

Министерство образования Российской Федерации  
Пермский государственный технический университет

В.П. Гладков

## Знаешь ли ты Паскаль?

Утверждено Редакционно-издательским советом  
университета в качестве учебного пособия

Пермь 2001

УДК 681.3  
Г 52

Рецензенты:

доктор технических наук, профессор *В.П. Первадчук*  
(Пермский государственный технический университет),  
кандидат педагогических наук, доцент *А.П. Шестаков*  
(Пермский государственный педагогический университет)

**Гладков В.П.**

Г 52            Знаешь ли ты Паскаль?: Учеб. пособие / Перм. гос. техн.  
ун-т, Пермь, 2001. – 109 с.

Приводятся задания, позволяющие проверить знания Паскаля, обратить внимание на сложные понятия языка, научиться читать написанные на Паскале программы. Многие фрагменты алгоритмов, приведенные в книге, можно применять как технические приемы при реальном программировании. Используемая методика позволяет быстро и глубоко освоить программирование на алгоритмическом языке.

Для студентов, преподавателей и любителей Паскаля.

УДК 681.3

©Пермский государственный  
технический университет, 2001  
©Гладков В.П., 2001

## Введение

Алгоритмический язык - это искусственный язык, предназначенный для точной и однозначной записи алгоритмов. Алгоритмический язык Паскаль был разработан профессором Цюрихского Федерального технологического института (ETH) Никлаусом Виртом. Предварительное сообщение появилось в 1968 году. С тех пор Паскаль становился все более и более популярен, причем не только при обучении принципам программирования, для чего вначале он предназначался, но и как средство создания сложного программного обеспечения, как исходный материал для построения новых алгоритмических языков (Модула, Ада).

Такой авторитет языка Паскаль объясняется прежде всего тем, что в нем отражены фундаментальные и наиболее важные концепции алгоритмов в очевидной, естественной и легко воспринимаемой форме. Объем языка не велик. Все конструкции хорошо продуманы и в настоящее время используются не только для записи алгоритмов, но и в научных и научно-популярных статьях.

Язык Паскаль позволяет применять регулярные процедуры проектирования программ, например, структурное, модульное или инструментальное, объектно-ориентированное программирование.

Язык Паскаль имеет хороший выбор структур данных, что позволяет формулировать более простые и эффективные алгоритмы.

Язык Паскаль способствует написанию надежных программ, которые легко отлаживать и сопровождать. Это достигается многими средствами, например, требованием описывать все переменные. Хотя это ведет к избыточности программ, но зато позволяет обнаруживать ошибки в употреблении переменных в программе на этапе трансляции, а не во время исполнения.

Язык Паскаль проектировался с учетом легкости построения транслятора для него. Неслучайно самый быстрый в мире компилятор построен именно для Паскаля (Turbo Pascal фирмы Borland International).

Язык Паскаль проектировался Н.Виртом с учетом легкости обучения. У обучаемого, начинающего изучение программирования с Паскаля, вырабатывается простое и ясное алгоритмическое мышление, он скорее постигает суть программирования и начинает писать надежные и эффективные программы.

Цель данной работы – помочь начинающему в изучении Паскаля. По нашему мнению, решающим условием успеха является большое желание изучить Паскаль. Вторым условием является чтение соответствующей литературы. Третье условие - проверка и опробование изученных конструкций на компьютере.

Первое условие целиком зависит от обучаемого. Что касается литературы, то ее по языку Паскаль много (см. список литературы). На наш взгляд, не следует стремиться искать какую-то особую книгу (наиболее понятную, популярную, наиболее полную, доступную для понимания и т.п.), так как любая из имеющихся книг имеет как достоинства, так и недостатки. Нужно взять любую книгу и проработать ее от корки до корки. Здесь не случайно

употреблен термин проработать, так как чтение технической литературы вещь особенная, подразумевающая не просто чтение и осмысливание прочитанного, но и запоминание образцов, фрагментов программ и их восстановление по памяти, обязательную трассировку (ручное исполнение программы) всех фрагментов программ, выполнение упражнений, построение всех возможных вариантов записей изучаемых конструкций, создание памяток, куда выписываются синтаксис конструкций, правила их исполнения, перечень предопределенных констант, встроенных функций и т.п., т.е. всего того, что сразу запомнить трудно, но что постоянно требуется и для программирования, и для понимания и уяснения последующего материала.

