



Сборник задач по Turbo Pascal



ГБОУ школа № 594

Санкт-Петербург

Вывод текста. Модуль Crt. Цветной текст

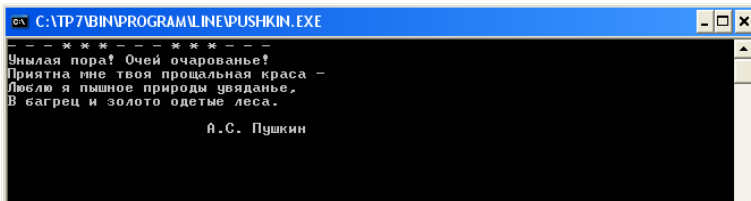
5.1) Напишите программу, которая выводит на стандартном экране декоративную строчку и четверостишие:

```
___**_*___**_*___**_*___**_*___
```

Унылая пора! Очей очарованье!
 Приятна мне твоя прощальная краса –
 Люблю я пышное природы увяданье,
 В багрец и золото одетые леса.

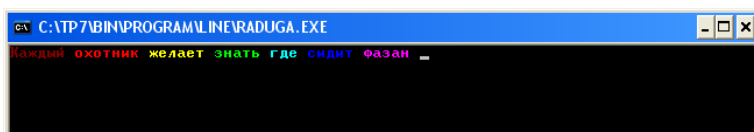
(пропустите пустую строчку)

А. С. Пушкин



5.2) Напишите программу, которая выводит в одной строке **на черном фоне** фразу:

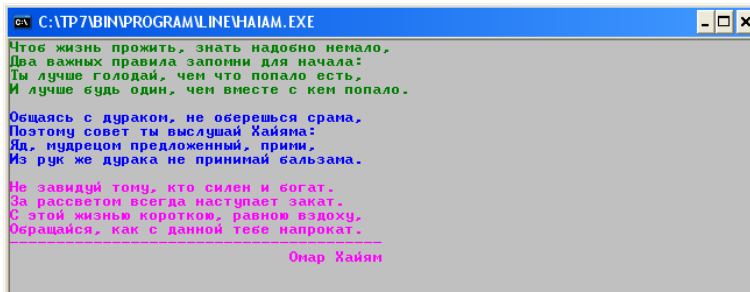
Каждый охотник желает знать, где сидит фазан (каждое слово должно быть написано наиболее подходящим цветом шрифтом повышенной яркости).



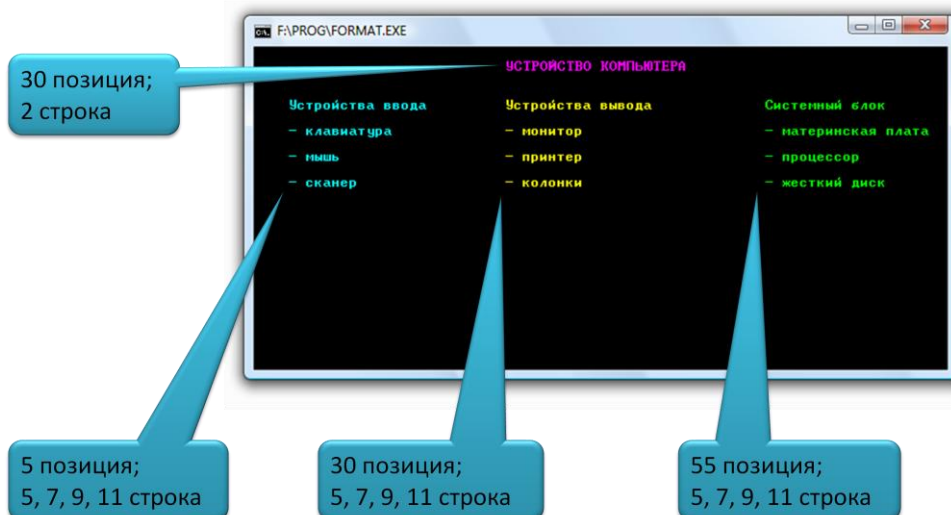
5.3) Напишите программу, которая выводит **на сером фоне** разноцветные четверостишия:

Чтоб мудро жизнь прожить, знать надобно немало,
 Два важных правила запомни для начала:
 Ты лучше голодай, чем что попало есть,
 И лучше будь один, чем вместе с кем попало.
 Общаясь с дураком, не оберёшься срама,
 Поэтому совет ты выслушай Хайяма:
 Яд, мудрецом тебе предложенный, прими,
 Из рук же дурака не принимай бальзама.
 Не завидуй тому, кто силен и богат.
 За рассветом всегда наступает закат.
 С этой жизнью короткою, равною вздоху,
 Обращайся, как с данной тебе напрокат.

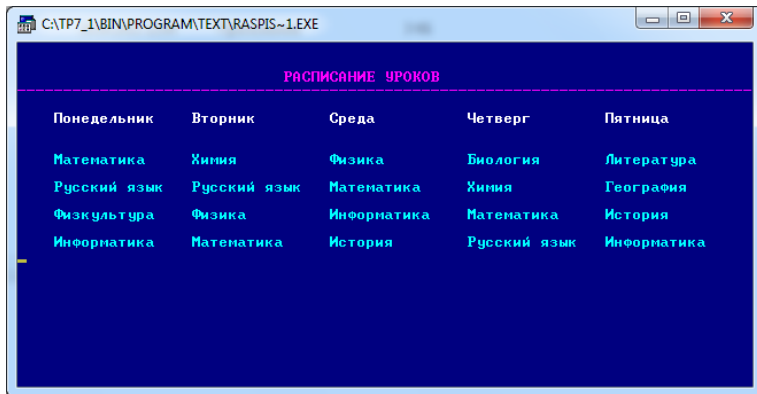
Омар Хайям



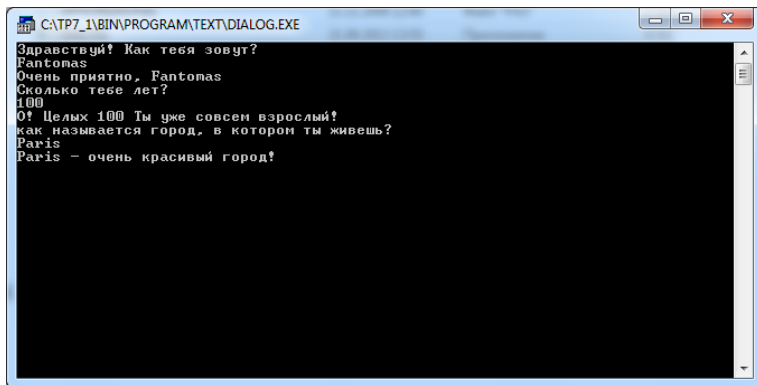
5.4) Напишите программу, которая выводит текст в таком виде (цвета на ваше усмотрение)



5.5) Напишите программу, которая выводит на экран расписание уроков. Цвет фона и шрифта выберите самостоятельно. Нужные позиции для текста рассчитайте самостоятельно, учитывая, что на экране помещается 80 символов.

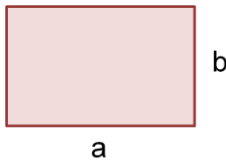


5.6) Напишите программу, которая будет осуществлять диалог с компьютером. Используйте 2 строковые переменные (для имени и названия города) и 1 переменную целого типа (для возраста).



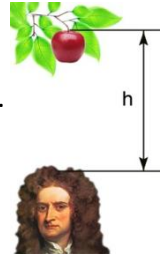
Линейные программы

- 6.1) Напишите программу вычисления площади и периметра прямоугольника (пользователь вводит стороны прямоугольника **a** и **b** – целые числа).



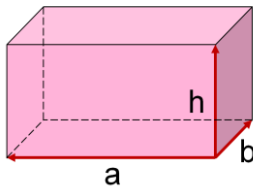
Площадь: $S = a * b$
 Периметр: $P = 2 * (a + b)$

- 6.2) Напишите программу, чтобы определить, сколько времени будет падать с яблони на голову Ньютону яблоко. Высота ветки вводится пользователем с клавиатуры. Ускорение свободного падения $g = 9,8 \text{ м/с}^2$ укажите в разделе описаний как константу. Выведите ответ с двумя знаками после запятой, например, в формате 6:2

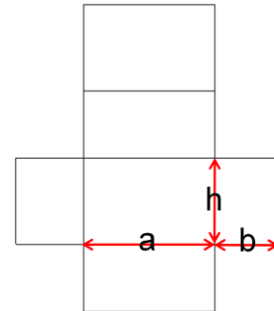


Формула: $t = \sqrt{\frac{2h}{g}}$

- 6.3) Напишите программу вычисления площади поверхности параллелепипеда (площадь развертки), пользователь вводит размеры параллелепипеда **a**, **b** и **h** – вещественные числа. Формулу напишите самостоятельно.



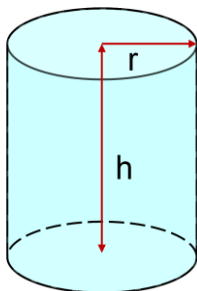
Развертка параллелепипеда:



```

C:\TP7\BIN\PROGRAM\LINE\KUB.EXE
Введите длину параллелепипеда a
9.2
Введите ширину параллелепипеда b
7.5
Введите высоту параллелепипеда h
5.4
-----
Ответ
Площадь поверхности параллелепипеда S равна 318.36 кв. см.
    
```

- 6.4) Напишите программу вычисления объема **V** и площади боковой поверхности **S** цилиндра. Пользователь вводит радиус и высоту. Формулы для расчетов напишите самостоятельно. Вместо числа Пи = 3,1415... используйте стандартную функцию Паскаля **pi**



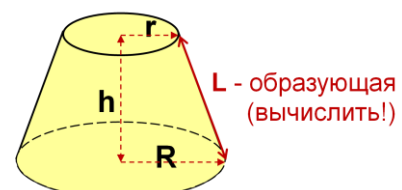
```

C:\TP7\BIN\PROGRAM\LINE\CIL\INDR.EXE
Введите радиус основания цилиндра r
5.2
Введите высоту цилиндра h
8.7
-----
Ответ
Площадь боковой поверхности цилиндра S равна 284.25 кв. см.
Объем цилиндра V равен 739.05 куб. см.
    
```

- 6.5) Найдите объем **V** и площадь боковой поверхности **S** усеченного конуса. Пользователь вводит радиусы оснований **R** и **r** и высоту усеченного конуса **h**. Формулы для расчета запишите самостоятельно.

```

C:\TP7\BIN\PROGRAM\LINE\KONUS.EXE
Введите большой радиус нижнего основания
10.5
Введите малый радиус верхнего основания
5.4
Введите высоту конуса
7.2
-----
Объем конуса 1478.63 куб. см.
Площадь боковой поверхности конуса 440.73 кв. см.
    
```



6.6) При нагревании куска меди от T_1 °C до T_2 °C было затрачено Q КДж тепла (все эти данные вводит пользователь). Определить массу меди.

Для справки: Удельная теплоемкость меди при 20°: $C = 400$ Дж/кг*С (описать как константу)
Не забудьте в программе перевести КДж в Дж!



```

C:\TP7\BIN\PROGRAM\LINEKUSOK_ME.EXE
Введите начальную температуру T1
20
Введите конечную температуру T2
170
Введите количество теплоты Q в Кдж затраченное на нагревание
140
Масса меди      2.33 кг
    
```

6.7) Определите, сколько попугаев поместится на удаве? Пользователь вводит с клавиатуры длину попугая и длину удава в метрах. Ответ округлите до целого числа с помощью функции **trunc**.



```

C:\TP7_1\BIN\PROGRAM\LINEPOPU38.EXE
Введите длину удава
11.6
Введите длину попугая
0.3
Понесится 38 попугаев и одно попугайское крылышко
    
```

6.8) Напишите программу для пересчета величины временного интервала, заданного в минутах – в величину, выраженную в часах и минутах. Используйте операции **div** и **mod**.



```

C:\TP7\BIN\PROGRAM\LINECLOCK.EXE
Введите время в минутах:
150
150 минут – это 2 час 30 минут
    
```

6.9) Напишите программу, разбивающую любое трехзначное число на разряды, и вычисляющую сумму цифр в этом трехзначном числе. Для того, чтобы разбить число на разряды, используйте операции **div** и **mod**.

```

C:\TP7\BIN\PROGRAM\LINEZNA.EXE
Введите трехзначное число от 100 до 999
245
Разряды: Сотни- 2 Десятки- 4 Единицы- 5
Сумма всех цифр этого числа = 11
    
```

6.10) Составьте программу, находящую разность между трехзначным числом и числом, составленным из тех же цифр, но взятых в обратном порядке.

```

C:\TP7_1\BIN\PROGRAM\LINESUM_ZAD.EXE
Введите трехзначное число
235
разница между 235 и 532 равна 297
    
```

6.11) Напишите программу для перевода длины отрезка, заданного в дюймах (**1 дюйм = 2,54 см**) в метрическую систему, т.е. выразите её в **см** и **мм**. Используйте операции: **trunc** (округление с отбрасыванием дробной части), **frac** (взятие дробной части числа), **round** (округление по правилам).

```

C:\TP7\BIN\PROGRAM\LINEMETRE.EXE
Введите длину отрезка в дюймах:
21
53 сантиметров 3 миллиметров
    
```



6.12) Напишите программу вычисления величины дохода по вкладу. От пользователя требуется ввод суммы вклада, срока вклада (в днях) и процентной ставки (в годовых процентах).



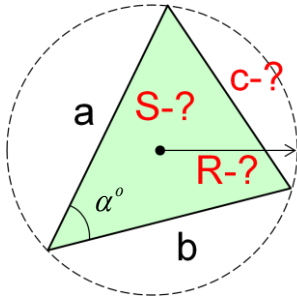
```

C:\TP7\BIN\PROGRAM\LINEBANK.EXE
Введите величину вклада
15000
Введите срок вклада <дней>
90
Введите процентную ставку банка
7.6
Сумма по окончании вклада 15281.10 руб
    
```

6.13) В треугольнике известны длины сторон **a** и **b** и угол между ними α (в градусах).

Составьте программу для вычисления третьей стороны **c** (по теореме косинусов), площади треугольника **S** и радиуса окружности **R**, описанной вокруг треугольника.

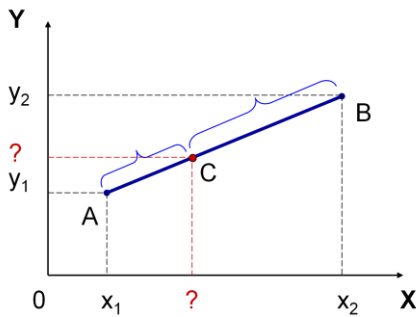
Формула для перевода угла из градусов в радианы: $\alpha(\text{рад}) = \frac{\alpha(\text{град}) \pi}{180}$



```

C:\TP7\BIN\PROGRAM\LINE\TREUG.EXE
Введите длину стороны a
15.2
Введите длину стороны b
12.3
Введите угол в градусах
50
Длина стороны c 11.9 см
Площадь треугольника 71.6 кв см
Радиус описанной окружности 7.8 см
    
```

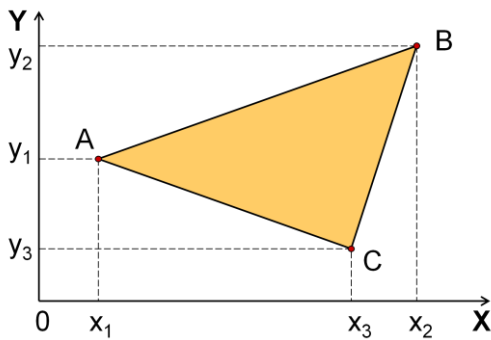
6.14) Напишите программу, в которой определяются координаты точки **C(x,y)**, делящей отрезок **AB** с координатами концов **A(x1,y1)** и **B(x2,y2)** на части таким образом, что **AC=m*AB** (где $0 < m < 1$). От пользователя требуется ввод координат точек **A** и **B** и коэффициента **m**.



```

C:\TP7\BIN\PROGRAM\LINE\OTREZOK.EXE
Введите через пробел координаты X и Y точки A
2 3
Введите через пробел координаты X и Y точки B
7 6
Введите число m [0;1]
0.3
-----
Ответ
Координаты X и Y точки C 3.2; 3.7
    
```

6.15) Напишите программу, в которой по введенным пользователем координатам трех вершин треугольника **A**, **B** и **C** вычисляются длины его сторон **AB**, **BC**, **AC** и площадь **S** (по формуле Герона).

