

Задачник по программированию на языке Оберон

Часть 1. Консольный ввод-вывод, IF, ABS, DIV, MOD,
равенство вещественных чисел.

Задача №1

Вводятся два целых числа. Определить наименьшее и
наибольшее из них.

Пример:

A = 51

B = 17

Min: 17

Max: 51

Задача №2

Вводятся три целых числа. Определить наименьшее и
наибольшее из них.

Пример:

A = 22

B = 26

C = 5

Min: 5

Max: 26

Задача №3

Вводятся три целых числа. Определить количество положительных
чисел.

Пример:

A = 4

B = -6

C = 5

Positive count: 2

Задача №4

Вводятся три целых числа. Определить количество положительных и количество отрицательных чисел.

Пример:

$$A = -1$$

$$B = 0$$

$$C = 2$$

Positive count: 1

Negative count: 1

Задача №5

Вводятся три целых числа. Определить среднее среди них, то есть число, расположенное между наименьшим и наибольшим.

Пример:

$$A = 8$$

$$B = 9$$

$$C = 2$$

Median: 8

Задача №6

Вводятся три целых числа: A, B, C. Если их значения упорядочены по возрастанию, то удвоить их; в противном случае заменить значение каждой переменной на противоположное. Вывести новые значения переменных.

Пример:

$$A = -2$$

$$B = 6$$

$$C = 10$$

New values:

$$A = -4$$

$$B = 12$$

$$C = 20$$

Пример:

$$A = 72$$

$$B = -81$$

$$C = 12$$

New values:

$$A = -72$$

$$B = 81$$

$$C = -12$$

Задача №7

Вводятся три целых числа: A , B , C . Определить, какое из двух чисел ближе к A : B или C . Вывести это число и расстояние от этого числа до числа A .

Пример:

$$A = 7$$

$$B = 2$$

$$C = 33$$

Closest to A : 2

Distance: 5

Задача №8

Даны три числа. Найти сумму двух наибольших из них.

Пример:

$$A = 7$$

$$B = 2$$

$$C = 33$$

Sum: 40

Задача №9

Вводятся шесть целых чисел — координаты трёх вершин прямоугольника, стороны которого параллельны координатным осям. Найти координаты его четвертой вершины.

Пример:

$$x_1 = 2 \quad y_1 = 2$$

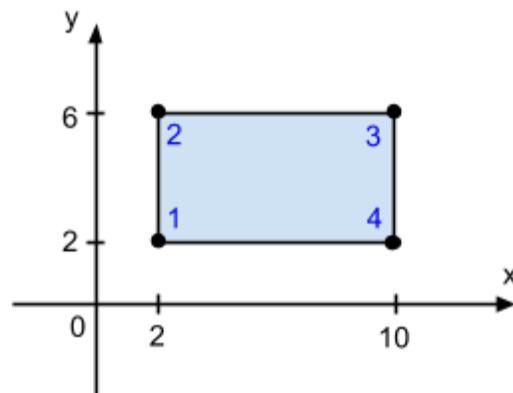
$$x_2 = 2 \quad y_2 = 6$$

$$x_3 = 10 \quad y_3 = 6$$

Fourth vertex:

$$x_4 = 10 \quad y_4 = 2$$

Примечание: Точки могут вводиться в любом порядке, а не обязательно в указанном на рисунке.



Задача №10

Даны целочисленные координаты точки на плоскости. Если точка совпадает с началом координат, то вывести 0. Если точка не совпадает с началом координат, но лежит на оси OX или OY, то вывести соответственно 1 или 2. Если точка не лежит на координатных осях, то вывести 3.

Примеры:

x = 6

y = -15

Answer: 3

x = 0

y = 7

Answer: 2

Задача №11

Дан номер года. Определить количество дней в этом году, учитывая, что обычный год насчитывает 365 дней, а високосный — 366 дней. Високосным считается год, делящийся на 4, за исключением тех годов, которые делятся на 100, но, при этом, не делятся на 400.