Как показывает опыт обучения, если с чтением дела обстоят более или менее благополучно, то остальное чаще всего опускается. Поэтому начинающие программисты много времени (и своего, и машинного) тратят на уточнение правил выполнения различных конструкций языка непосредственно за компьютером. Задача этой книги замедлить процесс чтения и тем улучшить понимание и запоминание прочитанного. Больше того, как неудивительно, этот подход позволяет изучить Паскаль быстрее.

Для этого здесь собраны известные, опубликованные в литературе, задававшиеся на экзаменах в Пермском государственном техническом университете и придуманные автором вопросы о языке Паскаль. Они сгруппированы по темам. Отвечая на данные вопросы, обучаемый обращает внимание на те вещи и обстоятельства, которым обычно не придается значения и которые как раз и являются очень важными и значимыми. Ответы на эти вопросы позволяют обучаемому глубже понять язык Паскаль и избежать многих, часто дорогостоящих ошибок.

Здесь предлагается следующая методика работы:

1. Выбрать любую книгу по языку Паскаль.
2. Прочитать в книге о некоторой конструкции языка Паскаль.
3. Ответить на вопросы из данной работы по изученной теме.
4. Проверить свой ответ с помощью компьютера.
6. Повторять пункты 2, 3, 4 до тех пор, пока не будет изучен весь Паскаль.

Известно, что изучение алгоритмического языка будет более успешным, если использовать при этом компьютер. Однако здесь следует учесть то, что разные реализации Паскаля отличаются друг от друга. Поэтому при работе с конкретной реализацией Паскаля обязательно знакомство с документацией по данной реализации.

В данной работе используется ядро языка Паскаль, описанное Н.Виртом [6, 7], рассмотрены некоторые существенные дополнения, имеющиеся в Turbo Pascal фирмы Borland International [40, 41, 42].

Книга рассчитана на начинающих непрофессионалов: школьников, студентов, инженеров, всех, кто желает научиться записывать алгоритмы для своих целей.

В заключение заметим, что предлагаемый метод требует известной усидчивости и старательности. Но в то же время, ответив на все вопросы и добравшись до конца, Вы не только будете готовы к сдаче экзамена по языку,

но и почувствуете свою преданность Паскалю. А, как верно заметил Д. Алкок [3], «лекарства от любви к Паскалю еще не найдено».

## 1. Общие понятия

*Информатика* - отрасль науки, изучающая структуру и общие свойства информации, вопросы, связанные с её сбором, хранением, поиском, переработкой, преобразованием, распространением и использованием в разных сферах деятельности.

- 1.1. Что такое информация?
- 1.2. Почему нельзя дать строгое определение понятию «информация»?
- 1.3. В каком соотношении находятся понятия «информация» и «данные»?
- 1.4. Что такое алгоритм? Каковы его свойства?
- 1.5. Какие формы представления алгоритмов существуют? Приведите один и тот же алгоритм в разных формах записи.
- 1.6. Чем алгоритм отличается от программы?
- 1.7. Какова роль алгоритмов в современном мире?
- 1.8. Приведите примеры последовательностей действий, в которых отсутствует свойство детерминированности (определённости).
- 1.9. Приведите примеры последовательностей действий, в которых отсутствует свойство массовости.
- 1.10. Приведите примеры последовательности действий, в которых отсутствует свойство конечности.
- 1.11. Какой алгоритм называется линейным?
- 1.12. Какой алгоритм называется ветвящимся?
- 1.13. Какой алгоритм называется циклическим?
- 1.14. Какой алгоритм называется вспомогательным? Для каких целей используются вспомогательные алгоритмы?
- 1.15. Что такое «структура»? Приведите синонимы этого понятия.
- 1.16. Что такое «исполнитель»?
- 1.17. Что входит в систему команд исполнителя? Что произойдет, если при исполнении алгоритма обнаружится команда, которой нет в системе команд исполнителя данного алгоритма?
- 1.18. Представьте себе, что Вы исполнитель. Какие алгоритмы Вы можете исполнять? Приведите пример такого алгоритма. Из каких команд (предписаний) он состоит?
- 1.19. Что такое компьютер?
- 1.20. Какова роль компьютеров в современном мире?
- 1.21. Приведите синонимы к слову «компьютер».
- 1.22. Укажите основные узлы компьютера, без которых он работать не будет.
- 1.23. Какую работу выполняют оперативная память и центральный процессор компьютера при исполнении программы?
- 1.24. Сколько раз за время выполнения команды программы процессор обращается к оперативной памяти?
- 1.25. Для чего нужна операционная система?