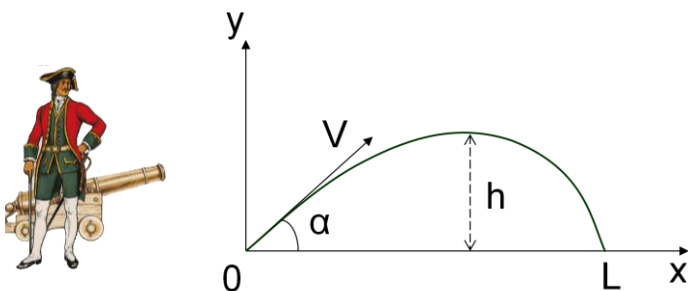


```

C:\TP7\BIN\PROGRAM\LINE\TREUG13.EXE
Введите через пробел координаты точки A
0.9 2.8
Введите через пробел координаты точки B
6.5 5.7
Введите через пробел координаты точки C
4.8 1.2
-----
Стороны треугольника AB= 6.3 BC= 4.8 AC= 4.2
Площадь треугольника ABC = 10.1 кв см
    
```

6.16) Пушка выстреливает под углом α (в градусах) к горизонту со скоростью **V** (м/сек).

На какое расстояние **L** (в метрах) улетит ядро? Найдите максимальную высоту подъема **h** и время полета снаряда **t**. Формулы для перевода угла из градусов в радианы и всех расчетов напишите самостоятельно. Ответ выведите в формате 10:2

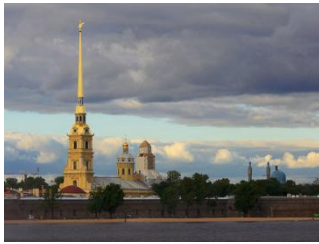


```

C:\TP7\BIN\PROGRAM\LINE\VISTREL.EXE
Введите угол полета снаряда:
60
введите скорость полета:
100
Дальность полета 883.70 метров
Максимальная высота 382.65 метров
Время полета снаряда: 17.67сек
    
```

Ветвление в программах. Оператор If

7.1) Написать программу проверки знания даты основания Санкт-Петербурга. В случае неверного ответа пользователя программа должна выводить правильный ответ. При запуске программы проверьте верный и неверный вариант ответа.



```
C:\TP7\BIN\PROGRAM\IF\SPB.EXE
В каком году был основан Санкт-Петербург?
1650
Вы ошиблись! Год основания - 1703

C:\TP7\BIN\PROGRAM\IF\SPB.EXE
В каком году был основан Санкт-Петербург?
1703
Вы ответили правильно!
```

7.2) Составить программу, определяющую результат гадания на ромашке.

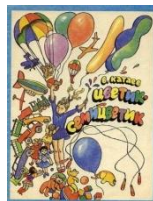
Пользователь вводит исходное количество лепестков. Используйте для проверки нечетности числа функцию **odd(x)**, где **x** – число лепестков (запись условия: **if odd(x) = true**) или проверяйте остаток от деления на **2**



```
C:\TP7\BIN\PROGRAM\IF\ROMASHKA.EXE
Скажи? сколько лепестков у ромашки, и я скажу, любит или нет
25
Любит!

C:\TP7\BIN\PROGRAM\IF\ROMASHKA.EXE
Скажи? сколько лепестков у ромашки, и я скажу, любит или нет
20
Не любит...
```

7.3) Составьте тест из одного вопроса с тремя вариантами ответа. Пользователь должен выбрать ответ и ввести его номер, а программа должна сообщить, верный ответ или нет.



```
C:\TP7\BIN\PROGRAM\IF\MINI_TES.EXE
Кто автор сказки "Яленький цветочек"?
1) П. Бажов
2) С. Аксаков
3) В. Катаев
Введите ваш ответ:
3
Вы все перепутали!

C:\TP7\BIN\PROGRAM\IF\MINI_TES.EXE
Кто автор сказки "Яленький цветочек"?
1) П. Бажов
2) С. Аксаков
3) В. Катаев
Введите ваш ответ:
1
Вы все перепутали!

C:\TP7\BIN\PROGRAM\IF\MINI_TES.EXE
Кто автор сказки "Яленький цветочек"?
1) П. Бажов
2) С. Аксаков
3) В. Катаев
Введите ваш ответ:
2
Правильно!
```

7.4) Написать программу определения стоимости разговора по телефону. Цену за минуту разговора 2.30 руб. опишите как константу. По субботам и воскресеньям предоставляется скидка 20%.



```
C:\TP7\BIN\PROGRAM\IF\TELEFON.EXE
Введите длительность разговора (целое количество минут)
3
Введите день недели: 1-пн, 2-вт ... 7-вс
6
Вам предоставляется скидка 20%
Стоимость разговора 5.52 руб

C:\TP7\BIN\PROGRAM\IF\TELEFON.EXE
Введите длительность разговора (целое количество минут)
3
Введите день недели: 1-пн, 2-вт ... 7-вс
2
Стоимость без скидки 6.90 руб
```


7.5) Фирма осуществляет набор сотрудников. Условия приема требуют **не менее 5 лет** рабочего стажа и возраст **не более 40 лет**. Напишите программу, проверяющую, подходит ли кандидат по этим параметрам.



```

C:\TP7\BIN\PROGRAM\IFRABOTA.EXE
введите возраст сотрудника
28
введите стаж работы
2
Не принимаем
  
```

```

C:\TP7\BIN\PROGRAM\IFRABOTA.EXE
введите возраст сотрудника
50
введите стаж работы
10
Не принимаем
  
```

```

C:\TP7\BIN\PROGRAM\IFRABOTA.EXE
введите возраст сотрудника
35
введите стаж работы
6
Принимаем
  
```

7.6) Напишите программу, которая вычисляет оптимальный вес пользователя, сравнивает его с реальным и выдает рекомендацию о необходимости поправиться или похудеть на нужное количество килограмм.

Для простоты оптимальный вес вычисляйте по формуле: **Рост (в сантиметрах) – 100**.



```

C:\TP7\BIN\PROGRAM\IFVES.EXE
Введите рост в см
160
Введите вес в кг
60
Ваш вес оптимален
  
```

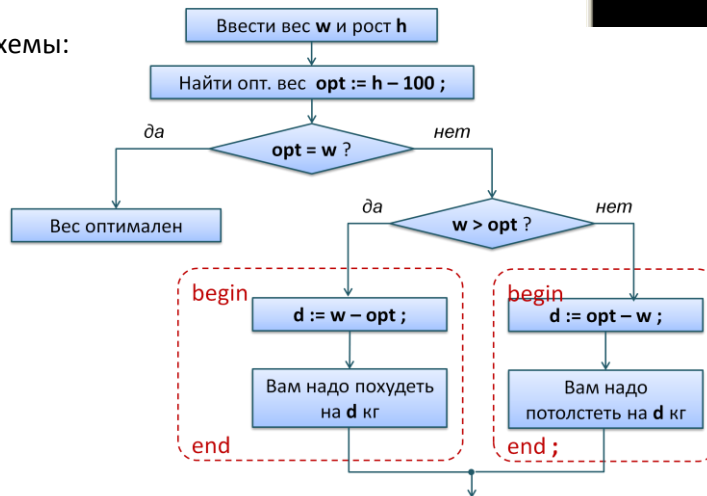
```

C:\TP7\BIN\PROGRAM\IFVES.EXE
Введите рост в см
170
Введите вес в кг
65
Вам надо поправиться на 5 кг
  
```

```

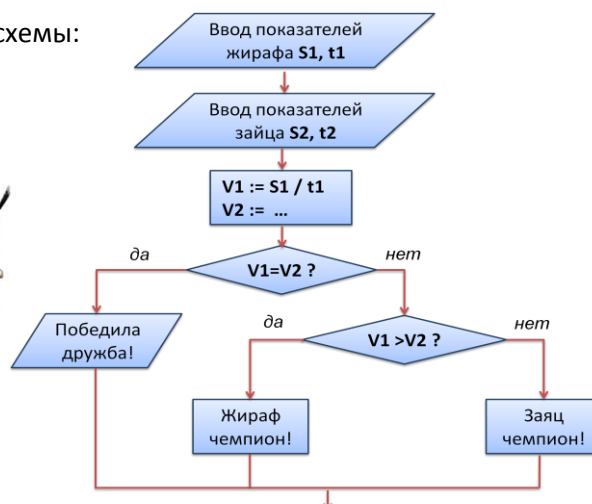
C:\TP7\BIN\PROGRAM\IFVES.EXE
Введите рост в см
165
Введите вес в кг
80
Вам надо похудеть на 15 кг
  
```

Фрагмент блок-схемы:



7.7) Развивая предельно возможную скорость движения, жираф пробегает S_1 метров за t_1 секунд. Заяц пробегает S_2 метров за t_2 секунд. Напишите программу, чтобы определить, кто бежит быстрее?

Фрагмент блок-схемы:



```

C:\TP7\BIN\PROGRAM\IFCHAMPION.EXE
Введите через показатели жирафа s1 t1
335.8 23
Введите через провела показатели зайца s2 t2
218.5 55
Заяц чемпион!
  
```

```

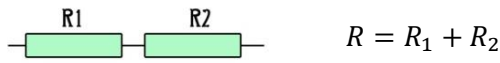
C:\TP7\BIN\PROGRAM\IFCHAMPION.EXE
Введите через показатели жирафа s1 t1
450.3 20
Введите через провела показатели зайца s2 t2
315.2 20
Жираф чемпион!
  
```

```

C:\TP7\BIN\PROGRAM\IFCHAMPION.EXE
Введите через показатели жирафа s1 t1
300 25
Введите через провела показатели зайца s2 t2
600 50
победила дружба!
  
```


7.8) Напишите программу вычисления общего сопротивления электрической цепи, состоящей из двух сопротивлений, соединенных последовательно или параллельно (на выбор 1 или 2). Создайте «защиту» от неправильного выбора типа соединения (реакция на ввод любых чисел, кроме 1 и 2)

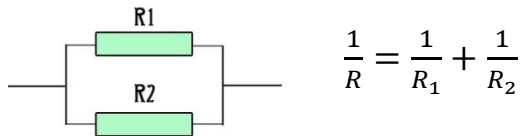
Последовательное соединение:



```

C:\TP7\BIN\PROGRAM\FITOK.EXE
Введите величину первого сопротивления Ом
15.4
Введите величину второго сопротивления Ом
27.3
Выберите тип соединения: 1 - последовательное, 2 - параллельное
1
Общее сопротивление 42.7 Ом
    
```

Параллельное соединение:



```

C:\TP7\BIN\PROGRAM\FITOK.EXE
Введите величину первого сопротивления Ом
15.4
Введите величину второго сопротивления Ом
27.3
Выберите тип соединения: 1 - последовательное, 2 - параллельное
2
Общее сопротивление 9.8 Ом
    
```

```

C:\TP7\BIN\PROGRAM\FITOK.EXE
Введите величину первого сопротивления Ом
15.4
Введите величину второго сопротивления Ом
27.3
Выберите тип соединения: 1 - последовательное, 2 - параллельное
0
Ошибка ввода!
    
```

7.9) Напишите программу для решений квадратных уравнений вида $ax^2 + bx + c = 0$ с любыми коэффициентами. Программа должна выводить корни уравнения (или один корень), или сообщение о том, что корней нет. Вывод ответа в формате 7:2

Дискриминант: $D = b^2 - 4ac$

$$\text{Корни: } x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

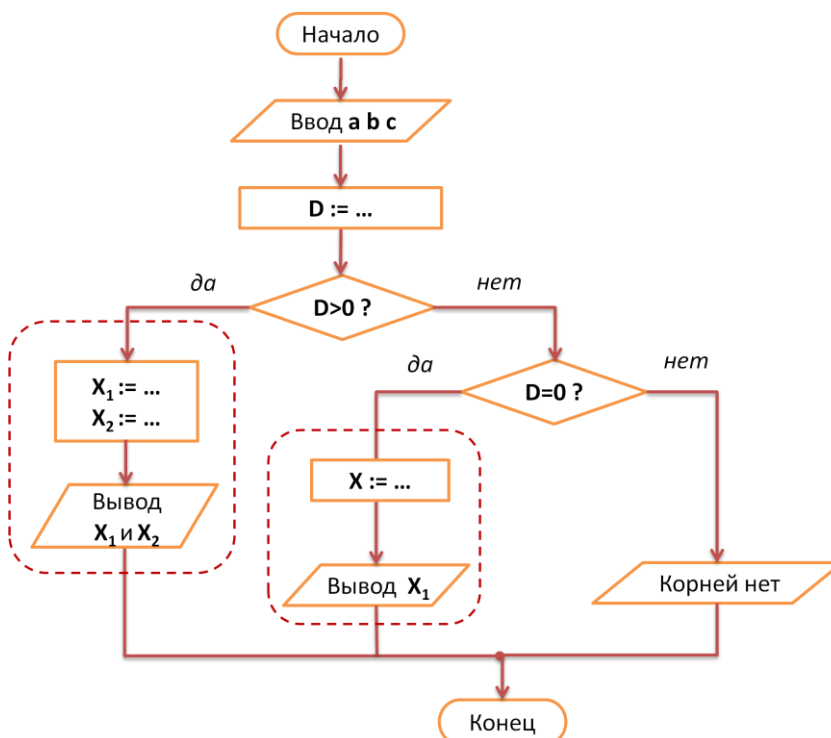
```

C:\TP7\BIN\PROGRAM\FURAVNENI.EXE
Введите коэффициент а:
-5
Введите коэффициент б
7
Введите коэффициент с:
-11
Корней нет!
    
```

Блок-схема:

```

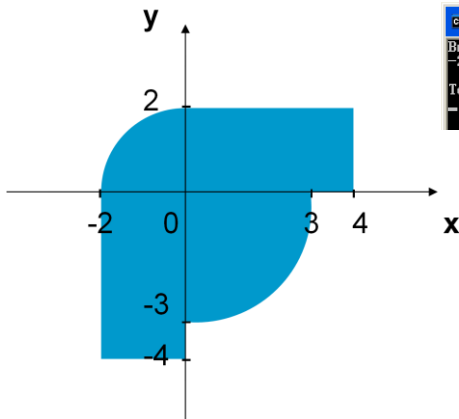
C:\TP7\BIN\PROGRAM\FURAVNENI.EXE
Введите коэффициент а:
1
Введите коэффициент б
6
Введите коэффициент с:
9
Корень уравнения: -3.00
    
```



```

C:\TP7\BIN\PROGRAM\FURAVNENI.EXE
Введите коэффициент а:
3.5
Введите коэффициент б
9
Введите коэффициент с:
-10
Корни уравнения: x1= -3.41 x2= 0.84
    
```

7.10) Составьте программу, определяющую принадлежит ли точка с указанными координатами выделенной области (если точка лежит на границе области, считать, что она принадлежит ей). Выведите в ответе координаты, введенные пользователем.



```

C:\TP7\BIN\PROGRAM\OBLAST.EXE
Введите координаты точки X и Y:
-2.8 1.7
Точка <-2.8; 1.7> НЕ принадлежит выделенной области
  
```

```

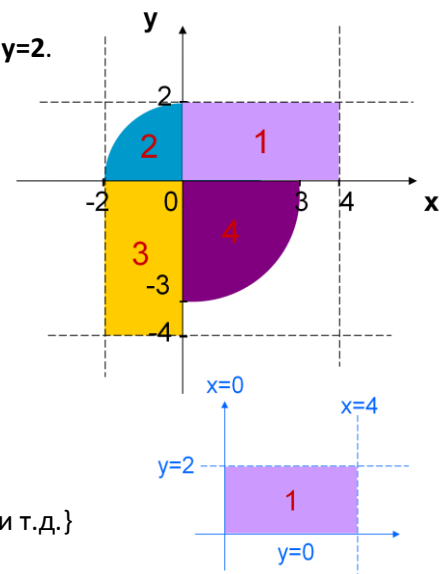
C:\TP7\BIN\PROGRAM\OBLAST.EXE
Введите координаты точки X и Y:
2.5 1.5
Точка < 2.5; 1.5> принадлежит выделенной области
  
```

```

C:\TP7\BIN\PROGRAM\OBLAST.EXE
Введите координаты точки X и Y:
-2 -4
Точка <-2.0;-4.0> принадлежит выделенной области
  
```

Подсказка: для решения этой задачи разбейте область на несколько простых фигур и организуйте в программе проверку сложного условия.