Пример:

Year = 1904

Days: 366

Year = 1917

Days: 365

Year = 2000

Days: 366

Year = 1900

Days: 365

Задача №12

Даны три числа. Определить, возможен ли треугольник, длины сторон которого соответствуют этим трём числам.

Примеры:

$$A = 3$$

$$B = 6$$

$$C = 5$$

Triangle exists.

$$A = 5$$

$$B = 6$$

$$C = 11$$

Triangle does not exist.

Задача №13

Дано натуральное трёхзначное число. Найти сумму и произведение цифр этого числа.

Пример:

$$N = 134$$

Sum: 8

Product: 12

$$N = 980$$

Sum: 17

Product: 0

Задача №14

Дано натуральное четырёхзначное число. Вывести реверсную запись четырёхзначного числа.

Пример:

$$n = 9842$$

Reverse: 2489

$$n = 1230$$

Reverse: 321

Задача №15

Дано натуральное пятизначное число. Определить, является ли оно палиндромом.

Пример:

n = 24542

Palindrome

n = 98412

Not palindrome

Задача №16

Написать программу, которая решает квадратное уравнение путём вычисления дискриминанта.

Пример:

a = 2

b = 8

c = 3

$D = b^2 - 4ac = 64 - 24 = 40$

Square root of D = 6.32

$x_1 = -3.58$ $x_2 = -0.42$

Примечание:

Пишите программу поэтапно, добавляя необходимые проверки и улучшения в следующем порядке:

- 1) Добейтесь того, чтобы программа решала простейшие квадратные уравнения с двумя корнями, как в примере
- 2) Убедитесь, что числа выводятся на экран в адекватной форме (например, с двумя знаками после запятой)
- 3) Добавьте проверку на положительность дискриминанта. Если дискриминант отрицательный, выводите сообщение о том, что действительных корней нет.
- 4) Добавьте проверку на равенство дискриминанта нулю. Если дискриминант равен нулю, выводите только один корень.

- 5) Сделайте, чтобы программа выводила строчку расчёта дискриминанта так, как это показано в примере выше.
- 6) При отрицательном произведении $4ac$, в строчке расчёта дискриминанта будет выведен двойной минус. Например: «64 - -24». Улучшите программу так, чтобы в таком случае выводился знак плюс.
- 7) В случае $a = 0$ квадратное уравнение вырождается до линейного. Добавьте соответствующую проверку и решайте это уравнение как линейное.
- 8) В случае $a = b = 0$ уравнение вырождается до вида « $c = 0$ » и больше не зависит от x . Если c действительно равен 0, то правильный ответ: « x — любое число», в противном случае, правильный ответ: «нет решения».

Задача №17

Дано натуральное трёхзначное число. Вывести на экран все трёхзначные числа, которые можно получить из цифр данного числа путём их перестановки.

Примеры:

$N = 242$

242 224 422

$N = 157$

157 175 517 571 715 751

$N = 680$

680 860 608 806

Задача №18

Даны координаты точки и радиус круга с центром в начале координат. Определить, принадлежит ли данная точка кругу. Оформить решение в виде функциональной процедуры.

Задача №19

Даны координаты центра круга и его радиус, а также координаты точки. Определить, принадлежит ли данная точка кругу. Оптимизируйте программу, заменив извлечение квадратного корня на возведение в квадрат, а также добавив предварительную проверку.

Задача №20

Вычисление площади и периметра фигур. В начале программа спрашивает у пользователя, с какой фигурой он хочет иметь дело: с квадратом, прямоугольником, кругом или треугольником. Затем, программа позволяет ввести параметры этих фигур:

для квадрата — длину стороны,

для прямоугольника — длину и ширину,

для круга — его радиус.

В случае с треугольником, программа задаёт дополнительный вопрос — хочет ли человек задать его через длины сторон (тогда вводятся 3 числа) или через координаты вершин (6 чисел).

