

Тема 6 а)

Контрольные задания

Указание. Для оформления заданий с операторами программирования можно использовать процедуры.

1. Реализовать условный оператор **if...then...else**: если x не является простым числом, то вывести само число и разложение его на простые сомножители, иначе вывести сообщение о том, что x является простым числом (с подстановкой значения x). Для проверки рассмотреть несколько чисел.

Указание. Для определения, является ли число x простым, использовать функцию **isprime(x)**. Для разложения числа x на простые сомножители использовать функцию **ifactor(x)**. Для вывода сообщения с подстановкой значения использовать команду **printf("%a является простым числом.", x)** ;

2. Дано число x . Реализовать условный оператор **if...then...else**: если x является полным квадратом, то вывести сообщение об этом и значение числа, полным квадратом которого оно является, в противном случае вывести сообщение о том, что x не является полным квадратом (с подстановкой значения x). Для проверки рассмотреть несколько чисел.

Указание. Для определения, является ли число x полным квадратом, использовать функцию **issqr(x)**. Для вывода сообщения с подстановкой использовать команду **printf**.

3. Реализовать условный оператор **if...then...elif...else**: если x не является натуральным числом, то вывести сообщение об этом, иначе проверить количество цифр в числе. Для однозначного числа вывести сообщение о том, что « x является однозначным числом», для двузначного числа вывести сообщение о том, что « x является двузначным числом», в остальных случаях вывести сообщение о том, что « x является многозначным числом, количество цифр в котором больше двух». Для проверки рассмотреть несколько чисел. В сообщениях использовать подстановку значения x .

Указание. Для определения, является ли число x натуральным, можно использовать функцию **type(x, posint)** .

4. Реализовать оператор цикла **while**: вывести на экран все простые числа в диапазоне от 1 до 100 в виде последовательности. Формирование последовательности организовать в цикле. Сделать то же самое с помощью оператора **for/from**.
5. Реализовать оператор цикла **while**: вывести на экран список всех полных квадратов в диапазоне от 1 до 100. Формирование списка организовать в цикле. Сделать то же самое с помощью оператора **for/from**.

6. Реализовать оператор цикла с параметром **for/in**: вычислить произведение тех чисел в диапазоне от 1 до 1000, которые являются полными кубами. Перед работой цикла сформировать множество таких чисел (полных кубов от 1 до 1000) с помощью оператора \$.
7. Реализовать оператор цикла с параметром **for/in**: перевести список значений в градусах [0, 45, 90, 180, 360] в радианы. Полученные значения также вывести в виде списка. Организовать наращивание списка в цикле.
8. Реализовать оператор цикла **for/from...while**: вычислить сумму квадратов первых пятнадцати простых чисел. Вывести полученную сумму и значение счетчика цикла.
9. Реализовать оператор цикла **for/in...while**: вычислить сумму тех чисел в диапазоне от 1 до 200, которые являются одновременно полными квадратами и полными кубами. Перед работой цикла сформировать только множество полных кубов (с помощью оператора \$). Проверку на полный квадрат организовать в теле цикла с помощью условного оператора и функции **issqr**.
10. Вывести буквы русского алфавита от А до К двумя способами: с помощью оператора формирования последовательности и с помощью команды создания последовательности **seq**
11. Использовать команду **seq(. . . in . .)** для решения примера №7: перевести список значений в градусах [0, 45, 90, 180, 360] в радианы.
12. Вычислить приближенные значения суммы и произведения степеней экспонент от 1 до 10, используя команды **add** и **mul**.
13. Вывести список тех чисел от 1 до 100, которые являются полными квадратами.
Указание. Использовать команду **select** и команду **issqr** для определения того, является ли число полным квадратом.
14. Вывести основу каждого слова строки “Maple is my favorite mathematical software”, используя команду **map**. Для работы со строками использовать команды **Split** и **Stem** из пакета **StringTools**.