- 1 - прямоугольник, ограниченный осями Ox , Oy и прямыми $x=4$, $y=2$.
- 2 - внутренняя часть сектора, ограниченного отрицательной частью оси Ox , положительной Oy и окружностью с центром в начале координат и радиусом 2.
- 3 - прямоугольник, ограниченный осями Ox , Oy и прямыми $x=-2$, $y=-4$.
- 4 - внутренняя часть сектора, ограниченного положительной частью оси Ox , отрицательной Oy и окружностью с центром в начале координат и радиусом 3.



Сложное условие для 1-ой области:

if $(x \geq 0)$ and $(x \leq 4)$ and $(y \geq 0)$ and $(y \leq 2)$

or ... {далее записать сложное условие для следующей области и т.д.}

7.11) Написать программу вычисления стоимости покупки с учетом скидки. Пользователь вводит стоимость единицы товара и количество товара. Скидка 5% предоставляется, если сумма покупки ≥ 5000 руб. Скидка 10% – если сумма покупки ≥ 10000 руб.

```

C:\TP7\BIN\PROGRAM\IF\POKUPKI.EXE
Введите стоимость товара:
2500
Введите количество товара:
3
Вам предоставляется скидка 5%
Общая сумма с учетом скидки: 7125.00 руб
  
```

```

C:\TP7\BIN\PROGRAM\IF\POKUPKI.EXE
Введите стоимость товара:
12000.5
Введите количество товара:
10
Вам предоставляется скидка 10%
Общая сумма с учетом скидки: 10804.50 руб
  
```

```

C:\TP7\BIN\PROGRAM\IF\POKUPKI.EXE
Введите стоимость товара:
2000
Введите количество товара:
2
Скидка не предоставляется! Общая сумма: 4000.00 руб
  
```

7.12) Пользователь вводит день и месяц своего рождения. Напишите программу для вывода соответствующего знака Зодиака: Водолей (20.1–18.2), Рыбы (19.2–20.3), Овен (21.3–19.4), Телец (20.4–20.5), Близнецы (21.5–21.6), Рак (22.6–22.7), Лев (23.7–22.8), Дева (23.8–22.9), Весы (23.9–22.10), Скорпион (23.10–22.11), Стрелец (23.11–21.12), Козерог (22.12–19.1).



```

C:\TP7_1\BIN\PROGRAM\ZODIAK.EXE
Введите число и месяц рождения
30 1
Водолей
  
```

```

C:\TP7_1\BIN\PROGRAM\ZODIAK.EXE
Введите число и месяц рождения
2 5
Телец
  
```

Оператор выбора Case

8.1) Напишите программу, вычисляющую стоимость междугороднего телефонного разговора. Пользователь вводит код города и продолжительность разговора (название соответствующего города и стоимость минуты должны выводиться в программе по коду):



Город	Код	Цена минуты
Владивосток	423	3,5 руб.
Москва	495	1,2 руб.
Мурманск	815	1,8 руб.
Самара	846	2,4 руб.

```

C:\TP7\BIN\PROGRAM\CASE\TELEFON.EXE
Введите код города
815
Введите длительность разговора <мин>
7
Город Мурманск
Стоимость минуты 1.8 руб
Стоимость разговора 12.60 руб
    
```

```

C:\TP7\BIN\PROGRAM\CASE\TELEFON.EXE
Введите код города
423
Введите длительность разговора <мин>
10
Город Владивосток
Стоимость минуты 3.5 руб
Стоимость разговора 35.00 руб
    
```

8.2) На станции автозаправки есть топливо пяти марок по разной цене. Клиент выбирает марку бензина и указывает количество литров. Напишите программу для расчета суммы, которую нужно заплатить.



```

C:\TP7\BIN\PROGRAM\CASE\BENZIN.EXE
Введите марку бензина
1 - А80
2 - А92
3 - А95
4 - А98
5 - ДТ
Укажите количество литров
20
Заплатите в кассу 316.00 руб
    
```

```

C:\TP7\BIN\PROGRAM\CASE\BENZIN.EXE
Введите марку бензина
1 - А80
2 - А92
3 - А95
4 - А98
5 - ДТ
Укажите количество литров
20
Заплатите в кассу 464.00 руб
    
```

8.3) Напишите программу, которая печатает «Доброе утро», «Добрый день», «Добрый вечер» или «Спокойной ночи», в зависимости от времени суток, которое вводится с клавиатуры на запрос программы. В случае ввода неверных чисел (больше 24 часов или больше 59 минут) выведите сообщение об ошибке. При запуске проверьте все варианты.

```

C:\TP7\BIN\PROGRAM\CASE\ZDRASTE.EXE
Введите время <часы минуты>
6 30
Доброе утро!
    
```

```

C:\TP7\BIN\PROGRAM\CASE\ZDRASTE.EXE
Введите время <часы минуты>
18 45
Добрый вечер!
    
```

```

C:\TP7\BIN\PROGRAM\CASE\ZDRASTE.EXE
Введите время <часы минуты>
14 00
Добрый день!
    
```

```

C:\TP7\BIN\PROGRAM\CASE\ZDRASTE.EXE
Введите время <часы минуты>
1 15
Спокойной ночи!
    
```

```

C:\TP7\BIN\PROGRAM\CASE\ZDRASTE.EXE
Введите время <часы минуты>
25 70
Ошибка!
    
```

8.4) Напишите программу, которая после введенного с клавиатуры числа (в диапазоне от 1 до 99), обозначающего денежную единицу, дописывает слово «копейка» в правильной форме, например: 23 копейки, 41 копейка, 50 копеек и т. д. Подумайте, от чего это зависит?



```

C:\TP7\BIN\PROGRAM\CASE\KOPEIKI.EXE
Введите количество копеек от 0 до 99
15
15 копеек
    
```

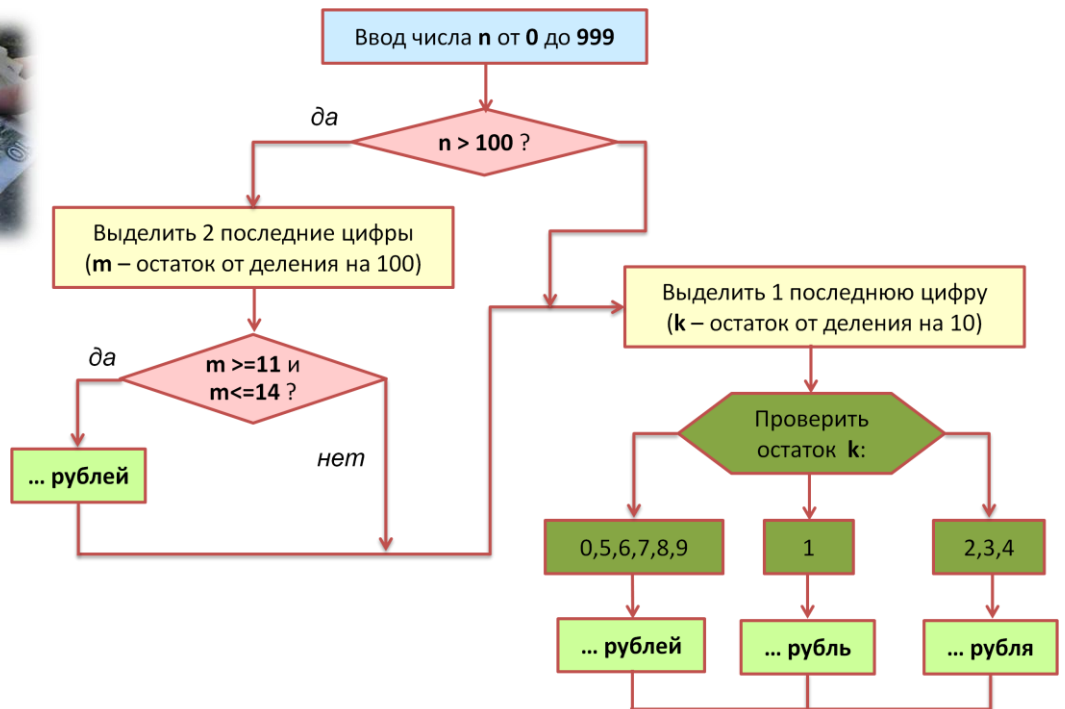
```

C:\TP7\BIN\PROGRAM\CASE\KOPEIKI.EXE
Введите количество копеек от 0 до 99
53
53 копейки
    
```

```

C:\TP7\BIN\PROGRAM\CASE\KOPEIKI.EXE
Введите количество копеек от 0 до 99
71
71 копейка
    
```

8.5) Напишите программу, которая после введенного с клавиатуры числа (в диапазоне от 1 до 999), дописывает слово «рубль» в правильной форме, например: 3 рубля, 45 рублей, 971 рубль и т.д. Подумайте, от чего зависит форма слова «рубль», какие есть исключения, как можно это математически реализовать?



```

C:\TP7\BIN\PROGRAM\CASE\RUBL.EXE
Введите число от 1 до 999
71
-----
71 рубль
  
```

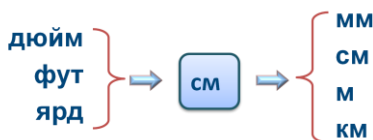
```

C:\TP7\BIN\PROGRAM\CASE\RUBL.EXE
Введите число от 1 до 999
852
-----
852 рубля
  
```

```

C:\TP7\BIN\PROGRAM\CASE\RUBL.EXE
Введите число от 1 до 999
5
-----
5 рублей
  
```

8.6) Составьте программу для перевода английских мер длины в общепринятые. При запуске проверьте все варианты. Используйте оператор Case дважды: сначала переведите выбранную меру в какую либо базовую единицу, например, в сантиметры (по таблице), а затем переведите в мм, м, км.



1 дюйм			2,54 см
1 фут	12 дюймов		30,48 см
1 ярд	3 фута	36 дюймов	91,44 см

Переменные:

x – исходное число;

n – выбранные исходные единицы (дюйм, фут, ярд);

y – базовая единица (промежуточный результат, его выводить не нужно!);

m – выбранные единицы (во что перевести мм, см, м, км);

z – результат

```

C:\TP7\BIN\PROGRAM\CASE\ED_DLINI.EXE
Введите число
6.5
Введите единицу измерения:
1 - дюйм
2 - фут
3 - ярд
2
Введите, в какие единицы перевести:
1 - мм
2 - см
3 - м
4 - км
3
Ответ 1.98
  
```

```

C:\TP7\BIN\PROGRAM\CASE\ED_DLINI.EXE
Введите число
25.5
Введите единицу измерения:
1 - дюйм
2 - фут
3 - ярд
1
Введите, в какие единицы перевести:
1 - мм
2 - см
3 - м
4 - км
1
Ответ 647.70
  
```

```

C:\TP7\BIN\PROGRAM\CASE\ED_DLINI.EXE
Введите число
1500
Введите единицу измерения:
1 - дюйм
2 - фут
3 - ярд
3
Введите, в какие единицы перевести:
1 - мм
2 - см
3 - м
4 - км
4
Ответ 1.37
  
```

Задачи о календаре

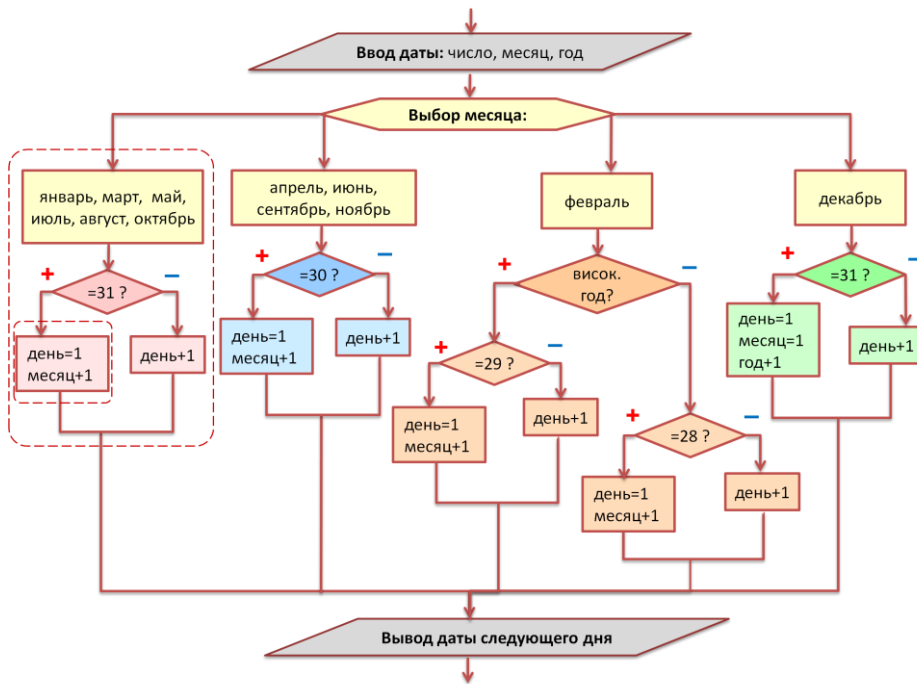
8.7) Напишите программу, которая вычисляет дату следующего дня. Учтите количество дней в каждом месяце, високосный год или нет. Особое внимание уделите последнему дню в каждом месяце (установка 1 дня следующего месяца), последнему дню года (установка 1 дня 1 месяца следующего года) и 28/29 февраля в високосном или невисокосном году.

Год является **високосным**:

- если он кратен 4, и при этом не кратен 100;
- **или** кратен 400.

Кратное число – целое число, делящееся на какое-либо число без остатка, т.е. **остаток = 0**

Сложное условие: $(Y \bmod 4 = 0) \text{ and } (Y \bmod 100 > 0) \text{ or } (Y \bmod 400 = 0)$



```

C:\TP7\BIN\PROGRAM\CASEWEXT_DAY.EXE
Введите любую дату
13 5 1996
Следующий день: 14 5 1996
-

C:\TP7\BIN\PROGRAM\CASEWEXT_DAY.EXE
Введите любую дату
31 1 2003
Следующий день: 1 2 2003
-

C:\TP7\BIN\PROGRAM\CASEWEXT_DAY.EXE
Введите любую дату
28 2 2006
Следующий день: 1 3 2006
-

C:\TP7\BIN\PROGRAM\CASEWEXT_DAY.EXE
Введите любую дату
28 2 2008
Следующий день: 29 2 2008
-

C:\TP7\BIN\PROGRAM\CASEWEXT_DAY.EXE
Введите любую дату
31 12 2009
Следующий день: 1 1 2010
-
    
```

8.8) Напишите программу, вычисляющую день недели для любой даты григорианского календаря.

Исходные данные: **d** – день, **m** – месяц, **y** – год;

Промежуточные данные: **a** – сдвиг, **b** – год с учетом сдвига, **c** – месяц с учетом сдвига;

Результат: **denned** – день недели.

Алгоритм:

- Номер месяца преобразовывается таким образом, чтобы март был 1-ым месяцем, а февраль – 12-ым и относился к прошлому году (например, 03 1998 → 01 1998, 02 1983 → 12 1982)

$$a = (14 - m) \text{ div } 12;$$

- Определяются год и месяц с учётом этого сдвига:

$$b = y - a;$$

$$c = m + 12 * a - 2;$$

- Вычисляется день недели:

$$denned = (7000 + (d + b + (b \text{ div } 4) - (b \text{ div } 100) + (b \text{ div } 400) + (31 * c) \text{ div } 12)) \text{ mod } 7;$$

Результат: 0 – воскресенье, 1 – понедельник, 2 – вторник, 3 – среда и т. д.