Затем, программа выводит на экран значения площади и периметра заданной фигуры (в случае с кругом вместо периметра выводится длина окружности).

Примеры:

Choose figure

[1 - Square, 2 - Rectangle, 3 - Circle, 4 - Triangle]: 3

Radius = 7.36

Circumference: 46.24

Area: 170.18

Choose figure

[1 - Square, 2 - Rectangle, 3 - Circle, 4 - Triangle]: 4

Choose input method [1 - Sides, 2 - Vertices]: 2

x1 = 1.5

y1 = 2

x2 = 4.5

y2 = 2

x3 = 1.5

y3 = 3

Perimeter: 8.61

Area: 3

Задача №21

Вводится значение x . Программа вычисляет значение одной и той же функции $f(x)$ для данной x по данной формуле:

$$f(x) = \begin{cases} / f(x) = x - 2, & \text{если } x > 0, \\ f(x) = 0, & \text{если } x = 0, \\ \backslash f(x) = |x|, & \text{если } x < 0. \end{cases}$$

Задача №22.

Дано: ширина и длина прямоугольного отверстия; ширина, длина и высота кирпича. Определить, может ли кирпич пройти сквозь отверстие.

Задача №23.

Найти медиану пяти вводимых чисел. Медиана - число, стоящее посередине отсортированного ряда чисел.

Пример:

A = 5

B = 23

C = 2

D = 7

E = 21

Mediana: 7

Часть 2. Циклы WHILE, REPEAT UNTIL. Вложенные циклы.

Задача №1.

Написать процедуру, подсчитывающую сумму цифр данного натурального числа.

Пример:

N = 12345

Sum: 15

Задача №2.

Написать процедуру, подсчитывающую количество цифр данного натурального числа.

Пример:

N = 29787

Digit count: 5

Задача №3.

Написать процедуру, возвращающую перевёрнутое натуральное число.

Пример:

N = 2149

Reverse: 9412

Задача №4.

Написать программу, выводящую через запятую все четырёхзначные числа, для которых одновременно выполняются все следующие условия:

- а) число делится на 7,
- б) сумма цифр числа не меньше 12 и не больше 20,
- в) среди цифр числа есть хотя бы одна чётная цифра,
- г) в числе нет ни одной цифры 3.

Задача №5.

Написать программу, выводящую все простые числа, не превосходящие 200. Простое число — это число, делящееся только на себя и на единицу.

Задача №6.

Найти все числа Фибоначчи, не превосходящие 30000. Первые два числа Фиббоначи - это единицы. Каждое следующее число представляет собой сумму двух предыдущих. Ряд чисел Фиббоначи: 1 1 2 3 5 8 13 21 34...

Задача №7.

Найти предел суммы ряда:

$$1/(1*2*3) + 1/(2*3*4) + 1/(3*4*5) + \dots$$

Задача №8.

Человек загадывает целое число от 1 до 100. Программа задаёт вопросы вида: «Загаданное число больше 50? [д/н]» и определяет загаданное число методом половинного деления за 7 попыток.

Задача №9.

Игра. В начале с клавиатуры вводится количество спичек. Затем, два игрока по очереди забирают по 1, 2 или 3 спички. Кто взял последнюю - проиграл.

Задача №10.

Перевернуть данное натуральное число задом наперёд.

Задача №11.

Написать процедуру Power2, которая отвечает на вопрос: «Какая это степень двойки?» для данного числа n. Если n - не степень двойки, процедура должна возвращать -1.

PROCEDURE Power2(n: INTEGER): INTEGER;

Примеры:

N = 16

Answer: 4

N = 128

Answer: 7

N = 1

Answer: 0

N = 199

Answer: -1

Задача №12.

Найти первую цифру данного натурального числа.

Примеры:

N = 7851

Answer: 7

N = 265

Answer: 2

Задача №13.

Установить, все ли цифры в данном натуральном числе имеют одинаковую чётность.