1.26. Приведите примеры задач, при решении которых компьютер сможет получить решение без обращения к дополнительному программному обеспечению.

1.27. Для чего предназначены устройства ввода?

1.28. Для чего предназначены устройства вывода?

1.29. Для чего предназначена внешняя память компьютера?

1.30. В чем сходство и различия оперативной и внешней памяти компьютера?

1.31. Можно ли дисковод назвать устройством ввода? А вывода?

1.32. За счет чего обеспечивается автоматическая работа компьютера?

1.33. Каковы преимущества человека перед компьютером?

1.34. Каковы преимущества компьютера перед человеком?

1.35. Как Вы думаете, какие задачи, с которыми человек справится, компьютер никогда не сможет решить?

1.36. Как Вы думаете, какие задачи человек не сможет решить без компьютера?

1.37. Что такое алгоритмический язык?

1.38. Что нужно для того, чтобы компьютер «понимал» алгоритмический язык?

1.39. Что такое транслятор, интерпретатор, компилятор?

1.40. Какие ошибки позволяет исправлять транслятор: синтаксические, семантические или логические?

1.41. Из каких этапов складывается процесс построения программы?

1.42. Что такое математическая модель задачи?

1.43. Зачем нужна отладка?

1.44. Что такое тестирование программы?

1.45. Что такое тест?

1.46. Какой набор тестов считается минимально необходимым?

1.47. Что такое ветвь алгоритма?

1.48. Можно ли по блок-схеме алгоритма узнать, сколько в нем веток? Путей? Как это сделать?

1.49. Выберите верные, на Ваш взгляд, высказывания. Объясните, почему они верны.

1) «В программировании опыт - великий учитель»;

2) «Не бойтесь экспериментировать - измените часть программы, чтобы увидеть, к чему это приведет»;

3) «Чем больше Вы сделаете самостоятельно, тем большему научитесь»;

4) «Нужно решать все задачи на компьютере».

1.50. Сколько двоичных цифр(бит) достаточно, чтобы закодировать порядковые номера дней недели? А порядковые номера дней года?

1.51. Чем программирование отличается от алгоритмизации?

1.52. Напишите термины информатики, которые начинаются буквой «п».

1.53. Напишите термины информатики, которые оканчиваются буквой «я».

## 2. Типы данных (константы, переменные, операции, функции)

- 2.1. Что такое алфавит? Какие символы входят в алфавит Паскаля?
- 2.2. Совпадает ли алфавит Паскаля со всеми символами кодовой таблицы ASCII?
- 2.3. Что называют зарезервированными словами? Какова их роль?
- 2.4. Что называют предопределенными словами? Какова их роль?
- 2.5. Приведите примеры символов Паскаля, которые являются словами в обычном понимании.
- 2.6. Для чего в программах используются комментарии?
- 2.7. Являются ли комбинации символов:  $\leq$ ,  $:=$ ,  $\geq$ ,  $\langle \rangle$  символами Паскаля?
- 2.8. Что такое «имя»? Как оно устроено и где используется?
- 2.9. Приведите синонимы «имени».
- 2.10. Какие имена записаны неверно? Почему?
  - а) time ,      д) alpha ,      и) a2 ,      н) IBM\_PC ,      с) endbegin ,
  - б) a' ,          е) a<sub>5</sub> ,          к) \_3d ,          о) t2.5 ,          т) DoRoGa ,
  - в) альфа ,      ж) 2a ,          л) begin ,      п) pascal ,      у) ASU-PGTU ,
  - г) a.b.c ,      з) z<sup>2</sup> ,          м) a/b ,      р) час\_пик ,      ф) computer .
- 2.11. Что такое «литерал»? Для чего он используется?
- 2.12. На какие два больших класса можно разделить все данные в Паскаль-программе?
- 2.13. Чем отличаются переменные от констант?
- 2.14. Как представляются константы в программе?
- 2.15. Как представляются переменные в программе?
- 2.16. Может ли константа иметь имя?
  - а) если может, то как оно задается и для каких целей?
  - б) если не может, то почему?
- 2.17. Что такое тип данных?
- 2.18. Какие типы данных имеются в Паскале?
- 2.19. Что такое порядковый тип? Какие функции для него разрешены?
- 2.20. Что такое тип *integer*?
- 2.21. Как записываются константы целого типа?
- 2.22. Как записываются переменные целого типа?
- 2.23. Какие операции над целыми операндами возвращают целый результат?
- 2.24. Какие операции над целыми операндами возвращают вещественный результат?
- 2.25. Какие операции над целыми операндами возвращают логический результат?
- 2.26. Перечислите функции, которые разрешены для целого аргумента.
- 2.27. Перечислите функции, которые возвращают целый результат.
- 2.28. Какие операции запрещены для данных целого типа?
- 2.29. Какие операции разрешены над вещественными данными?
- 2.30. Перечислите функции, разрешенные для данных вещественного типа.
- 2.31. Перечислите операции и функции, разрешенные для данных символьного типа.