2007 October							Февраль 2008							Июнь 2009						
Пн	Вт	Ср	Чт	Пт	Сб	Вс	Пн	Вт	Ср	Чт	Пт	Сб	Вс	Пн	Вт	Ср	Чт	Пт	Сб	Вс
1	2	3	4	5	6	7					1	2	3	1	2	3	4	5	6	7
8	9	10	11	12	13	14	4	5	6	7	8	9	10	8	9	10	11	12	13	14
15	16	17	18	19	20	21	11	12	13	14	15	16	17	15	16	17	18	19	20	21
22	23	24	25	26	27	28	18	19	20	21	22	23	24	22	23	24	25	26	27	28
29	30	31	25	26	27	28	29	29	30											

```

C:\TP7\BIN\PROGRAM\CASEWEEK.EXE
Введите дату (число, месяц, год):
12 1 2010
- Вторник

C:\TP7\BIN\PROGRAM\CASEWEEK.EXE
Введите дату (число, месяц, год):
1 9 2000
- Пятница
    
```

Случайные числа. Новогодние задачи

9.1) К празднику подготовили 36 пронумерованных записок, из которых 30 записок – с предсказаниями, например: «Готовьте кошелек для больших денег», «Вас ждет счастье в Новом году», «Весь год будешь ходить в обновках» и т.д. В операторе Case предсказания можно повторять для нескольких не идущих подряд чисел, но не более чем для 3-4 чисел (например, для 1, 12, 25 и 30). В остальных случаях напишите «Новый год ничего не изменит в вашей жизни». Выбор записки с номером должен производиться случайным образом.



```

C:\TP7\BIN\PROGRAM\GADALKA.EXE
* * * * *
НОВОГОДНИЕ ГАДАНИЯ
* * * * *
Для того, чтобы узнать свою судьбу, нажмите Enter
>
C:\TP7\BIN\PROGRAM\CASE\GADALKA.EXE
* * * * *
НОВОГОДНИЕ ГАДАНИЯ
* * * * *
Для того, чтобы узнать свою судьбу, нажмите Enter
>
Вам выпало число 6
Новый год ничего не изменит в вашей жизни...
_
C:\TP7\BIN\PROGRAM\CASE\GADALKA.EXE
* * * * *
НОВОГОДНИЕ ГАДАНИЯ
* * * * *
Для того, чтобы узнать свою судьбу, нажмите Enter
>
Вам выпало число 27
Вас ждут интересные путешествия!
_
    
```

9.2) Создайте программу, которая **случайным образом** выбирает Деда мороза из разных стран мира и поздравляет с Новым годом на соответствующем языке:

№	Страна	Имя	Поздравление
1.	Россия	Дед Мороз	С Новым годом!
2.	Италия	Баббо Натале	Felice anno nuovo!
3.	Финляндия	Йоулупукки	Onnellista Uutta Vuotta!
4.	Норвегия	Юлебукк	Godt Nyttar!
5.	Великобритания	Фазер Кристмас	Happy New Year!
6.	Албания	Бабадимпи	Gezuar Vitin e Ri!
7.	Греция	Агиос Василис	Kenourios Chronos!
8.	Татарстан	Кышбабай	Яна ел белэн!
9.	Франция	Пер Ноэль	Bonne Annee!
10.	Япония	Сегацу-сан	Akimashite Omedetto Gozaimasu!



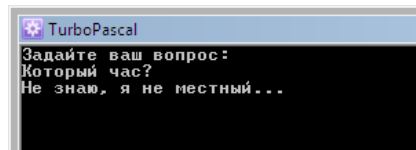
```

C:\TP7\BIN\PROGRAM\DED_MORO.EXE
Нажмите Enter, чтобы получить новогоднее поздравление! Кривле-кравле-бунс!
* * * * *
Вас поздравляет русский Дед Мороз – С новым годом!
_
C:\TP7\BIN\PROGRAM\DED_MORO.EXE
Нажмите Enter, чтобы получить новогоднее поздравление! Кривле-кравле-бунс!
* * * * *
Японский Сегацу-сан говорит вам – Akimashite Omedetto Gozaimasu!
_
C:\TP7\BIN\PROGRAM\DED_MORO.EXE
Нажмите Enter, чтобы получить новогоднее поздравление! Кривле-кравле-бунс!
* * * * *
Вам пришло поздравление от финского Йоулупукки – Onnellista Uutta Vuotta!
_
    
```


9.3) Напишите шуточную программу-тест, в которой пользователь задает свой вопрос, а программа выдает ответ **случайным образом** из списка «универсальных ответов»:

- Это элементарно, Ватсон!
- Спроси у Яндекса!
- Ишь ты, какой любопытный!
- А ты сам не знаешь?

и т.д. (всего придумайте 10 вариантов ответов)



9.4) Восточный календарь основан на астрономических циклах движения Солнца и Земли, Луны, Юпитера и Сатурна. Астрономы Древнего Востока определили, что кругооборот Юпитера вокруг Солнца совершается за 12 лет, а Сатурна – около 30. Было решено за основу цикла принять время, за которое Сатурн дважды совершает свой кругооборот. Это время равно 60 годам. Получается, что за 60 лет (2 оборота Сатурна) Юпитер успевает сделать вокруг Солнца 5 оборотов, каждый из которых занимает 12 лет. Так появился 12-летний календарь. По этому календарю, каждому году покровительствует одно из 12 животных, которые поочередно сменяют друг друга: **Крыса, Буйвол, Тигр, Кролик (или Кот), Дракон, Змея, Лошадь, Овца, Обезьяна, Петух, Собака и Свинья**. Получается, что в календарном цикле, который длится 60 лет, одно животное появляется 5 раз. И с каждым новым появлением оно окрашивается в новый цвет, определяющий какая стихия или элемент природы им управляет.

Определить это можно **по последней цифре** в номере наступающего года:

- 0, 1 – Белый** (Металл);
- 2, 3 – Черный** (Вода);
- 4, 5 – Голубой** (Дерево);
- 6, 7 – Красный** (Огонь);
- 8, 9 – Желтый** (Земля)



Напишите программу, которая позволяет по введенному номеру года вывести его название по восточному календарю. За начало отсчета примите **год Белой Крысы**, например, **1900 г.** или **1960 г.**

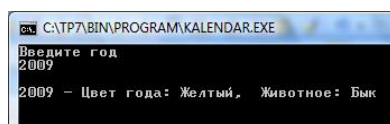
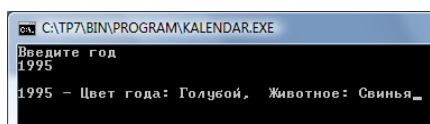
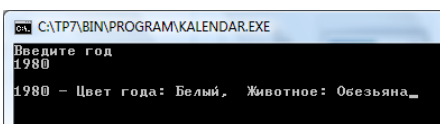
Подсказки:

a – год начала отсчета (1900 или 1960) – опишите как константу;

t – исследуемый год (вводит пользователь);

k – остаток от деления года **t** на **10** (последняя цифра в номере года) – для определения цвета года;

d – остаток от деления разницы между исследуемым годом и годом начала отсчета (**t – a**) на **12** – для определения животного: 0 – Крыса, 1 – Бык, 2 – Тигр, 3 – Кролик... 11 – Свинья



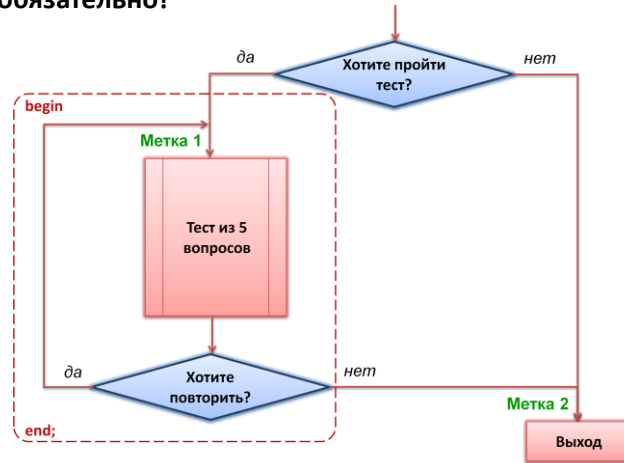
Оператор безусловного перехода Goto

10.1) Составьте программу-тест из **5 вопросов** (к каждому вопросу дается **3 варианта ответа**).

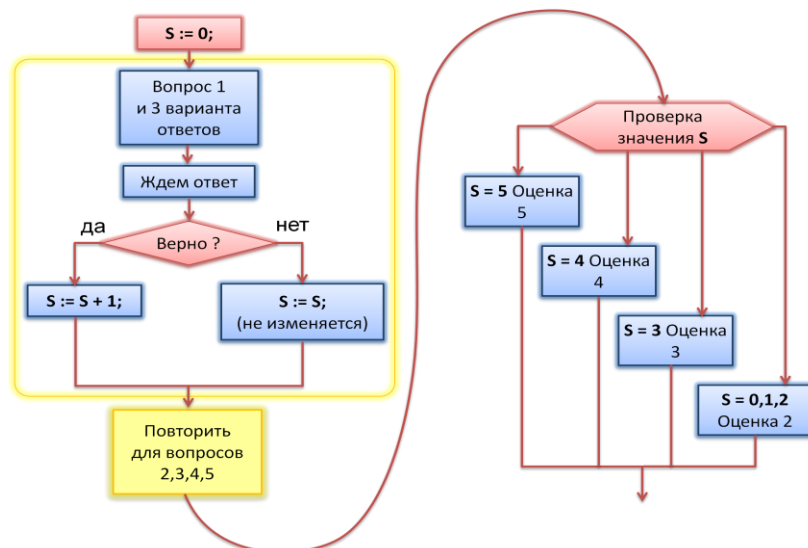
В начале программы предложите выбрать, проходить ли тест, если выбран ответ «нет» – происходит выход из программы. Если выбран ответ «да», то появляется первый вопрос. За каждый правильный ответ начисляется **1 балл**. В конце теста выведите сообщение о количестве набранных баллов и поставьте оценку. В конце предложите повторить тест, в случае согласия вернуться к 1-му вопросу.

Использование цветного шрифта обязательно!

Блок-схема в общем виде:



Блок-схема теста:



```

C:\TP7\BIN\PROGRAM\CASE\TEST.EXE
Хотите пройти шуточный тест(Y/N)?
Y
-----
1) Из чего был сделан Буратино?
1- из снега
2- из полена
3- из теста
1
-----
2) Какой предмет искали в сказке про Буратино?
1- сундук
2- дверь
3- ключик
2
-----
3) В какой цвет Мальвина красила свои волосы?
1- в голубой
2- в черный
3- в красный
1
-----
4) Какой породы был Артемон?
1- овчарка
2- боксер
3- пудель
3
-----
5) Что пропагандировал Дуремар как самое лучшее лечебное средство?
1- черепаховый суп
2- пивяки
3- похлебку из лука
2
-----
Вы набрали 4 баллов
Хорошо, но можно лучше!
Хотите повторить тест(Y/N)?
N
До свидания!
    
```

Арифметический цикл For

11.1) Напишите программу, которая выводит таблицу степеней 2^n (от 2^0 до 2^{10}).

Используйте вывод чисел в столбик по формату (x:3, y:10), где x – показатель степени, y – значение.

```

C:\TP7\BIN\PROGRAM\stepen2.EXE
Таблица степеней 2
0      1
1      2
2      4
3      8
4     16
5     32
6     64
7    128
8    256
9    512
10   1024
    
```

11.2) Напишите программу, которая вычисляет сумму n -первых целых положительных чисел

$$S = 1 + 2 + 3 + \dots + n$$

Количество суммируемых чисел n должно вводиться пользователем во время работы программы.

```

C:\TP7\BIN\PROGRAM\SUMMA1.EXE
Вычисление суммы n-целых положительных чисел
Введите количество суммируемых чисел
20
Сумма чисел = 210

C:\TP7\BIN\PROGRAM\FORSUMMA1.EXE
Вычисление суммы n-целых положительных чисел
Введите количество суммируемых чисел
100
Сумма чисел = 5050
    
```

11.3) Остров Манхэттен (район Нью-Йорка) был выкуплен индейцев переселенцами из Амстердама в **1626 г.** всего за **24 \$**. Каково было бы в настоящее время состояние их счета, если бы эти деньги были помещены тогда в банк под **7%** годового дохода?



```

C:\TP7\BIN\PROGRAM\MANHATTE.EXE
Введите год покупки острова Манхэттен:
1626
Сколько долларов положили в банк?
24
Введите банковский процент
7
Введите текущий год:
2009
Сейчас у них на счету было бы 4307230318000.00 долларов!
    
```

11.4) Напишите программу, которая вычисляет сумму n -первых членов ряда $S = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots$

Количество суммируемых n -членов ряда задается пользователем.

```

C:\TP7\BIN\PROGRAM\SUMA_DRO.EXE
Сумма числового ряда 1+1/2+1/3+...+1/n
Введите количество чисел:
15
Сумма равна 3.318
    
```

```

C:\TP7\BIN\PROGRAM\FORSUMA_DRO.EXE
Сумма числового ряда 1+1/2+1/3+...+1/n
Введите количество чисел:
100
Сумма равна 5.187
    
```

11.5) Выведите на экран таблицу перевода из градусов по шкале Цельсия в градусы по Фаренгейту для значений от 0°C до 30°C с шагом в 1°C . Выведите ответ в 2 столбика.

Формула для перевода $^\circ\text{C}$ в $^\circ\text{F}$: $^\circ\text{F} = 1,8 \cdot ^\circ\text{C} + 32$

```

C:\TP7\BIN\PROGRAM\FARENGEI.EXE
Цельсий  Фаренгейт
0      32.0
1      33.8
2      35.6
3      37.4
4      39.2
5      41.0
6      42.8
7      44.6
8      46.4
9      48.2
10     50.0
11     51.8
12     53.6
13     55.4
14     57.2
15     59.0
16     60.8
17     62.6
18     64.4
19     66.2
20     68.0
21     69.8
22     71.6
23     73.4
24     75.2
25     77.0
26     78.8
27     80.6
28     82.4
29     84.2
30     86.0
    
```

- 11.6) Напишите программу, которая выводит таблицу значений функции: $y = |x - 2| + \frac{|x + 1|}{2}$
 В диапазоне значений $[-4, 4]$ с шагом приращения аргумента $0,5$.
 Выведите ответ в таблицу в два столбика по формату: **Writeln (x:8:2, ' | ', y:8:2);**

```

C:\TP7\BIN\PROGRAM\FUNKSIA.EXE
-----
x      |      y
-----
-4.00  |      7.50
-3.50  |      6.75
-3.00  |      6.00
-2.50  |      5.25
-2.00  |      4.50
-1.50  |      3.75
-1.00  |      3.00
-0.50  |      2.25
 0.00  |      2.50
 0.50  |      2.25
 1.00  |      2.00
 1.50  |      1.75
 2.00  |      1.50
 2.50  |      2.25
 3.00  |      3.00
 3.50  |      3.75
 4.00  |      4.50
    
```

- 11.7) Индийский раджа Шерам позвал к себе изобретателя шахмат ученого Сету и предложил, чтобы тот сам выбрал себе награду за создание столь интересной и мудрой игры. Сета попросил выдать ему за первую клетку шахматной доски 1 пшеничное зерно, за вторую – 2, за третью – 4, за четвертую – 8, за пятую – 16 и т. д., удваивая количество зерен за каждую следующую клетку. Жадный раджа обрадовался столь низкой плате и приказал слугам немедленно выдать награду. Но, как оказалось, сделать это было почти невозможно... Узнайте, почему?

Составьте программу, которая могла бы вычислить:

- Количество зерен на любой N-ой клетке шахматной доски.
- Сумму всех зерен до N-ой клетки включительно.
- Сколько весит такое количество зерна (в кг), если вес 1 пшеничного зернышка **0,05 г**?
- Какой объем займет такое количество зерен, если в **1 м³** примерно **15 млн. зерен**?



```

C:\TP7\BIN\PROGRAM\SHAH_MAT.EXE
Введите номер шахматной клетки от 1 до 64
30
На 30-ой клетке будет лежать 536870912 зерен пшеницы
Всего на 30-и клетках будет 1073741823 зернышка пшеницы
Вес зерна 53687.09 кг
Объем зерна 71.58 куб.м
    
```

- 11.8) Напишите программу для подсчета суммы всех положительных и суммы всех отрицательных **случайных целых чисел** из диапазона от **-100** до **100**. Количество случайных чисел вводится пользователем.

```

C:\TP7\BIN\PROGRAM\SLUCHAI.EXE
Введите количество случайных чисел:
25
Сумма отрицательных чисел равна -671
сумма положительных чисел равна 742
    
```

```

C:\TP7\BIN\PROGRAM\FOR\SLUCHAI.EXE
Введите количество случайных чисел:
120
Сумма отрицательных чисел равна -3487
сумма положительных чисел равна 2789
    
```

- 11.9) Старинная задача: сколько можно купить быков, коров и телят, если плата за быка 10 рублей, за корову – 5 рублей, за телёнка – полтинник (0,5 рубля), если на 100 рублей надо купить 100 голов скота. **Подсказка:** запишите два уравнения: для подсчета количества животных и их стоимости. Используя вложенные циклы, запишите программу для решения этой задачи.