Примеры:

N = 123

Answer: No

N = 173

Answer: Yes

N = 8240

Answer: Yes

Задача №14.

Определить, является ли данное число кубом целого числа.

Вычисления оформить как процедуру следующего вида:

```
PROCEDURE Root3(x: INTEGER; VAR root: INTEGER): BOOLEAN;
```

Процедура записывает найденный корень третьей степени в вар-параметр root и возвращает Истину, в противном случае возвращает Ложь.

Примеры:

N = 64

Answer: 4

N = -125

Answer: -5

N = 777

Answer: Not a cube

Задача №15.

Найти первую тысячу пар простых чисел, отличающихся на 2.

Задача №16.

В ряд без пробелов выписаны все натуральные числа (в порядке возрастания): 123456789101112...

Определить, какая цифра стоит на N'ом месте.

Примеры:

N = 3

Answer: 3

N = 11

Answer: 0

Задача №17.

В ряд без пробелов выписаны квадраты всех чисел (в порядке возрастания): 149162536496481...

Определить, какая цифра стоит на N-ом месте.

Пример:

N = 11

Answer: 9

Задача №18.

Написать программу, рисующую следующие узоры. Процедура рисования каждого узора принимает параметр N - размер узора.

1	1	1	xxxx	1111	xxxxxx	x	x x x
22	22	222	xxxx	222	x++++x	x x	x x
333	333	33333	xxxx	33	x++++x	x x	x x x
4444	22	222	xxxx	4	xxxxxx	x x	x x
55555	1	1				x	x x x
N=5	N=3	N=3	N=4	N=4	N=4	N=3	N=5

Задача №19.

Некоторые точки плоскости с целыми значениями координат закрашены (см. рис. 2). Написать процедуру, которая возвращает Истину для тех пар координат (x; y), которые закрашены и Ложь - для остальных.

PROCEDURE f(x, y: INTEGER): BOOLEAN;

