + b^2 = c^2$. Напечатать все пифагоровы тройки, в которых все числа меньше N . Каждую в отдельной строке, числа в каждой строке в возрастающем порядке через пробел, строки в возрастающем лексикографическом порядке (то есть возрастания первого, затем второго, затем третьего числа). Например, для $N = 6$ напечатать 3 4 5.
- 5.3. Даны через пробел натуральные числа n, m . Напечатать по одному в строке в возрастающем порядке все натуральные числа, меньшие n , квадрат суммы цифр которых равен m . Если таких нет, напечатать -1. Например для входных данных 100 4 напечатать числа 2, 11, 20, каждое в отдельной строке.
- 5.4. Последовательность Хэмминга образуют натуральные числа, не имеющие других простых делителей, кроме 2, 3 и 5. Введите с клавиатуры число N и выведите сумму первых N элементов последовательности Хэмминга. Например, для $N = 8$ вывести 38 (так как $1 + 2 + 3 + 4 + 5 + 6 + 8 + 9 = 38$).