- 11.10) Даны натуральные числа n и k . Составить программу для вычисления выражения $S = 1^k + 2^k + \dots + n^k$
Подсказка: используйте кроме переменных n, k дополнительные переменные: i – счетчик циклов для n , y – для накопления i^k , S – сумма. Для вычисления указанной суммы нужно организовать цикл, в котором вычислялось бы очередное значение $y = i^k$ и осуществлялось бы накопление суммы прибавлением полученного очередного слагаемого y к сумме всех предшествующих ($s := s + y$).

```

C:\TP7\BIN\PROGRAM\SUMMANK.EXE
Введите количество чисел:
10
Введите степень:
5
Ответ: 24217
  
```

- 11.11) Белки спрятали в дупле N орехов. Каждый день они брали половину для себя и своих друзей. Если число орехов было нечетным, то они откладывали один орех в сторону, а остальные делили пополам, причем отложенный орех опять возвращали в дупло. Напишите программу, подсчитывающую, сколько орехов останется в мешке через K дней?



```

C:\TP7\BIN\PROGRAM\BELKI.EXE
Введите количество орехов, которые собрали белки: 250
Введите количество дней: 7
Остаток: 2 орех(-а -ов).
  
```

```

C:\TP7\BIN\PROGRAM\FOR\BELKI.EXE
Введите количество орехов, которые собрали белки: 1000
Введите количество дней: 25
Остаток: 1 орех(-а -ов).
  
```

- 11.12) Напишите программу, выводящую на экран электронный таймер, который работает в течение, например, 5 минут. Не забудьте подключить модуль **Crt** и очищать экран при смене цифр процедурой **Clrscr**; Для отсчета времени при переборе цифр используйте процедуру **Delay (60000)**; – задержка на 60000 мс. Измените программу таймера так, чтобы в ней шел обратный отсчет времени (используйте цикл **downto**)

```

TURBO PASCAL 7.1
Таймер
3 : 34
  
```

```

TURBO PASCAL 7.1
Обратный отсчет
0 : 0
Ваше время истекло!
  
```

- 11.13) Напишите программу, чтобы получить перевертыш целого четырехзначного числа n , введенного пользователем (вспомните, как разбить число на разряды с помощью операций **div** и **mod**)?

```

C:\TP7\BIN\PROGRAM\FOR\PEREVERT.EXE
Введите целое число не большее 9999
5083
Число-перевертыш 3805
  
```

- 11.14) Напишите программу, которая преобразует введенное пользователем десятичное число (в интервале от 0 до 255) в восьмиразрядное двоичное число.

Числа для проверки:

- $8_{10} = 00001000_2$
- $99_{10} = 01100011_2$
- $155_{10} = 10011011_2$
- $200_{10} = 11001000_2$

```

C:\TP7\BIN\PROGRAM\PEREVOD_.EXE
Введите десятичное число:
49
Десятичному числу 49 соответствует двоичное 00110001_
  
```

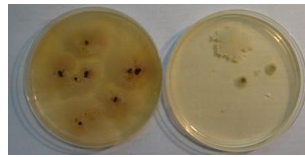
Итерационный цикл Repeat

12.1) В результате применения пенициллина концентрация болезнетворных бактерий в крови больного уменьшается на 1/3 в день от содержимого предыдущего дня. Через сколько дней наступит выздоровление, если концентрация бактерий должна уменьшиться от N1 до N2?

Для проверки:

n1=10000 n2=20 ⇒ 16 дней

n1=5200 n2=5 ⇒ 18 дней



```
TurboPascal
Введите начальную концентрацию: 27
Введите конечную концентрацию : 4
Больной поправится через: 5 дн.
```

12.2) Через сколько дней в зоне заражения уровень радиации упадет до безопасного значения 30 микрорентген в час, если начальное значение уровня заражения Z и каждый день оно уменьшается на 5% по отношению к предыдущему дню.

Для проверки:

700 ⇒ 62 дня

19500 ⇒ 127 дней

25000 ⇒ 132 дня

```
TURBO PASCAL 7.1
Введите начальное значение уровня радиации: 13000
Безопасным уровень радиации станет через <дней>: 119
```



12.3) Напишите программу, которая определяет, является ли целое число (>1), введенное пользователем с клавиатуры, простым или составным (простое число – это натуральное число, которое имеет только 2 различных делителя: 1 и само себя, все остальные числа, называются составными).

```
C:\TP7\BIN\PROGRAM\PROSTOE.EXE
Введите целое число
12345
12345 - составное число
```

```
C:\TP7\BIN\PROGRAM\PROSTOE.EXE
Введите целое число
3571
3571 - простое число
```

12.4) Составьте программу планирования закупки товара в магазине на сумму, не превышающую заданную величину. Для решения этой задачи используйте переменные:

P – сумма денег, которая есть на руках у покупателя,

X – цена товара,

k – количество товара,

S – общая стоимость покупки (начальное значение **S=0**),

d – сумма, на которую будет превышена общая стоимость покупок.

Значение имеющейся у него суммы денег вводит пользователь. Необходимо повторять запрос цены и количества выбранного товара, вычислять его стоимость, суммировать ее с общей стоимостью покупок и выводить результат на экран до тех пор, пока она не превысит имеющуюся у него сумму. В этом случае на экран нужно вывести сообщение о том, на сколько будет превышена сумма покупок.



```
C:\TP7\BIN\PROGRAM\REPEAT\SHOPING.EXE
Введите, сколько у вас денег
5600
Введите цену товара
200
Введите количество товара
5
Вы потратили 1000 руб
Введите цену товара
1500
Введите количество товара
3
Вы потратили 5500 руб
Введите цену товара
250
Введите количество товара
2
Вы потратили 6000 руб
Стоимость покупки превысила вашу сумму на 400 руб
```

Итерационный цикл While

13.1) На склад привозят однородный груз на машинах различной грузоподъемности. Пользователь вводит информацию о весе груза очередной машины. Составить программу подсчета количества машин, прибывших на склад до его заполнения, если вместимость склада не более 100 тонн.



```

CATP7_1\BIN\PROGRAM\WHILE\GRUZOVIK.EXE
Введите вес груза прибывшей машины
15
Введите вес груза прибывшей машины
20
Введите вес груза прибывшей машины
23
Введите вес груза прибывшей машины
36
Введите вес груза прибывшей машины
28
Если разгрузить эту машину, то склад будет переполнен на 22.000 кг
Число полностью разгруженных машин: 4
    
```

```

CATP7_1\BIN\PROGRAM\WHILE\GRUZOVIK.EXE
Введите вес груза прибывшей машины
45
Введите вес груза прибывшей машины
30
Введите вес груза прибывшей машины
62
Если разгрузить эту машину, то склад будет переполнен на 37.000 кг
Число полностью разгруженных машин: 2
    
```

13.2) Напишите программу, чтобы определить, сколько лет нужно хранить **K** долларов в банке под **P** % годовых, чтобы стать миллионером?



```

C:\TP7\BIN\PROGRAM\MILLIO-1.EXE
Введите количество долларов: 12500
Введите банковский процент: 7.8
Вы станете миллионером всего через: 59 лет!
    
```

```

C:\TP7\BIN\PROGRAM\MILLIO-1.EXE
Введите количество долларов: 5000
Введите банковский процент: 9.5
Вы станете миллионером всего через: 84 лет!
    
```

```

C:\TP7\BIN\PROGRAM\MILLIO-1.EXE
Введите количество долларов: 5
Введите банковский процент: 10
Вы станете миллионером всего через: 129 лет!
    
```

13.3) У студента имеются накопления **S** руб. Ежемесячная стипендия составляет **A** рублей, а расходы на проживание **превышают ее** и составляют **B** руб. в месяц (все значения переменных пользователь вводит с клавиатуры). Рост цен ежемесячно увеличивает расходы на **3%**. Напишите программу, чтобы определить, сколько месяцев сможет прожить студент, используя только накопления и стипендию.

```

C:\TP7\BIN\PROGRAM\WHILE\STUDENT.EXE
Введите имеющуюся в наличии сумму сбережений
10000
Введите сумму, которую получает студент в виде стипендии
2250
Введите сумму, которую студент расходует на проживание
4000
Студент сможет прожить еще 7 месяцев
    
```

```

C:\TP7\BIN\PROGRAM\WHILE\STUDENT.EXE
Введите имеющуюся в наличии сумму сбережений
5000
Введите сумму, которую получает студент в виде стипендии
1200
Введите сумму, которую студент расходует на проживание
2500
Студент сможет прожить еще 5 месяцев
    
```



13.4) Спортсмен-лыжник начал тренировки, пробежав в первый день 10 км. Каждый следующий день он увеличивал дистанцию на **P**%. Напишите программу, определяющую, в какой день суммарный пробег лыжника за все дни превысит 200 км. Выведите в программе найденное количество дней (целое число) и суммарный пробег (вещественное число).



```

CATP7_1\BIN\PROGRAM\WHILE\LYNIK.EXE
Введите процент P
4
Суммарный пробег составил 208.25 км
Лыжник бежал 16 дней
    
```

```

CATP7_1\BIN\PROGRAM\WHILE\LYNIK.EXE
Введите процент P
10.2
Суммарный пробег составил 206.43 км
Лыжник бежал 11 дней
    
```


- 13.5) Напишите программу для перестановки цифр в числе в обратном порядке, используя цикл **while**.
Используйте тип данных **longint** (целые числа в диапазоне от -2147483648 до 2147483647).

```

C:\TP7_1\BIN\PROGRAM\WHILE\PEREVER.EXE
Введите число
1245879
перевертыш 9785421
  
```

- 13.6) Напишите программу, подсчитывающую количество цифр во введенном числе и разбивающую это число на разряды (т.е. выделить в нем отдельно сотни тысяч, тысячи, сотни, десятки и единицы).
Используйте тип данных **longint**.

```

C:\TP7_1\BIN\PROGRAM\WHILE\RAZRAD.EXE
Введите число
987654321
Разряд:1
Разряд:2
Разряд:3
Разряд:4
Разряд:5
Разряд:6
Разряд:7
Разряд:8
Разряд:9
Всего в вашем числе 9 разрядов
Повторить? <Y/N>:
  
```

- 13.7) Напишите программу для нахождения суммы всех цифр числа от 10 до 30000.

```

C:\TP7_1\BIN\PROGRAM\REPEAT\SUM_CIFR.EXE
Введите число не более 30000
12345
сумма цифр числа равна :15
  
```

```

C:\TP7_1\BIN\PROGRAM\REPEAT\SUM_CIFR.EXE
Введите число не более 30000
7823
сумма цифр числа равна :20
  
```

- 13.8) Напишите программу – текстовый калькулятор, умеющий выполнять 4 арифметических действия
Используйте цикл с условием для повторения вычислений (нужны дополнительно 2 символьные переменные типа **Char**: для Y/N и знаков действия «+», «-», «*» и «/»)



```

C:\TP7\BIN\PROGRAM\CALCULAT.EXE
КАЛЬКУЛЯТОР
Введите первое число
2356.4
Введите второе число
45
Введите знак действия + - * /
/
Результат: 52.364
Повторить еще раз <Y/N>?
y
Введите первое число
10785
Введите второе число
3
Введите знак действия + - * /
*
Результат: 32355.000
Повторить еще раз <Y/N>?
n
Работа калькулятора завершена
  
```


Задачи о делителях. Алгоритм Евклида. Решето Эратосфена

13.9) Напишите программу, которая выводила бы на экран **все** делители введенного пользователем натурального числа **n** (от 1 до 30000) с проверкой принадлежности числа заданному интервалу при вводе.

```

C:\TP7_1\BIN\PROGRAM\WHILEVSE_DELEXE
Введите целое число <от 1 до 30000>
0
Ошибка! Введите правильное число!
-50
Ошибка! Введите правильное число!
35000
Ошибка! Введите правильное число!
12345
Все делители этого числа:
1
3
5
15
823
2469
4115
12345
    
```

```

C:\TP7_1\BIN\PROGRAM\WHILEVSE_DELEXE
Введите целое число <от 1 до 30000>
594
Все делители этого числа:
1
2
3
6
9
11
18
22
27
33
36
54
66
99
198
297
594
    
```

```

C:\TP7_1\BIN\PROGRAM\WHILEVSE_DELEXE
Введите целое число <от 1 до 30000>
21789
Все делители этого числа:
1
3
9
27
81
269
807
2421
7263
21789
    
```

13.10) Напишите программу для нахождения **НОД** (наибольшего общего делителя) и **НОК** (наименьшего общего кратного) двух целых неотрицательных чисел.

Алгоритм Евклида – алгоритм нахождения наибольшего общего делителя (НОД) двух натуральных чисел. Существует несколько вариантов реализации этого алгоритма:

- **С вычитанием:** даны два числа **M** и **N**. Если они равны, то одно из них (например, **M**) и есть НОД. Если числа не равны, то нужно заменить большее из них разностью этих чисел и т. д. повторять до тех пор, пока не останется одно ненулевое число. Это и будет НОД.
- **С делением:** если числа **M** и **N** не равны, разделить большее число на меньшее, и заменить его на остаток от деления, повторять эти действия до тех пор, пока остаток не станет равным нулю.
- **Бинарный алгоритм:** Вначале считаем, что НОД=1. Если числа **M** и **N** – четные, то разделить каждое из них на 2, НОД умножить на 2. Пока оба числа **>0** повторять: если одно из чисел четное, то разделить его на 2, иначе заменить большее их разностью, выбрать из получившихся чисел **M** и **N** ненулевое и умножить на него НОД.

```

C:\TP7\BIN\PROGRAM\EVKLID_2.EXE
Алгоритм Евклида с делением
Введите первое число
123
Введите второе число
240
НОД равен 3
НОК равен 9840
    
```

```

C:\TP7\BIN\PROGRAM\EVKLID_1.EXE
Алгоритм Евклида с вычитанием
Введите первое число
60
Введите второе число
108
НОД равен 12
НОК равен 540
    
```

```

C:\TP7\BIN\PROGRAM\EVKLID_3.EXE
Бинарный Алгоритм Евклида
Введите первое число
186
Введите второе число
234
НОД равен 6
НОК равен 7254
    
```

13.11) Напишите программу, которая выводит на экран все простые числа в интервале от 1 до 1000.

«Решето Эратосфена» – алгоритм для поиска простых чисел в указанном интервале от 1 до **N**:

- выписать все целые числа от 1 до **N**
- **1 и 2 – первые простые числа;**
- вычеркнуть из списка все числа, кратные 2;
- **3 – следующее простое число;**
- из оставшихся чисел вычеркнуть все числа, кратные 3;
- числа, кратные 4 вычеркивать уже не надо (почему?)
- повторять вычеркивание до \sqrt{N} . Все оставшиеся в списке не вычеркнутые числа будут простыми.

```

C:\TP7\BIN\PROGRAM\VSEPRO-1.EXE
Таблица простых чисел до 1000
1 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47
53 59 61 67 71 73 79 83 89 97 101 103 107 109 113 127
131 137 139 149 151 157 163 167 173 179 181 191 193 197 199 211
223 227 229 233 239 241 251 257 263 269 271 277 281 283 293 307
311 313 317 331 337 347 349 353 359 367 373 379 383 389 397 401
409 419 421 431 433 439 443 449 457 461 463 467 479 487 491 499
503 509 521 523 541 547 557 563 569 571 577 587 593 599 601 607
613 617 619 631 641 643 647 653 659 661 673 677 683 691 701 709
719 727 733 739 743 751 757 761 769 773 787 797 809 811 821 823
827 829 839 853 857 859 863 877 881 883 887 907 911 919 929 937
941 947 953 967 971 977 983 991 997
    
```

13.12) Напишите программу для вывода всех совершенных чисел в интервале от 1 до 10000.

Совершенное число – натуральное число, равное сумме всех своих собственных делителей (отличных от самого числа).

Первое совершенное число: $6 = 1 + 2 + 3$

Второе совершенное число: $28 = 1 + 2 + 4 + 7 + 14$

По мере возрастания натуральных чисел, совершенные числа встречаются всё реже и реже.