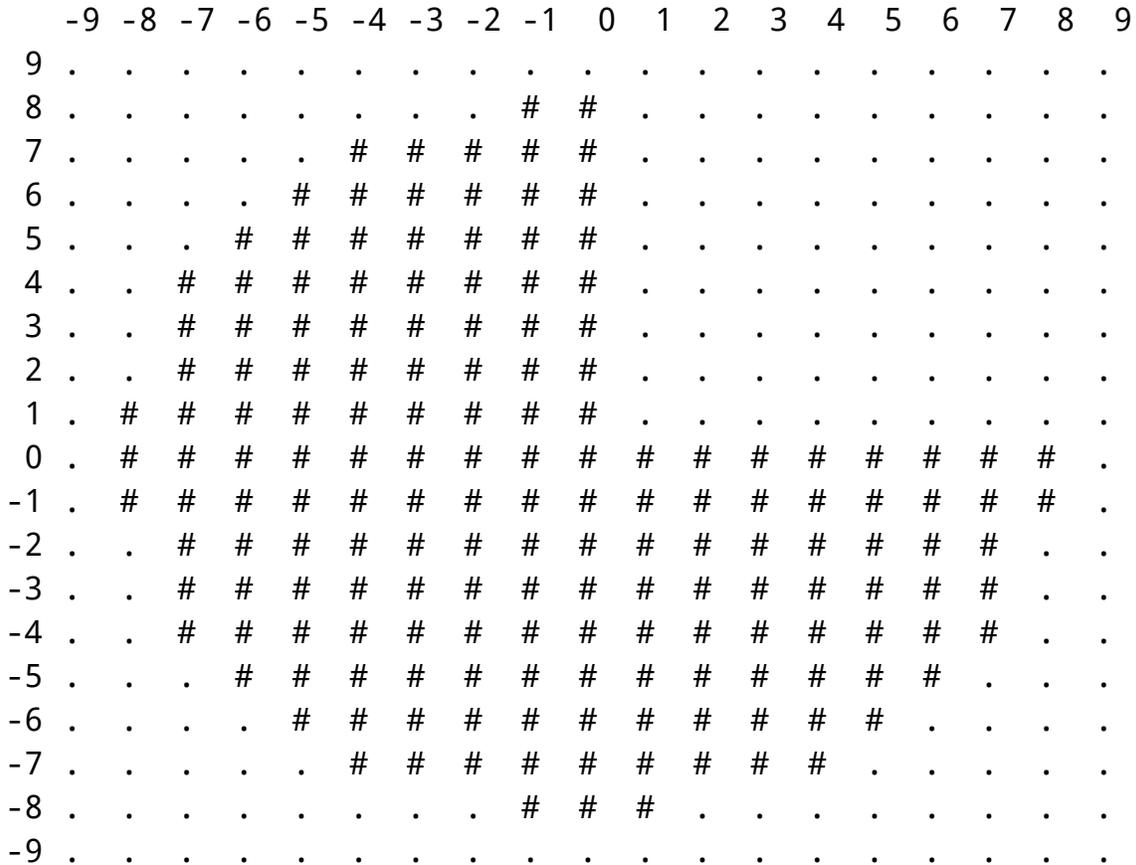


рис. 2

Закрашенные точки помечены символом решётки (#).

Задача №20.

Написать функциональную процедуру, принимающую три числа: a, b и step и возвращающую произведение чисел от a до b включительно с шагом step.

PROCEDURE f(a, b, step: INTEGER): LONGINT;

Например, $f(5, 13, 3) = 5 \cdot 8 \cdot 11 = 440$.

Часть 3. Одномерные массивы. Обработка массивов на месте.

Задача №1.

Вводится массив натуральных чисел, но первое число может быть равно нулю. Вывести последовательность нулей и единиц таким образом, чтобы каждое число данного массива соответствовало количеству подряд идущих нулей или единиц. Первое число в массиве соответствует длине первой последовательности нулей.

Пример:

Source: 3 2 5 1 2

Result: 0001100000100

Source: 0 2 3 7 2

Result: 11000111111100

Задача №2.

Перевернуть одномерный массив на месте, не используя вспомогательный массив.

Пример:

Source: 1 9 2 8 3 5

Result: 5 3 8 2 9 1

Задача №3.

Дан массив из N натуральных чисел и число M . N делится нацело на M . Из каждой группы по M подряд идущих элементов массива выбрать наибольший и вычислить сумму выбранных элементов.

Пример:

$N = 15$

$M = 5$

Source: 4 8 2 7 9 **2 1 5 3 4** 1 7 2 5 1

Maximums: 9 5 7

Sum: 21

Задача №4.

Дан массив из N натуральных чисел и число M. N делится нацело на M. В каждой группе по M подряд идущих элементов массива найти медиану и вычислить сумму этих медиан. Для поиска медианы не заводить второй массив. Использовать разбиение задачи на процедуры.

Пример:

N = 15

M = 5

Source: 4 8 2 7 9 **2 1 5 3 4** 1 7 2 5 1

Maximums: 7 3 2

Sum: 12

Задача №5.

Каждый элемент массива от второго до предпоследнего заменить на среднее арифметическое трёх чисел: его самого, соседа слева и соседа справа.

Пример:

Source: 4 9 2 7 3 1

Result: 4 5 6 4 3.67 1

Задача №6.

Каждый элемент массива от третьего до предпредпоследнего заменить на медиану пяти чисел: его самого, двух соседей слева и двух соседей справа.

Пример:

Source: 3 2 2 8 7 9 5 3 6

Result: 3 2 3 7 7 7 6 3 6

Задача №7.

Определить количество элементов массива, обладающих следующим свойством: величина всех элементов массива, стоящих слева от него, меньше его величины, а величина всех элементов массива, стоящих справа от него, больше величины этого элемента.

Пример:

Source: 2 **2** 7 5 4 4 3 **9**

Count: 2

Задача №8.