- 2.32. Можно ли по записи константы в Паскале определить её тип?
- 2.33. Можно ли по записи переменной в Паскале определить её тип?
- 2.34. Описать две переменные целого и одну вещественного типа, если для построения имён этих переменных разрешается использовать только два символа  $x$  и  $y$ .
- 2.35. Описать пять разных переменных, используя только одну букву  $z$ .
- 2.36. Сколько разных имен длиной четыре можно построить, используя только три символа  $b$ ,  $0$  и  $1$ ?
- 2.37. Сколько разных имен длиной не более трёх можно построить, используя три символа  $a$ ,  $b$  и  $0$ ?
- 2.38. Сколько разных трехзначных чисел можно образовать, используя цифры  $0$ ,  $5$ ,  $9$ ?
- 2.39. Записать константу  $24.5$  на алгоритмическом языке двумя разными способами.
- 2.40. Может ли цифра  $5$  использоваться в алгоритмических языках не только как константа или её часть?
- 2.41. Пусть  $min$  - минимальное значение типа  $real$ , а  $max$  - максимальное значение этого типа. Какое число меньше -  $min$  или  $1/max$ ?
- 2.42. Помогает ли объявление типа переменной сэкономить память компьютера?
- 2.43. Можно ли в Паскале обойтись без констант?
- 2.44. Может ли число, правильно записанное на Паскале, оканчиваться точкой?
- 2.45. Может ли число, правильно записанное на Паскале, начинаться с точки?
- 2.46. Может ли правильная запись числа на Паскале содержать более одной точки?
- 2.47. Может ли правильная запись числа на Паскале содержать более одного знака  $+$  или  $-$ ?
- 2.50. Из букв  $a$  и  $b$  запишите четырнадцать имен, состоящих не более чем из трёх символов.
- 2.51. Запишите на Паскале числа, являющиеся значениями следующих выражений:
- 1)  $1987$ ,
  - 2)  $0,0000000097$ ,
  - 3)  $-10^{-6}$ ,
  - 4)  $721 \cdot 10^7$ ,
  - 5)  $-1272321,01$ ,
  - 6)  $-31171,14 \cdot 10^{-3}$ ,
  - 7)  $\sqrt{3}$ ,
  - 8)  $\frac{2}{5}$ ,
  - 9)  $1\frac{1}{2}$ ,



10)  $-17^{0,25}$  ,

11) LXIV .

2.52. Возможна ли на Паскале запись? Почему?

```
Const tri = 3;  
  drei = +3;  
  three = tri;  
  four = three+1;  
  seven = four+drei; .
```

2.53. Найдите и объясните ошибки в следующих описаниях констант:

```
CoNsT dwa = 2;  
  zwei = dwa + two;  
  two = 4;  
  funf = tri * two;  
  tri = dwa + dwa + 1;  
  two = funf div dwa; .
```

2.54. Найдите и объясните ошибки в описаниях переменных целого типа.

```
var a,b,c : integer;  
  s{путь} v{скорость} t{время} : integer;  
  b,c,d, : integer;  
  const : integer;  
  dwa+tri : integer;  
  x,,y ; integer;  
  p,r,s,t,u : integer  
  w : integer;  
  q{описание переменной}integer; .
```

2.55. Вычислите и проверьте делением уголком:

20 div 6 ,	20 mod 6 ,
20 div 4 ,	20 mod 4 ,
2 div 5 ,	2 mod 5 ,
123 div 0 ,	3.0 mod 0 ,
-8 div (-3) ,	-8 mod (-3) ,
-8 div 3 ,	-8 mod 3 ,
8 div (-3) ,	8 mod (-3) ,
8 div 3 ,	8 mod 3 .