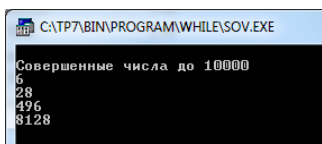
На сегодняшний день известно всего 46 совершенных чисел.

1 способ: В цикле нужно для каждого числа из указанного интервала проверять сумму его делителей и сравнивать ее с самим числом; если они равны, то это число совершенное.

2 способ: Евклидом, а впоследствии Л. Эйлером была выведена формула:

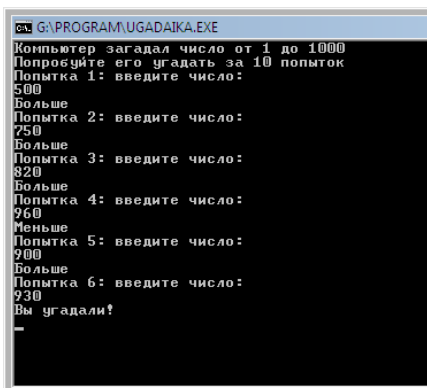
Если число $p = 1 + 2 + 4 + \dots + 2^n = 2^{n+1} - 1$ – простое, то число $s = 2^n * p$ – совершенное.

Таким образом, возникает задача поиска простых чисел, а затем нужно подставить их в формулу.

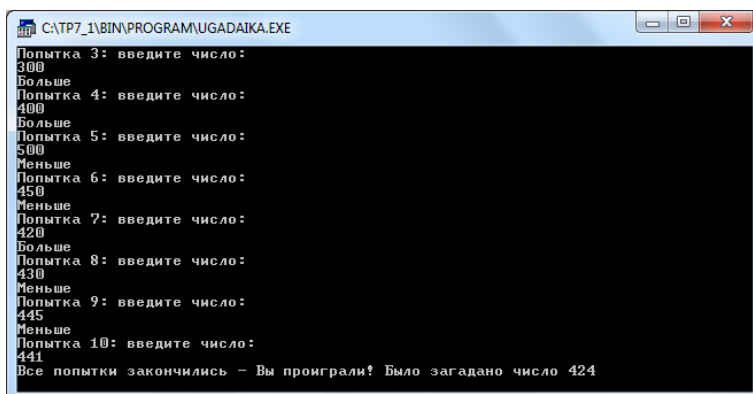


```
C:\P7_1\BIN\PROGRAM\WHILE\SOV.EXE
Совершенные числа до 10000
6
28
496
8128
```

13.13) Напишите программу-игру, в которой компьютер загадывает случайное число в интервале от 1 до 1000. Пользователю дается 10 попыток, чтобы угадать это число. После каждого введенного числа компьютер дает подсказку, больше оно или меньше, чем загаданное число. Если число не угадано после 10-ой попытки, программа выводит загаданное число.



```
G:\PROGRAM\UGADAIKA.EXE
Компьютер загадал число от 1 до 1000
Попробуйте его угадать за 10 попыток
Попытка 1: введите число:
500
Больше
Попытка 2: введите число:
750
Больше
Попытка 3: введите число:
820
Больше
Попытка 4: введите число:
960
Меньше
Попытка 5: введите число:
900
Больше
Попытка 6: введите число:
930
Вы угадали!
```



```
C:\P7_1\BIN\PROGRAM\UGADAIKA.EXE
Попытка 3: введите число:
300
Больше
Попытка 4: введите число:
400
Больше
Попытка 5: введите число:
500
Меньше
Попытка 6: введите число:
450
Меньше
Попытка 7: введите число:
420
Больше
Попытка 8: введите число:
430
Меньше
Попытка 9: введите число:
445
Меньше
Попытка 10: введите число:
441
Все попытки закончились - Вы проиграли! Было загадано число 424
```

Массивы

- 14.1) Напишите программу, которая вычисляет среднюю температуру воздуха за неделю (среднее арифметическое). Исходные данные должны вводиться пользователем. Используйте **2 массива**: для дней недели создайте массив в виде строчной константы и обычный массив для значений t:

Описание массивов:

```
File Edit Search Run Compile Debug Tools Options Window Help
PROGRAM\MASSIV_1\TEMPERAT.PAS
program temperatura;
const
day: array [1..7] of string = ('Пн', 'Вт', 'Ср', 'Чт', 'Пт', 'Сб', 'Вс');
var
t: array [1..7] of real;
s, sr: real;
i: integer;
begin
```

```
C:\TP7\BIN\PROGRAM\MASSIV_1\TEMPERAT.EXE
Введите температуру воздуха за неделю:
Пн: -8.5
Вт: -6.2
Ср: -4.8
Чт: -2
Пт: 0
Сб: 1.2
Вс: 0.5
Средняя температура за неделю: -2.8
```

- 14.2) Напишите программу, чтобы найти **минимальное** значение элемента и **его порядковый номер** в массиве из **50 случайных целых** чисел в диапазоне от **-100 до 100**. Сделайте контрольный вывод получившегося массива в строку.

```
C:\TP7\BIN\PROGRAM\MASSIV_1\32.EXE
Случайный массив:
Элементы:
65 53 15 -1 86 -20 -90 25 -76 85 54 -25 -20 91 45 20
-15 -32 60 -96 48 -37 -8 79 -28 -36 -22 72 -69 66 -32 -3
52 72 -12 -39 -97 -74 28 -14 -66 -96 -79 -98 -34 40 38 -87
75 -62
Минимальный элемент -98
Номер элемента 44
```

- 14.3) Напишите программу, которая из двух различных массивов, одинаковой размерности (например, по 5 чисел) формирует и выводит третий массив так, чтобы каждый элемент в нем вычислялся по формуле:
- $$c[i] = \frac{a[i] * b[i]}{2}$$

Элементы исходных массивов вводятся случайным образом из интервала [1,10)

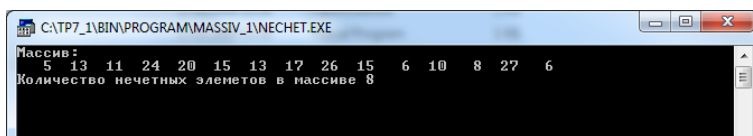
```
C:\TP7_1\BIN\PROGRAM\MASSIV_1\MASS_ABC.EXE
А-массив:
5 6 4 3 1
В-массив:
5 5 4 4 5
-----
Результат - массив C, где c[i] = (a[i]*b[i])/2: 12.5 15.0 8.0 6.0 2.5_
```

- 14.4) Напишите программу, которая проверяет, находится ли в массиве введенное с клавиатуры число, и какое оно по счету. Элементы массива и образец поиска вводятся пользователем.

```
C:\TP7\BIN\PROGRAM\000.EXE
Поиск в массиве.
Введите 10 целых чисел в одной строке через пробел: и нажмите <Enter>
10 20 30 40 50 60 70 80 90 100
Введите образец для поиска (целое число): 40
Совпадение с элементом номер 4.
```

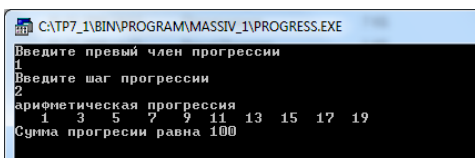
```
C:\TP7\BIN\PROGRAM\000.EXE
Поиск в массиве.
Введите 10 целых чисел в одной строке через пробел: и нажмите <Enter>
10 20 30 40 50 60 70 80 90 100
Введите образец для поиска (целое число): 55
Совпадений с образцом нет.
```

- 14.5) Напишите программу, которая формирует массив из 15 случайных целых чисел диапазоне от 1 до 30. Сделайте контрольный вывод массива. Подсчитайте количество нечетных элементов массива.

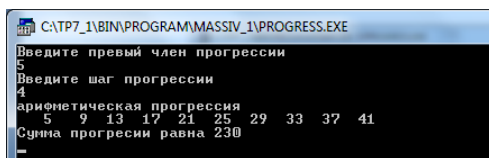


```
C:\TP7_1\BIN\PROGRAM\MASSIV_1\NECHET.EXE
Массив:
5 13 11 24 20 15 13 17 26 15 6 10 8 27 6
Количество нечетных элементов в массиве 8
```

- 14.6) Напишите программу, которая формирует массив из 10 целых чисел – членов арифметической прогрессии (пользователь задает значение первого члена и шаг арифметической прогрессии). Найдите сумму всех членов этой арифметической прогрессии.



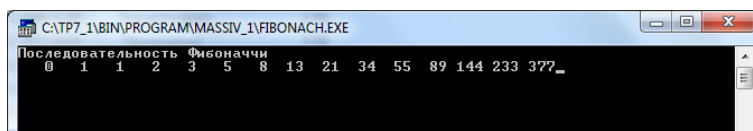
```
C:\TP7_1\BIN\PROGRAM\MASSIV_1\PROGRESS.EXE
Введите первый член прогрессии
1
Введите шаг прогрессии
2
арифметическая прогрессия
1 3 5 7 9 11 13 15 17 19
Сумма прогрессии равна 100
```



```
C:\TP7_1\BIN\PROGRAM\MASSIV_1\PROGRESS.EXE
Введите первый член прогрессии
5
Введите шаг прогрессии
4
арифметическая прогрессия
5 9 13 17 21 25 29 33 37 41
Сумма прогрессии равна 230
```

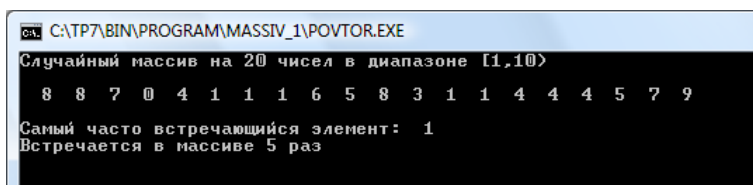
- 14.7) Создайте массив из 15 чисел и заполните его числами Фибоначчи.

Числа Фибоначчи – числовая последовательность, в которой каждое последующее число равно сумме двух предыдущих чисел: $F_1=0$, $F_2=1$, ... $F_n = F_{n-1} + F_{n-2}$



```
C:\TP7_1\BIN\PROGRAM\MASSIV_1\FIBONACH.EXE
Последовательность Фибоначчи
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377
```

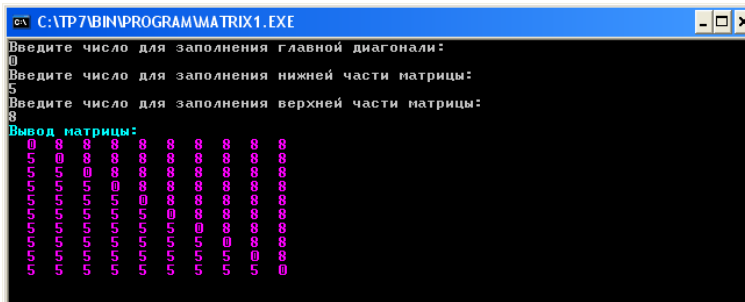
- 14.8) Напишите программу, которая формирует массив из 20 случайных целых чисел в диапазоне [0, 10). Найдите самый часто встречающийся элемент и подсчитайте сколько раз он повторяется в массиве.



```
C:\TP7_1\BIN\PROGRAM\MASSIV_1\POVTOР.EXE
Случайный массив на 20 чисел в диапазоне [1,10)
8 8 7 0 4 1 1 1 6 5 8 3 1 1 4 4 4 5 7 9
Самый часто встречающийся элемент: 1
Встречается в массиве 5 раз
```

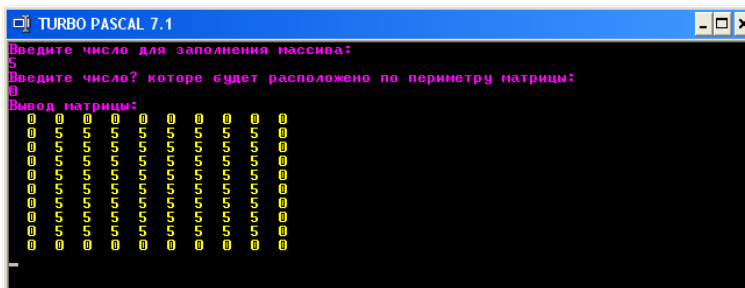
Двумерные массивы

- 15.1) Составьте программу, которая запрашивает у пользователя 3 числа, первое из этих чисел размещается на главной диагонали, второе – ниже, а третье – выше главной диагонали матрицы размерностью 10*10.



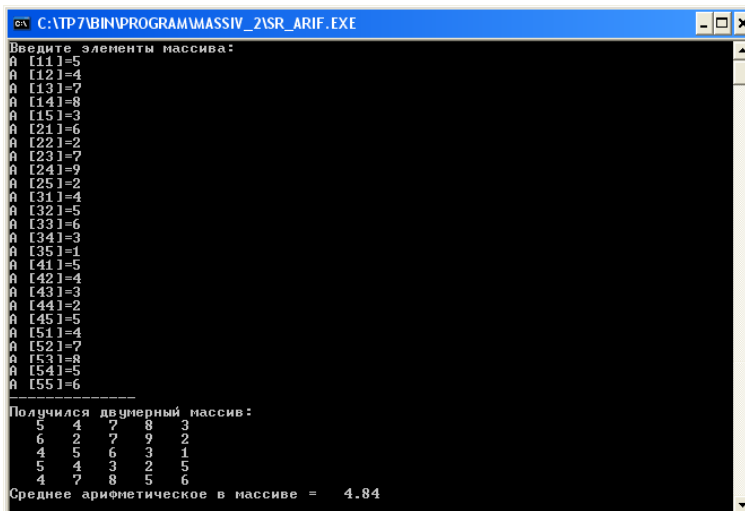
```
C:\TP7\BIN\PROGRAMMATRIX1.EXE
Введите число для заполнения главной диагонали:
0
Введите число для заполнения нижней части матрицы:
5
Введите число для заполнения верхней части матрицы:
8
Вывод матрицы:
0 8 8 8 8 8 8 8 8 8
5 0 8 8 8 8 8 8 8 8
5 5 0 8 8 8 8 8 8 8
5 5 5 0 8 8 8 8 8 8
5 5 5 5 0 8 8 8 8 8
5 5 5 5 5 0 8 8 8 8
5 5 5 5 5 5 0 8 8 8
5 5 5 5 5 5 5 0 8 8
5 5 5 5 5 5 5 5 0 8
5 5 5 5 5 5 5 5 5 0
```

- 15.2) Напишите программу, которая запрашивает у пользователя 2 числа, первым числом заполняется весь массив, а второе число располагается по периметру массива 10*10.



```
TURBO PASCAL 7.1
Введите число для заполнения массива:
5
Введите число? которое будет расположено по периметру матрицы:
0
Вывод матрицы:
0 0 0 0 0 0 0 0 0 0
0 5 5 5 5 5 5 5 5 0
0 5 5 5 5 5 5 5 5 0
0 5 5 5 5 5 5 5 5 0
0 5 5 5 5 5 5 5 5 0
0 5 5 5 5 5 5 5 5 0
0 5 5 5 5 5 5 5 5 0
0 5 5 5 5 5 5 5 5 0
0 5 5 5 5 5 5 5 5 0
0 0 0 0 0 0 0 0 0 0
```

- 15.3) Найдите среднее арифметическое в двумерном массиве 5*5 элементов. Элементы массива вводятся вручную. Сделайте контрольный вывод двумерного массива.