Переставить элементы одномерного массива таким образом, чтобы сначала шли все отрицательные элементы, затем все нули, а затем все положительные элементы, не используя вспомогательный массив. Итоговый порядок положительных элементов, как и отрицательных, не имеет значения.

Пример:

Source: 8 2 0 5 -1 2 9 -5 0

Result: -1 -5 0 0 9 2 5 8 2

Примечание: Сортировка всего массива для решения данной задачи является избыточной, данная задача должна быть решена гораздо эффективнее.

Задача №9.

Отсортировать данный одномерный массив чисел так, чтобы элементы в нём шли в неубывающем порядке. Использовать метод пузырьковой сортировки. Для этого в массиве слева направо перебираются пары подряд идущих элементов: сначала первый и второй, затем второй и третий, затем третий и четвёртый и так далее. В каждой такой паре элементы меняются местами, если они идут в неправильном порядке. В результате такого перебора-перестановки в конец массива гарантированно попадает наибольший элемент. После того, как перебор завершён, он повторяется вновь, но заканчивается уже не в конце массива, а на один элемент раньше. В результате

этого на предпоследнее место становится наибольший элемент из оставшихся. Процесс повторяется до тех пор, пока он имеет смысл. Примечание: Длина массива задаётся константой. Значения элементов массива задаются случайным образом. Массив выводится на экран до и после сортировки.

Задача №10.

Отсортировать данный одномерный массив чисел так, чтобы элементы в нём шли в неубывающем порядке. Использовать метод сортировки выбором. Алгоритм сортировки выбором перебирает все элементы массива от первого до предпоследнего и для каждого такого элемента определяет, какое значение должно стоять на данном месте. Для того, чтобы определить это, необходимо просмотреть все элементы от следующего за данным до последнего и найти среди них наименьший, а затем найденный элемент поменять местами с данным. Для поиска наименьшего элемента используйте цикл WHILE. Примечание: Длина массива задаётся константой. Значения элементов массива задаются случайным образом. Массив выводится на экран до и после сортировки.

Задача №11.

Дано натуральное число N . Перевести число N в двоичную систему счисления методом вычёрпывания. Для этого, сначала определяется наибольшая степень двойки, помещающаяся в число N . Обозначим её буквой x . Например, для числа $N = 75$ такая степень двойки равна 64. Затем, в цикле происходит следующее. Если из N можно вычесть x так, чтобы N осталось неотрицательным, то из N вычитается x , а к получившемуся двоичному числу приписывается единица (происходит вычёрпывание степени двойки из N). В противном случае к получившемуся двоичному числу приписывается ноль. Затем, x делится на 2 (то есть степень двойки понижается). Процесс повторяется до тех пор, пока x не станет равно нулю.

Пример:

$N = 147$

Answer: 10010011

Примечание: число N имеет тип LONGINT, а получившееся двоичное число должно быть массивом достаточной длины, например ARRAY 32 OF CHAR. Длина используемого участка массива должна быть записана в целочисленную переменную len.

Задача №12.

Дано натуральное число N. Перевести число N в двоичную систему счисления методом деления пополам. Число N несколько раз делится на 2 (при помощи целочисленного деления), пока оно не уменьшится до нуля. Всякий раз остаток от деления (а это либо 0, либо 1) приписывается к получившемуся двоичному числу. В конце, получившуюся двоичную запись числа необходимо перевернуть задом наперёд.

Пример:

N = 19

Answer: 10011

Задача №13.

Задача №14.

Задача №15.

Задача №16.

Заменить во введенной строке все кратные вхождения символов одинарными символами.

Пример:

String: Hello Pool AAA B CCD----C

Answer: Helo Pol A B CD-C

Задача №17

Вводится строка (один раз в начале), затем пользователь вводит два числа. Программа меняет местами буквы с индексами, соответствующими этим двум числам. Строка отображается на экране, после чего пользователь опять вводит два числа. Процесс повторяется до тех пор, пока одно из двух введенных чисел не станет меньше нуля. Символы в строке нумеруются с нуля.