2.56. Операция «число div 10» позволяет отбросить последнюю цифру числа. Операция «число mod 10» позволяет достать из числа последнюю цифру. Например,  $123 \text{ div } 10 = 12$ ,  $123 \text{ mod } 10 = 3$ . Запишите, как определить вторую цифру (цифру десятков) целого числа?

2.57. Определите, что будет найдено при вычислении выражений?

$123 \text{ mod } 10 \text{ div } 10$  ,  $123 \text{ mod } 100 \text{ div } 10$  .

Чем неудачно второе решение?

2.58. Найдите третью цифру числа двумя способами.

2.59. Опишите целую переменную  $k$  и запишите выражение, вычисляющее сумму её цифр, считая, что переменная содержит трехзначное значение.

2.60. Опишите целую переменную  $k$  и запишите выражение, вычисляющее сумму её цифр, считая, что переменная содержит четырехзначное значение.

2.61. Опишите целую переменную  $k$  и запишите выражение, вычисляющее сумму её цифр, считая, что переменная содержит произвольное значение.

2.62. Найдите первую цифру дробной части вещественного числа.

2.63. Найдите сумму трех цифр дробной части вещественного числа.

2.64. Какие числа можно получить в ответе при вычислении выражения?

1)  $x \bmod 5 = ?$ ,

2)  $x \operatorname{div} 5 = 8$ ,

3)  $50 \operatorname{div} x = 7$ ,

4)  $50 \bmod x = 7$ .

2.65. С какими значениями  $x$  следующие равенства будут правильными?

1)  $x \operatorname{div} 5 = x \bmod 5$ ,

2)  $20 \operatorname{div} x = 20 \bmod x$ .

2.66. Идет  $k$ -я секунда суток. Сколько это составит полных часов, минут и секунд?

2.67. Определите операции  $\operatorname{div}$  и  $\bmod$  через другие операции и стандартные функции.

2.68. Вычислите значения выражений:

$\operatorname{succ}(3) + \operatorname{pred}(5)$ ,

$\operatorname{pred}(3) + \operatorname{succ}(5)$ ,

$\operatorname{succ}(a) + \operatorname{pred}(4)$ ,

$\operatorname{succ}(\operatorname{succ}(b)) + \operatorname{pred}(\operatorname{pred}(c))$ ,

$\operatorname{succ}(\operatorname{pred}(\operatorname{succ}(6))) + \operatorname{pred}(\operatorname{succ}(\operatorname{pred}(10)))$ ,

$\operatorname{succ}(-\maxint)$ ;

$\operatorname{pred}(\maxint)$ .

2.69. Запишите константы Паскаля в традиционной записи:

$-12.3\text{E}+2$ ,  $-0.8\text{e}-6$ ,  $1\text{E}3$ ,  $+1\text{e}-6$ ,  $+0.5\text{E}+03$ ,  $-0.666\text{e}+066$ .

2.70. Вычислите значения выражений:

$\operatorname{trunc}(2.8)$ ,  $\operatorname{round}(2.8)$ ,

$\operatorname{trunc}(2.1)$ ,  $\operatorname{round}(2.1)$ ,

$\operatorname{trunc}(-1.6)$ ,  $\operatorname{round}(1.6)$ ,

$\operatorname{round}(0.5)$ ,  $\operatorname{round}(-0.5)$ .

2.71. Запишите выражение на Паскале:

1)  $x^5$ , 2)  $\cos^7 x^4$ , 3)  $\log_6 \frac{x}{5}$ , 4)  $\sqrt[3]{1+x}$ ,

5)  $|x^{-3}|$ , 6)  $\arcsin x$ , 7)  $\sin 8^\circ$ , 8)  $2^{1+x}$ .

2.72. Какие значения относятся к типу *char*?

2.73. Объясните справедливость формулы:  $\langle \text{число} \rangle = \operatorname{ord}(\langle \text{цифра} \rangle) - \operatorname{ord}('0')$ .

2.74. Почему для типа *char* справедливы равенства:  $\operatorname{pred}(c) = \operatorname{chr}(\operatorname{ord}(c) - 1)$ ,  $\operatorname{succ}(c) = \operatorname{chr}(\operatorname{ord}(c) + 1)$ ?

2.75. Скажите одним словом, в чем сходство следующих записей: 101, 10.1, '101'.

2.76. Скажите одним словом, в чем отличие следующих записей: 101, 10.1, '101'.

2.77. Как компилятор определяет тип константы? А тип переменной?

2.78. Перечислите, для каких целей используются имена в Паскале.