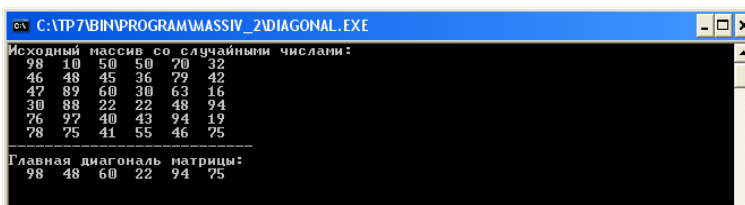


```
C:\TP7\BIN\PROGRAMMASSIV_2VSR_ARIF.EXE
Введите элементы массива:
A [11]=5
A [12]=4
A [13]=7
A [14]=8
A [15]=3
A [21]=6
A [22]=2
A [23]=7
A [24]=9
A [25]=2
A [31]=4
A [32]=5
A [33]=6
A [34]=3
A [35]=1
A [41]=5
A [42]=4
A [43]=3
A [44]=2
A [45]=5
A [51]=4
A [52]=7
A [53]=8
A [54]=5
A [55]=6

Получился двумерный массив:
5 4 7 8 3
6 2 7 9 2
4 5 6 3 1
5 4 3 2 5
4 7 8 5 6

Среднее арифметическое в массиве = 4.84
```

- 15.4) Заполните массив 6*6 случайными двузначными числами (от 10 до 99). Выведите на экран элементы главной диагонали матрицы в строку.



```
C:\TP7\BIN\PROGRAMMASSIV_2DIAGONAL.EXE
Исходный массив со случайными числами:
98 10 50 50 70 32
46 48 45 36 79 42
47 89 60 30 63 16
30 88 22 22 48 94
76 97 40 43 94 19
78 75 41 55 46 75

Главная диагональ матрицы:
98 48 60 22 94 75
```

- 15.5) Составьте программу, которая запрашивает у пользователя 2 числа, одно из которых заполняет всю матрицу, а второе размещается на главной и побочной диагонали матрицы 10×10 .

```

C:\TP7\BIN\PROGRAM\MASSIV_2\MATRIX2.EXE
Введите число для заполнения матрицы:
1
Введите число для заполнения главной и побочной диагонали:
0
Вывод матрицы:
0 1 1 1 1 1 1 1 1 0
1 0 1 1 1 1 1 1 0 1
1 1 0 1 1 1 1 0 1 1
1 1 1 0 1 1 0 1 1 1
1 1 1 1 0 0 1 1 1 1
1 1 1 1 0 0 0 1 1 1
1 1 0 1 1 1 1 0 1 1
1 1 0 1 1 1 1 1 0 1
1 0 1 1 1 1 1 1 1 0
0 1 1 1 1 1 1 1 1 0
  
```

- 15.6) В двумерном массиве 3×3 вычислить произведение всех элементов массива. Введите элементы массива вручную. Сделайте контрольный вывод массива в виде таблицы. При выборе типа данных для переменной – произведения, учтите, что это будет достаточно большое число! Выберите тип данных **longint** (от -2147483648 до 2147483648)

```

C:\TP7\BIN\PROGRAM\MASSIV_2\PROIZV.EXE
Введите элементы массива:
a [11]=2
a [12]=8
a [13]=6
a [21]=3
a [22]=4
a [23]=7
a [31]=6
a [32]=9
a [33]=2
-----
Получился двумерный массив:
2 8 6
3 4 7
6 9 2
-----
Произведение всех элементов двумерного массива: 870912
  
```

- 15.7) Составьте программу для заполнения двумерного массива 5×7 случайными числами от 1 до 99. Найдите максимальный элемент массива и его индекс.

```

C:\TP7\BIN\PROGRAM\MASSIV_2\MAX_POIS.EXE
Исходный массив 5*7 со случайными числами от 1 до 99
98 12 46 77 52 61 43
97 56 23 99 83 13 73
97 27 43 16 6 88 73
72 81 70 90 38 81 35
36 24 23 75 36 57 34
-----
Максимальное число в массиве: 99
Номер элемента: a[2,4]
  
```

- 15.8) Составьте программу для подсчета четных элементов в двумерном массиве размерностью 4×4 . Массив заполните случайными однозначными числами. Сделайте контрольный вывод исходного массива.

```

C:\TP7_1\BIN\PROGRAM\MASSIV_2\CHETNOST.EXE
Исходный массив
2 9 1 4
4 1 1 2
1 3 3 5
2 8 8 4
-----
Четных элементов 8
  
```

- 15.9) Составьте программу, печатающую таблицу Пифагора (таблицу умножения), в которой в 1-ой строке записаны единицы, а в 1-ом столбце – десятки, столбцы и первая строка отделены друг от друга декоративными линиями из символов «-», «|» и «+», а числа в таблице являются результатом умножения числа десятков и единиц.

```

C:\TP7_1\BIN\PROGRAM\MASSIV_2\PIPHAGOR.EXE
Таблица Пифагора
1 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
+---+---+---+---+---+---+---+---+---+
2 | 4 | 6 | 8 | 10 | 12 | 14 | 16 | 18 |
3 | 6 | 9 | 12 | 15 | 18 | 21 | 24 | 27 |
4 | 8 | 12 | 16 | 20 | 24 | 28 | 32 | 36 |
5 | 10 | 15 | 20 | 25 | 30 | 35 | 40 | 45 |
6 | 12 | 18 | 24 | 30 | 36 | 42 | 48 | 54 |
7 | 14 | 21 | 28 | 35 | 42 | 49 | 56 | 63 |
8 | 16 | 24 | 32 | 40 | 48 | 56 | 64 | 72 |
9 | 18 | 27 | 36 | 45 | 54 | 63 | 72 | 81 |
  
```

Работа с массивами

16.1) Напишите программу, создающую массив из 10 случайных чисел в интервале от 1 до 50 и печатающий его в обратном порядке (любым способом).



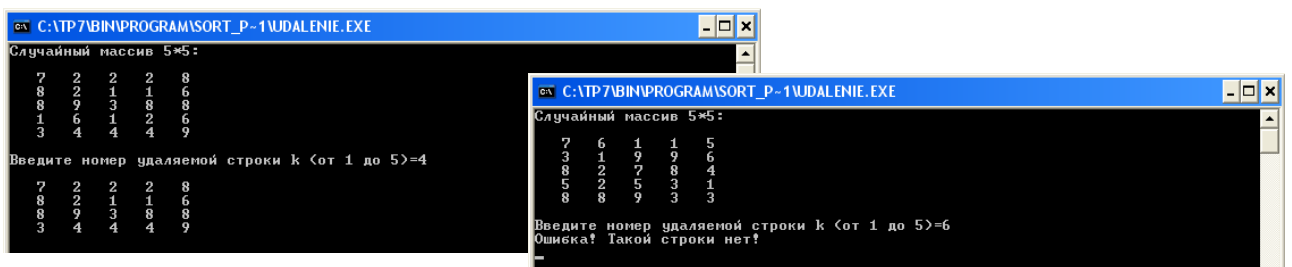
```
cmd C:\TP7\BIN\PROGRAM\MASSIV_1\PEREVERN.EXE
Исходный массив:
49 34 9 3 8 37 6 20 39 18
-----
Перевернутый массив:
18 39 20 6 37 8 3 9 34 49
```

16.2) Заполнить вручную одномерный массив из 10 чисел и выполните реверс отдельно для 1-ой и 2-ой половины массива. Сделайте в программе контрольный вывод исходного массива.



```
cmd C:\TP7\BIN\PROGRAM\PEREST-1\INVERSIA.EXE
Введите элементы массива
N1 : 1
N2 : 3
N3 : 5
N4 : 7
N5 : 9
N6 : 2
N7 : 4
N8 : 8
N9 : 0
N10 : 6
Исходный массив:
1 3 5 7 9 2 4 8 0 6
Инверсия двух половинок массива:
9 7 5 3 1 6 0 8 4 2
```

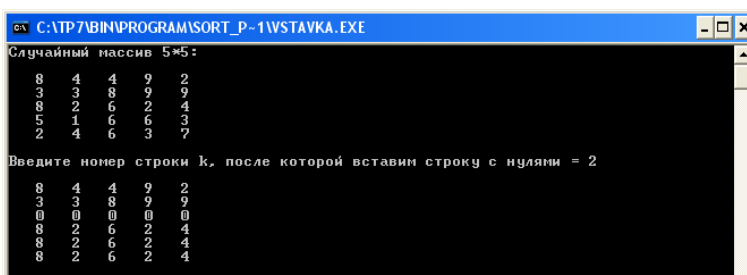
16.3) Создайте двумерный массив 5*5 со случайными числами от 1 до 9 и удалите в нем строку с номером k (номер удаляемой строки вводит пользователь). В случае отсутствия такой строки, выведите сообщение об этом и выйдите из программы.



```
cmd C:\TP7\BIN\PROGRAM\SORT_P-1\UDALENIE.EXE
Случайный массив 5*5:
7 2 2 2 8
8 2 1 1 6
8 9 3 8 8
1 6 1 2 6
3 4 4 4 9
Введите номер удаляемой строки k (от 1 до 5)=4
7 2 2 2 8
8 2 1 1 6
8 9 3 8 8
3 4 4 4 9

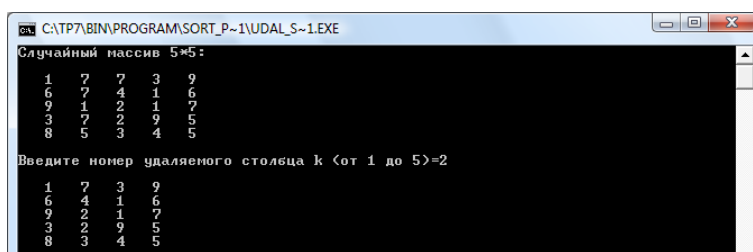
cmd C:\TP7\BIN\PROGRAM\SORT_P-1\UDALENIE.EXE
Случайный массив 5*5:
7 6 1 1 5
3 1 9 9 6
8 2 7 8 4
5 2 5 3 1
8 8 9 3 3
Введите номер удаляемой строки k (от 1 до 5)=6
Ошибка! Такой строки нет!
```

16.4) Создайте двумерный массив 5*5 со случайными числами от 1 до 9 и вставьте в него строку после k-той, заполнив ее нулями.



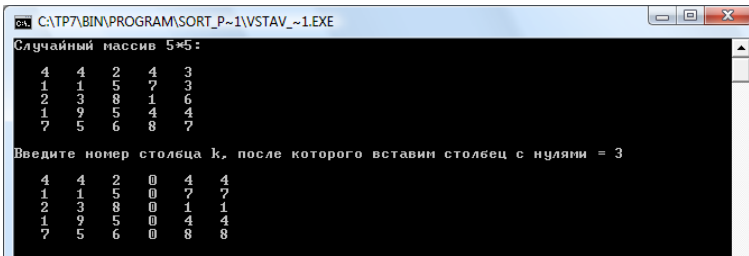
```
cmd C:\TP7\BIN\PROGRAM\SORT_P-1\VSTAVKA.EXE
Случайный массив 5*5:
8 4 4 9 2
3 3 8 9 9
8 2 6 2 4
5 1 6 6 3
2 4 6 3 7
Введите номер строки k, после которой вставим строку с нулями = 2
8 4 4 9 2
3 3 8 9 9
0 0 0 0 0
8 2 6 2 4
8 2 6 2 4
```

16.5) Напишите программу для удаления столбца с номером k (используйте алгоритмы из предыдущей задачи, только изменяйте размерность и производите сдвиг столбцов j).



```
cmd C:\TP7\BIN\PROGRAM\SORT_P-1\UDAL_S-1.EXE
Случайный массив 5*5:
1 7 7 3 9
6 7 4 1 6
9 1 2 1 7
3 7 2 9 5
8 5 3 4 5
Введите номер удаляемого столбца k (от 1 до 5)=2
1 7 3 9
6 4 1 6
9 2 1 7
3 2 9 5
8 3 4 5
```


16.6) Напишите программу для вставки столбца с нулями после указанного.

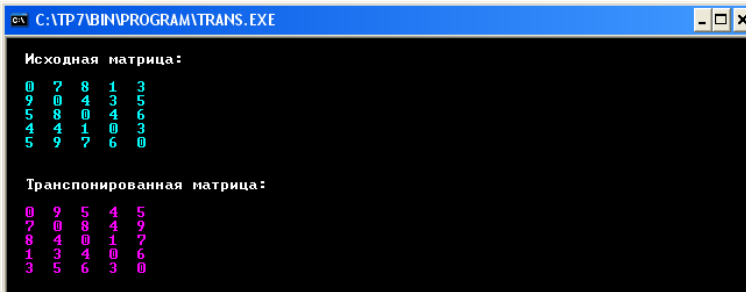


```
C:\TP7\BIN\PROGRAM\SORT_P~1\VSTAV_~1.EXE
Случайный массив 5*5:
4 4 2 4 3
1 1 5 7 3
2 3 8 1 6
1 9 5 4 4
7 5 6 8 7

Введите номер столбца k, после которого вставим столбец с нулями = 3
4 4 2 0 4 4
1 1 5 0 7 7
2 3 8 0 1 1
1 9 5 0 4 4
7 5 6 0 8 8
```

16.7) Напишите программу для транспонирования* квадратной матрицы 5*5. Все элементы на главной диагонали матрицы заполните нулями, нижняя и верхняя части заполняются случайным образом числами из диапазона от 1 до 9. Сделайте контрольный вывод исходной матрицы, а затем транспонированной.

Транспонирование матрицы – это операция, при которой элементы матрицы отражаются **симметрично относительно главной диагонали**, при этом элементы меняются местами по правилу: $A_{ij}^T = A_{ji}$



```
C:\TP7\BIN\PROGRAM\TRANS.EXE
Исходная матрица:
0 7 8 1 3
9 0 4 3 5
5 8 0 4 6
4 4 1 0 3
5 9 7 6 0

Транспонированная матрица:
0 9 5 4 5
7 0 8 4 9
8 4 0 1 7
1 3 4 0 6
3 5 6 3 0
```

Сортировка массивов

17.1) Напишите программу для формирования массива из 20 случайных чисел в диапазоне от 1 до 100 и выполните сортировку одного и того же массива **по возрастанию и по убыванию**.

```
TURBO PASCAL 7.1
Массив до сортировки:
 36 40 2 99 83 49 71 5 17 97 5 7 100 23 37 22
 36 46 52 30
Массив после сортировки по ВОЗРАСТАНИЮ:
 2 5 5 7 17 22 23 30 36 36 37 40 46 49 52 71
 83 97 99 100
Массив после сортировки по УБЫВАНИЮ:
100 99 97 83 71 52 49 46 40 37 36 36 30 23 22 17
 7 5 5 2_
```

17.2) Напишите программу для формирования массива из 15 случайных чисел в диапазоне от 1 до 100 и выполните сортировку массива **по возрастанию методом «пузырька»**.

```
TURBO PASCAL 7.1
Массив до сортировки:
 39 65 28 6 61 29 41 26 3 69 82 30 56 27 79
Массив после сортировки по возрастанию МЕТОДОМ "ПУЗЫРЬКА" :
 3 6 26 27 28 29 30 39 41 56 61 65 69 79 82_
```

17.3) Напишите программу для сортировки **методом выбора** массива из 50 случайных чисел в диапазоне от 1 до 300 **по убыванию**.