Пример:

String: HELLO WORLD

a = 1

b = 7

HOLLO WERLD

a = 1

b = 0

OHLLLO WERLD

a = -1

Задача №18.

Переставить символы введенной строки следующим образом: сначала последний символ, затем первый, затем предпоследний, за ним второй символ, и т. д., пока символы не кончатся. Результат вывести на экран и повторить с ним ту же операцию. Продолжать так до тех пор, пока не получится исходная строка.

Варианты преобразований: сначала все нечетные символы справа налево, затем все четные слева направо; сначала 1-й, 4-й, 7-й и т.д. символы, затем 2-й, 5-й, 8-й и т.д., затем 3-й, 6-й, 9-й и т.д.; переставить первую букву в конец слова и т.д.

Пример:

String: ABCDE

EADBC SEBAD DCAEB BDECA ABCDE

Задача №19.

Написать функциональную процедуру, которая для данной строки определяет, начинаются ли все её слова с одной и той же буквы.

PROCEDURE SameStart(s: ARRAY OF CHAR): BOOLEAN;

Например, SameStart('Krot kilut kidal kapustu') = TRUE.

Задача №20.

Убрать из введенной строки 1-й, 3-й, 5-й и т. д. символы. Не заводить второй массив.

Пример:

String: LENINGRAD

Answer: EIGA

Задача №21.

Дан случайный числовой массив m длины N . Необходимо определённым образом заполнить целыми числами второй массив x такой же длины, а также записать определённое значение в целочисленную переменную $start$. Значение, записанное в переменной $start$, должно равняться индексу наименьшего элемента в первом массиве. Во втором массиве «под этим элементом» (то есть по индексу $start$) должен находиться индекс следующего в порядке неубывания элемента. Под ним — индекс следующего и так далее. Под «последним» (наибольшим) элементом первого массива должно стоять число -1 .

Пример:

m : 51 32 94 20 79 10 42 79 67

x : 8 6 -1 1 7 3 0 2 4

i : 0 1 2 3 4 5 6 7 8 ← индексы выведены для

наглядности

$start = 5$

Задача №22.

Дан одномерный массив длины N с целыми числами от 0 до 15. Необходимо вывести на экран (в виде текста) следующую картинку, соответствующую данному массиву (в примере N = 5):

```
[_]
[ ] [ ]
[ ] [ ]
[ ] [ ]
[ ] [ ] [ ]
[ ] [ ] [ ] [ ]
[ ] [ ] [ ] [ ]
  7   3   6   0   2
```

Задача №23.

Заполнить двумерный массив числами из треугольника Паскаля. Верхний столбец и самая левая строка массива заполняются единицами, а каждый из оставшихся элементов заполняется числом, равным сумме двух чисел: стоящих непосредственно над ним и непосредственно слева от него:

```
1   1   1   1   1   1   1
1   2   3   4   5   6   7
1   3   6  10  15  21  28
1   4  10  20  35  56  84
1   5  15  35  70 126 210
1   6  21  56 126 252 462
1   7  28  84 210 462 924
```

Задача №24.

Построить и визуализировать маршрут движения коня на шахматной доске. Конь должен обойти всю доску, посетив каждое поле ровно один раз. Строить маршрут, используя правило Варнсдорфа: при обходе доски коня следует ставить всякий раз на то поле, из которого

он может сделать наименьшее количество ходов; если таких полей несколько, то случайным образом выбирается любое из них.

Итерационное решение будет более сложным, но и более оптимальным. Можно использовать два указателя — $p1$ и $p2$. Сначала оба указателя указывают на начало списка. Затем перемещаем $p2$ на k узлов вперед. Теперь мы начинаем перемещать оба указателя одновременно. Когда $p2$ дойдет до конца списка, $p1$ будет указывать на нужный нам элемент.