

Практическое занятие «Ассемблер–9»

28 апреля 2020 года

Для ввода-вывода используйте стандартные функции `printf`, `scanf`, `gets` из библиотеки языка Си.

1. Напишите программу, которая считывает с клавиатуры два целых 32-битных числа со знаком, находит их НОД и выводит его на экран.
2. Напишите программу, которая считывает с клавиатуры массив 16-битных чисел со знаком (то есть, считывает количество элементов, а потом сами элементы), а затем выводит его на экран, для каждого числа выводя дополнительно разницу между максимальным элементом массива и этим числом. Достаточно большой буфер под массив отводится статически в сегменте данных.
3. Считайте с клавиатуры z -строку. Создайте две строки, состоящие, соответственно, из символов исходной строки, стоящих на четных и нечетных местах. Выведите полученные строки на экран. Достаточно большие буферы под строки отведите статически в сегменте данных.
4. **Шифр Цезаря.** Считайте с клавиатуры 16-битное беззнаковое число и z -строку, состоящую только из заглавных символов латиницы и пробела (всего используется не более 27 различных символов), содержащую не более 100 символов. Затем спросите у пользователя, хочет ли он зашифровать или расшифровать строку (например, введя с клавиатуры 0 или 1), после чего зашифруйте/расшифруйте строку шифром Цезаря с указанным смещением и результат выведите на экран.

Напомним, что шифр Цезаря заключается в следующем. Чтобы зашифровать символ в строке, содержащей алфавит сообщения (в рамках условия задачи — в строке `ABCDEFGHIJKLMNOPQRSTUVWXYZ_` с последним символом пробела), выбирается символ, отстоящий *циклически* вправо от шифруемого на заданный сдвиг. При дешифровании выбирается символ, отстоящий от шифруемого влево на заданный сдвиг *циклически*. Цикличность означает, что мы переходим на начало алфавита, если при шифровании вышли за его конец, и переходим на конец, если при дешифровании вышли за начало. Например, если сдвиг равен 3, то соответствие символов при шифровании (\rightarrow) и дешифровании (\leftarrow) описывается как $A \leftrightarrow D, B \leftrightarrow E, C \leftrightarrow F, \dots, W \leftrightarrow Z, X \leftrightarrow _, Y \leftrightarrow A, Z \leftrightarrow B, _ \leftrightarrow C$.

- 5*. Напишите программу, считывающую с клавиатуры неотрицательное целое число n , не превосходящее 20, и *красиво* выводящую строки треугольника Паскаля с нулевой по n -ю. Строка с номером k содержит $k + 1$ натуральное число; крайние — единицы, каждое внутреннее есть сумма двух чисел предыдущей строки: стоящего над ним и левее него. Нулевая строка состоит только из единицы. Первые несколько строк треугольника Паскаля:

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
```

Для «красивого» вывода разумно использовать варианты форматтеров `%<число>X`, где `число` — ширина поля, в котором выводится объект (дополняясь слева пробелами при необходимости). Например, при выводе числа 123 форматтером `%5d`, на экране будем иметь `_123` (подчёркивания означают пробелы).