### 3. Арифметические выражения

3.1. Какие записи являются выражениями в Паскале:

1) 5, 2)  $\sin(x)$ , 3)  $y$ , 4)  $5+y*\sin(x)$ .

3.2. Почему нельзя опускать знак умножения?

3.3. Почему аргументы функции должны быть заключены в круглые скобки?

3.4. Расставьте круглые скобки в следующих выражениях Паскаля, чтобы показать последовательность их выполнения:

1)  $a*b*2+3.456/y$ ,

2)  $a+b*c-d/f$ ,

3)  $a+b+c+d+e$ ,

4)  $-a+\sin(\text{abs}(3*(b+c))-(e-d)/(2*d*c))-14$ ,

5)  $b \bmod a+c \operatorname{div} b*s$ ,

6)  $-3+7 \operatorname{div} 2 \bmod 7/2-\text{trunc}(\cos(\exp(-1)))$ ,

7)  $a+b-c$ ,

8)  $a+b*c$ ,

9)  $a-b*c-d$ ,

10)  $(a-b)*(-k)$ .

3.5. Какие круглые скобки в приведенных выражениях можно снять, не изменив значения этих выражений:

1)  $(a+b)/c$ ,

8)  $((((a-b)-c)-d)-e)$ ,

2)  $a+(b/c)$ ,

9)  $a-(b-(c-(d-e)))$ ,

3)  $a/(a*b)$ ,

10)  $((a-b)-(c-d))-e$ ,

4)  $a+(-3)*(s*d/r)-(a+25)$ ,

11)  $(a-(b-c))-(d-e)$ ,

5)  $(a+b)+((c+d)*2*e)$ ,

12)  $(a*b) \operatorname{div} c$ ,

6)  $(x1/x2)*y$ ,

13)  $b+(a-(c/3))$ ,

7)  $(\sqrt{p}*q)/r$ ,

14)  $(a*(b/(c*(d/(e*f))))))$ .

3.6. Сколько операций выполняется при вычислении выражений? Как сократить количество операций?

1)  $(a+1/2)*(b+7/10)-3/4$ ,

2)  $ax^4+bx^3+cx^2+dx+e$ .

3.7. Перечислите правила записи выражения.

3.8. Перечислите правила вычисления значения выражения.

3.9. Можно ли определить бесконечно длинное арифметическое выражение?

3.10. Для чего нужны скобки в арифметическом выражении? Можно ли обойтись без скобок при записи произвольного выражения? Можно ли обойтись без скобок при записи выражения:  $(2+a)*(3-b)$ ?

3.11. Какую операцию забыл выполнить программист при записи выражения:  $a*\sin(\text{abs}(-3*(s-k))+5.0)/(2*a*b)$ ?

3.12. Запишите выражения на Паскале:

1)  $\frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(x-a)^2}{2\sigma^2}}$ ,

2)  $\sqrt[5]{\frac{\pi}{8} \sqrt{\frac{\alpha^2 + \beta^2 - \beta}{\alpha^2 + \beta^2}}}$ ,

3)  $x_{m_k} y^{m_{n_k}}$ ,

4)  $2^{1+\sqrt{x}}$ ,

$$5) \sqrt[5]{1+x^8},$$

$$6) \sqrt[x]{x^7+7^x},$$

$$7) 10^4\alpha - 3\frac{1}{5}\beta,$$

$$8) \frac{ab}{c} + \frac{d-c}{ab},$$

$$9) \log_2 \frac{x}{6} + \cos^5 \frac{(x+y^7)^3}{2a^3b^5} - \sin^4 x^7,$$

$$10) \frac{\arcsin x, \left( \frac{2b\sqrt{a}}{a+\sqrt{ab}} \right)^{-1} + \left( \frac{2a\sqrt{b}}{b+\sqrt{ab}} \right)^{-1}}{\left( \frac{a+\sqrt{ab}}{2ab} \right)^{-1} + \left( \frac{b+\sqrt{ab}}{2ab} \right)^{-1}}$$

11)

$$12) \frac{\left( \frac{(a+x)^3}{ax} - 4 \right) \cdot \left( \frac{(a-x)^{a+x}}{\frac{a}{x} + y} \right) \cdot (a^x - b^x)}{\left( a^{\sin^2 x^5} + y - ax^6 \right) \cdot \left( (a+x)^2 - ax \right) \cdot \left( (a-x)^{x+b} + ax \right)}.$$