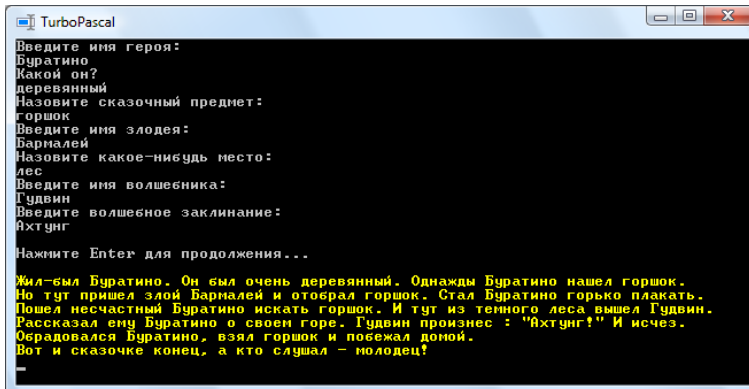
```
TURBO PASCAL 7.1
Массив до сортировки:
157 127 257 27 285 274 57 229 53 118 133 68 70 47 219 13
219 13 238 155 226 7 271 231 197 103 148 210 92 288 81 58
175 76 210 61 58 70 5 162 20 228 65 146 275 141 205 189
218 131
Массив после сортировки по убыванию МЕТОДОМ ВЫБОРА:
288 285 275 274 271 257 238 231 229 228 226 219 219 218 210 210
205 197 189 175 162 157 155 148 146 141 133 131 127 118 103 92
81 76 70 70 68 65 61 58 58 57 53 47 27 20 13 13
 7 5_
```

17.4) Напишите программу для сортировки **методом вставок** массива из 25 случайных чисел **по убыванию**.

```
TURBO PASCAL 7.1
Исходный массив:
64 2 100 88 6 17 78 56 50 24 45 12 97 9 96 92 18 63 40 2
 5 27 7 84 13
Массив после сортировки по убыванию МЕТОДОМ ВСТАВОК:
100 97 96 92 88 84 78 64 63 56 50 45 40 27 24 18 17 13 12 9
 7 6 5 2 2_
```

Символы и строки

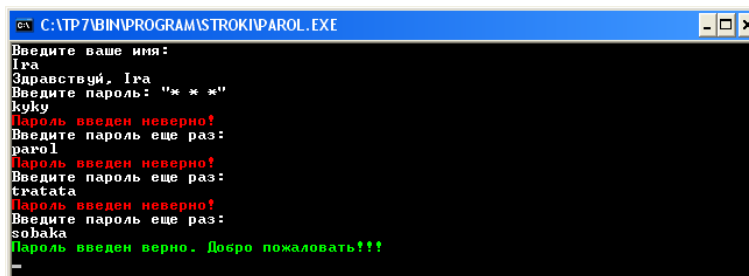
18.1) Составьте шуточную программу «Генератор сказок». Запросите у пользователя имена сказочных героев, предметов и т.д. Используйте эти данные для подстановки в текст-шаблон в качестве строковых переменных.



```
TurboPascal
Введите имя героя:
Буратино
Какой он?
деревянный
Назовите сказочный предмет:
горшок
Введите имя злодея:
Барналей
Назовите какое-нибудь место:
лес
Введите имя волшебника:
Гудвин
Введите волшебное заклинание:
Ахтунг
Нажмите Enter для продолжения...

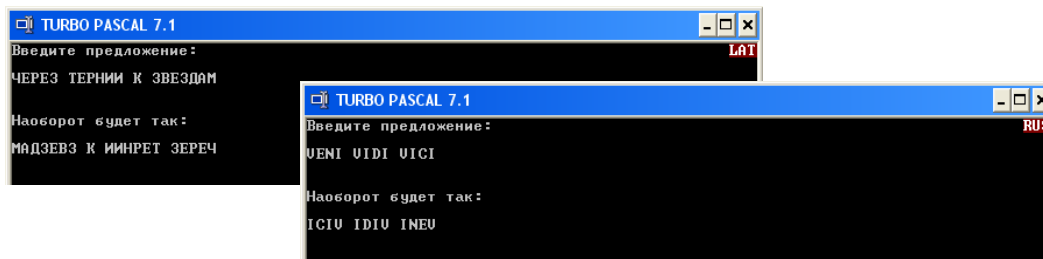
Жил-был Буратино. Он был очень деревянный. Однажды Буратино нашел горшок.
Но тут пришел злой Барналей и отобрал горшок. Стал Буратино горько плакать.
Пошел несчастный Буратино искать горшок. И тут из темного леса вышел Гудвин.
Рассказал ему Буратино о своей горе. Гудвин произнес : "Ахтунг!" И исчез.
Оправдался Буратино, взял горшок и поехал домой.
Вот и сказочке конец, а кто слушал - молодец!
```

18.2) Составьте программу, которая запрашивает имя пользователя, обращается к нему по имени и запрашивает пароль до тех пор, пока он не будет введен правильно.



```
C:\TP7\BIN\PROGRAM\STROKI\PAROL.EXE
Введите ваше имя:
Ira
Здравствуй, Ira
Введите пароль: '* * *'
kuku
Пароль введен неверно!
Введите пароль еще раз:
razo1
Пароль введен неверно!
Введите пароль еще раз:
treatata
Пароль введен неверно!
Введите пароль еще раз:
sobaka
Пароль введен верно. Добро пожаловать!!!
```

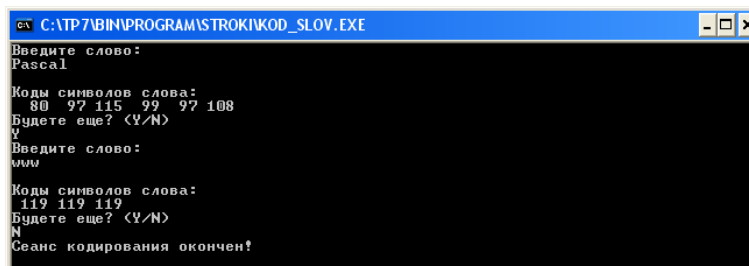
18.3) Составьте программу так, чтобы она переворачивала введенное слово или предложение наоборот.



```
TURBO PASCAL 7.1
Введите предложение:
LAI
ЧЕРЕЗ ТЕРНИИ К ЗВЕЗДАМ
Наоборот будет так:
ИАИ

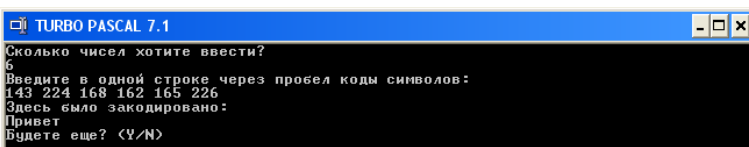
TURBO PASCAL 7.1
Введите предложение:
VENI VIDI VICI
Наоборот будет так:
ICIV IDIV INEV
```

18.4) Составьте программу, так чтобы она запрашивала у пользователя строку (или слово) и выводила коды использованных в слове символов (проверьте коды символов по таблице ASCII) Используйте цикл для повторения запроса на ввод слов.



```
C:\TP7\BIN\PROGRAM\STROKI\KOD_SLOV.EXE
Введите слово:
Pascal
Коды символов слова:
80 97 115 92 97 108
Будете еще? <Y/N>
Y
Введите слово:
mm
Коды символов слова:
119 119 119
Будете еще? <Y/N>
N
Сеанс кодирования окончен!
```

18.5) Составьте программу, которая запрашивает коды символов, и выводит текст.



```
TURBO PASCAL 7.1
Сколько чисел хотите ввести?
6
Введите в одной строке через пробел коды символов:
143 224 168 162 165 226
Здесь было закодировано:
Привет
Будете еще? <Y/N>
```

- 18.6) Составьте программу, которая запрашивает у пользователя текст (строку символов) и две разных буквы. Программа должна заменить в тексте все символы, совпадающие с первой буквой на вторую букву. Если искомого символа в тексте не обнаружится, то должно быть выведено сообщение об этом.

```
TURBO PASCAL 7.1
Введите текст
молоко
Введите 1-ю букву (образец поиска):
а
Введите 2-ю букву (чем заменить):
о
молоко

TURBO PASCAL 7.1
Введите текст
Привет
Введите 1-ю букву (образец поиска):
ж
Введите 2-ю букву (чем заменить):
з
Букв ж нет в этом тексте
```

- 18.7) Составить программу, которая в заданном тексте меняет все прописные английские буквы – на строчные.

```
TURBO PASCAL 7.1
Введите строку текста:
TO KNOW EVERYTHING IS TO KNOW NOTHING
to know everything is to know nothing
```

- 18.8) Составьте программу, которая запрашивает строку символов (слово), а затем выводит все буквы этого слова в алфавитном порядке.

```
C:\TP7\BIN\PROGRAM\STROKIVALFAVIT.EXE
Введите слово:
четыр
Буквы по алфавиту:
eqrtwy_

TURBO PASCAL 7.1
Введите слово:
информатика
Буквы по алфавиту:
ааиикннортф_
```

- 18.9) Составьте программу для перевода римских чисел от 1 до 3999 в десятичную систему счисления.

Для проверки:

DXCIV = 594

MCXCVI = 1196

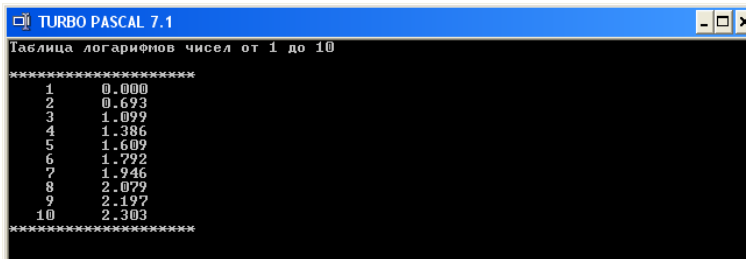
CDLXXVIII = 478

```
C:\TP7\BIN\PROGRAM\RIM.EXE
Введите число:
MDCCLXVII
Ваше число = 1747
```

Процедуры

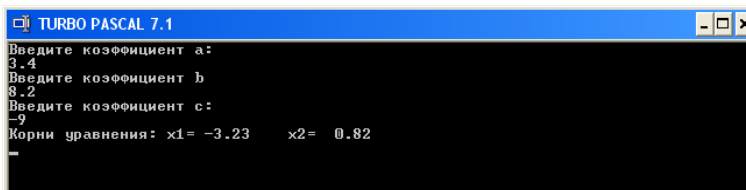
19.1) Составьте программу для вывода таблицы натуральных логарифмов чисел от 1 до 10.

Для оформления таблицы составьте **процедуру без параметров** вывода строки из 20 звездочек или каких-нибудь других символов.



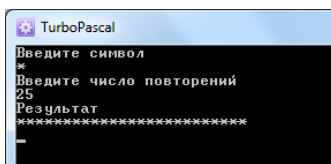
```
TURBO PASCAL 7.1
Таблица логарифмов чисел от 1 до 10
*****
1      0.000
2      0.693
3      1.099
4      1.386
5      1.609
6      1.792
7      1.946
8      2.079
9      2.197
10     2.303
*****
```

19.2) Напишите или отредактируйте программу к задаче 7.9 для решения квадратных уравнений, которая включает в себя **процедуру с параметрами** для вычисления дискриминанта.

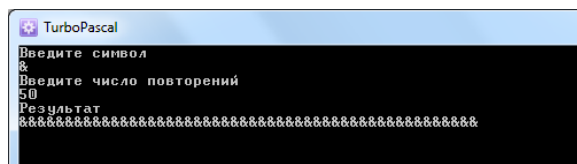


```
TURBO PASCAL 7.1
Введите коэффициент a:
3.4
Введите коэффициент b:
8.2
Введите коэффициент c:
-9
Корни уравнения: x1= -3.23   x2=  0.82
_
```

19.3) Составьте программу, в которой используется процедура с параметрами, которая выводит строку, состоящую из одинаковых символов. Длина строки и символ, который выводится, являются параметрами процедуры.

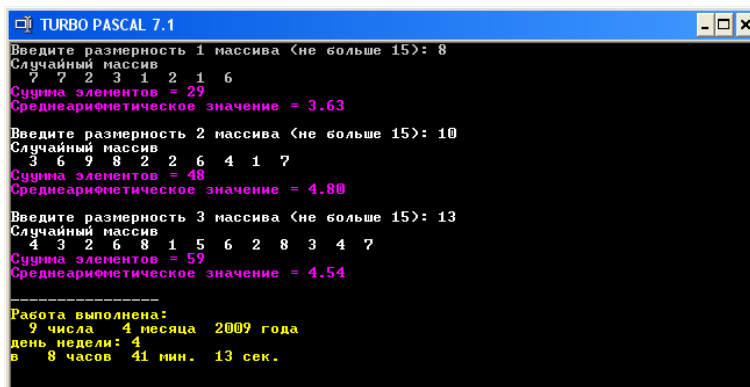


```
TurboPascal
Введите символ
*
Введите число повторений
25
Результат
*****
_
```



```
TurboPascal
Введите символ
*
Введите число повторений
50
Результат
*****
_
```

19.4) Даны 3 различных массива разной размерности, каждый из не более 15 случайных целых чисел (основная программа должна запрашивать размерность каждого массива). В каждом массиве найдите сумму элементов и их среднее арифметическое значение. Подключите в программе модули **Crt** и **Dos**. Используйте цвет. Нахождение суммы и среднего арифметического оформите в виде **процедуры с параметрами**. Используйте процедуру без параметров **DataTime** для подписи даты и времени в конце программы.



```
TURBO PASCAL 7.1
Введите размерность 1 массива (не больше 15): 8
Случайный массив
7 7 2 3 1 2 1 6
Сумма элементов = 29
Среднеарифметическое значение = 3.63

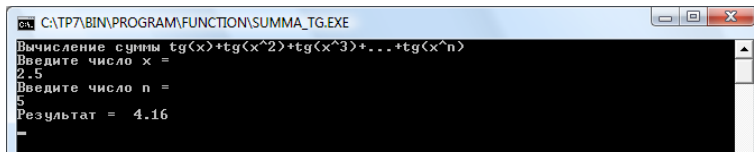
Введите размерность 2 массива (не больше 15): 10
Случайный массив
3 6 9 8 2 2 6 4 1 7
Сумма элементов = 48
Среднеарифметическое значение = 4.80

Введите размерность 3 массива (не больше 15): 13
Случайный массив
4 3 2 6 8 1 5 6 2 8 3 4 7
Сумма элементов = 59
Среднеарифметическое значение = 4.54

-----
Работа выполнена:
9 числа 4 месяца 2009 года
день недели: 4
в 8 часов 41 мин. 13 сек.
```

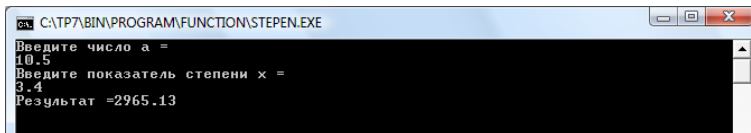
Функции

- 20.1) Напишите программу для вычисления выражения $z = \text{tg}(x) + \text{tg}(x^2) + \text{tg}(x^3) + \dots + \text{tg}(x^n)$ Используйте свою функцию для тангенса.



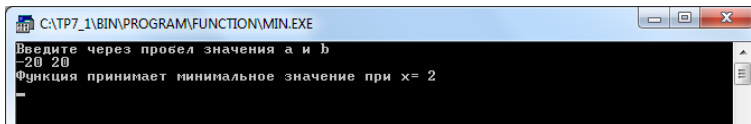
```
C:\TP7_1\BIN\PROGRAM\FUNCTION\SUMMA_TG.EXE
Вычисление суммы tg(x)+tg(x^2)+tg(x^3)+...+tg(x^n)
Введите число x =
2.5
Введите число n =
5
Результат = 4.16
```

- 20.2) Напишите программу с функцией возведения числа **a** в любую степень **b**. Для того, чтобы возвести число **a** в степень **b** ($x=a^b$), в паскале используют запись $x:=\text{exp}(b*\text{Ln}(a))$;



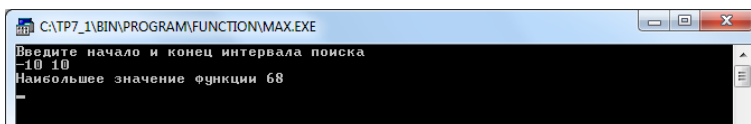
```
C:\TP7_1\BIN\PROGRAM\FUNCTION\STEPEN.EXE
Введите число a =
10.5
Введите показатель степени x =
3.4
Результат =2965.13
```

- 20.3) Напишите программу с функцией для нахождения значения аргумента алгебраической функции $F(x) = 4*(x-1)*(x-3)$ на интервале **[a, b]**, при котором эта функция достигает минимума. Значения **a** и **b** вводит пользователь, шаг просмотра 1.



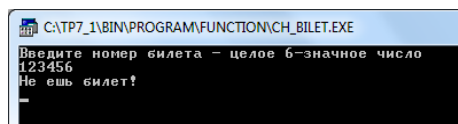
```
C:\TP7_1\BIN\PROGRAM\FUNCTION\MIN.EXE
Введите через пробел значения a и b
-20 20
Функция принимает минимальное значение при x= 2
```

- 20.4) Напишите программу с функцией для поиска наибольшего значения алгебраической функции $F(x) = x^2 + 4*x + 8$ на интервале **[a, b]**, просматривая значения от **a** до **b** с шагом 1. Значения **a** и **b** вводит пользователь.

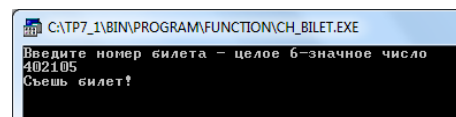


```
C:\TP7_1\BIN\PROGRAM\FUNCTION\MAX.EXE
Введите начало и конец интервала поиска
-10 10
Наибольшее значение функции 68
```

- 20.5) Напишите программу с функцией для проверки счастливый номер билета или нет. Пользователь вводит номер билета – целое 6-значное число (счастливым считается билет, в котором сумма 3 левых цифр совпадает с суммой 3 правых цифр). Функция должна считать сумму 3 цифр.



```
C:\TP7_1\BIN\PROGRAM\FUNCTION\CH_BILET.EXE
Введите номер билета – целое 6-значное число
123456
Не ешь билет!
```



```
C:\TP7_1\BIN\PROGRAM\FUNCTION\CH_BILET.EXE
Введите номер билета – целое 6-значное число
402105
Съешь билет!
```