3.13. Запишите в общепринятой форме следующие выражения Паскаля:

$$1) \sqrt{1+\sqrt{x}/\text{abs}(a+b*x)},$$

$$2) a+b/(c+d/(e+f/(g*h))),$$

$$3) (((a*x+b)*x+c)*x+d)*x+e,$$

$$4) (x+y)/a[1]*a[2]/(x-y),$$

$$5) \ln(\text{abs}(y-\sqrt{\text{abs}(x)})*(x-y/(z+\sqrt{x}/4))),$$

$$6) \sin(\sqrt{\text{abs}(\cos(x)/\exp(3)+\sin(y)/\cos(y-1))+\text{abs}(3*(d+2*s)+4))/2)/5,$$

$$7) \exp(0.25*\ln(x/y))+\exp(4*\ln(r)),$$

$$8) \sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{7}}}}}}+\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{8}}}}}}),$$

$$9) \sqrt{\sqrt{\sqrt{\sqrt{\sqrt{9}}}}} - \sqrt{\sqrt{\sqrt{\sqrt{16}}}} + \sqrt{\sqrt{\sqrt{\sqrt{25}}}} - \sqrt{\sqrt{\sqrt{\sqrt{36}}}},$$

$$10) 1*2*3*4*5*6/\sqrt{1*2*3}.$$

3.14. Всегда ли правильно записанное на Паскале выражение имеет смысл?

3.15. Проверьте записи выражений, установите выражения с семантическими ошибками:

$$1) \text{maxint}-\text{trunc}(\sqrt{16+a}),$$

$$2) 12+\text{maxint},$$

$$3) \text{random}(10)+a-\text{int}(a/b)*b,$$

$$4) \sin(\pi/4)+\sqrt{\cos(2*\pi/\sqrt{\sqrt{\sqrt{2}}})},$$

$$5) \text{trunc}(\text{round}(8)),$$

$$6) (2+3)*25 \text{ div } (9 \text{ mod } 3 \text{ div } 2 \text{ mod } 10),$$

$$7) \pi/\text{trunc}(\pi)+\text{round}(\pi)/\text{int}(\pi),$$

$$8) \text{chr}(\text{ord}('a'))+'a'.$$

3.16. Установите выражения с синтаксическими ошибками:

- 1)  $b + \sin(x)/k$ ,
- 2)  $\maxint - \sqrt{2} * d$ ,
- 3)  $a + b * (c - d * e)$ ,
- 4)  $(a + b) / 2 / a + b / (4 - \sqrt{2})$ ,
- 5)  $\sin(\text{abs}(\sqrt{x * (a + \cos(3 + 7 * x) + 5 * f)) - 2 * (\sin(2) + 2 - 2 / (a + b)))$ .

3.17. Определите значения выражений:

- 1)  $16 \text{ div } 4 * 2$ ,
- 2)  $16 + 4 * 2$ ,
- 3)  $16 \text{ div } (4 * 2)$ ,
- 4)  $1 + 19 \text{ mod } 5$ ,
- 5)  $(1 + 19) \text{ mod } 5$ ,
- 6)  $3 \text{ div } 10 + 25 \text{ mod } 5$ ,
- 7)  $1 + 25 \text{ div } 5 \text{ mod } 2$ ,
- 8)  $(1 + 25 \text{ div } 5) \text{ mod } 2$ ,
- 9)  $k \text{ mod } 10 + k \text{ div } 10 \text{ mod } 10 + k \text{ div } 10 \text{ div } 10 \text{ mod } 10$  (при  $k = 123, 987, 54$ ).

3.18. Определите тип выражения:

- 1)  $1 + 0.0$ ,
- 2)  $120 / 6$ ,
- 3)  $\sqrt{4}$ ,
- 4)  $\sqrt{4.0}$ ,
- 5)  $\sqrt{16}$ ,
- 6)  $\sqrt{4.0}$ ,
- 7)  $\sin(0)$ ,
- 8)  $\text{succ}(-1)$ ,
- 9)  $\text{trunc}(\pi)$ ,
- 10)  $\sqrt{2} + \text{trunc}(12 / 7)$ .

3.19. Какие из следующих выражений синтаксически правильны (совместимы по типу)? Определите типы этих выражений, исходя из следующих описаний переменных:

`var x, y, z: real; i, j, k: integer;`

- 1)  $x + y * i$ ;
- 2)  $i \text{ mod } (j + y)$ ;
- 3)  $i + j - k$ ;
- 4)  $i \text{ div } j + x$ ;
- 5)  $(x + y) < (i + j)$ ;
- 6)  $k - \text{trunc}(x * i)$ ;
- 7)  $i * x + j * y$ ;
- 8)  $(x < y) \text{ and } (y < i)$ ;
- 9)  $x = i$ .

3.20. Определите, эквивалентны ли записи следующих выражений на Паскале и запишите их в общепринятой записи:

- 1)  $a + c * d / b + e / (f * g)$ ,
- 2)  $a + c * d / b + e / f / g$ .

3.21. Установите, что будет получено при выполнении каждого из следующих операторов присваивания при заданных описаниях и известном значении X:

```
var X, x1, x2, x3 : word;
{задаётся значение X}
x1 := X and 31;
X := X div 32;
x2 := X and 27;
X := X div 128;
x3 := X and 7.
```

3.22. Ниже приводятся выражения, записанные в общепринятой форме, справа от которых даны выражения, записанные по правилам Паскаля. Установите, в каких случаях выражение, стоящее справа, является эквивалентом своего левого соседа, или укажите ошибки в записи:

1) $\frac{b + \sqrt{b^2 - 4ac}}{2a}$	$(b + \text{sqrt}(\text{sqr}(b) - 4 * a * c)) / 2 * a;$
2) $\cos^2 x$	$\text{sqrt}(\cos(x));$
3) $\frac{a}{c} \cdot \frac{b}{d}$	$ab/cd;$
4) $\frac{a}{c} \cdot \frac{b}{d}$	$((a/c) * b) / d;$
5) $\sin x + \cos \frac{y}{2}$	$\sin(x) + \cos(y/2);$
6) $\frac{x+1}{y+1}$	$x+1/y+1;$
7) $\sin \frac{x+y}{2}$	$\sin(x+y/2);$
8) $\frac{a + \sin(x)}{\sqrt{a^2 + x^2 + 1}}$	$a + \sin(x) / \text{sqrt}(\text{sqr}(a) + \text{sqr}(x) + 1);$
9) $\frac{2e^{- x+y }}{x^2 + y^2}$	$2 \exp(-\text{abs}(x+y)) / (\text{sqr}(x) + y * y);$
10) $\frac{\sin^2 x}{y_1 + 0,5}$	$\text{sqr}(\sin(x)) / (y1 + 0,5);$
11) $x \cdot 10^{-2}$	$x E - 2;$
12) $7,1 \cdot 10^{-z}$	$7.1 E - z;$
13) $(7,3 \cdot 10)^{-4}$	$7.3 E - 4;$
14) $3,6 \cdot 10^{-\sqrt{2}}$	$3.6 E - \text{sqrt}(2);$
15) $ \sin(1 + a_{i,k}) $	$\text{abs}(\sin(1 + a[i[k]]));$
16) $\frac{\sin x + \cos y}{2\sqrt{x^2 + y^2}}$	$\sin(x) * \cos(x) / 2 / \text{sqrt}(x^2 + y^2);$
17) $a + b \cdot [e^{-x} - \sin(x_k + 1)]$	$a + b * [\exp(-x) - \sin(x[k] + 1)];$
18) $\frac{a - \sin(x^2)}{ e^{2x} + \cos(a) }$	$(a - \sin(\text{sqr}(2))) /  \exp(2 * x) - \cos(a) ;$
19) $2 \cdot \cos\left(\frac{\pi}{3}\right)^{\sqrt{2}}$	$2 * \exp(\text{sqr}(2) * \ln(\cos(\pi/3))).$

#### 4. Логический тип данных. Логические выражения

4.1. Перечислите операции и функции, разрешенные над логическими данными.

4.2. Расставьте круглые скобки в следующих выражениях Паскаля, чтобы показать последовательность их выполнения:

- 1)  $a = b \text{ or } a < c \text{ and not } b \leq c * 2,$       2)  $a * 2 * c * \sin(x) - d = a * b / g.$

4.3. Какие круглые скобки в приведенных выражениях можно снять, не изменив значения этих выражений?

- 1)  $\text{not}(\text{not}(a < b) \text{ or } (c \geq d))$ ,
- 2)  $((a \text{ and } b) \text{ or } c) \text{ and } d$ ,
- 3)  $(((-a) < 25) \text{ and } (\text{not}(a \geq 16)))$ ,
- 4)  $((a = (b + c/4)) \text{ and } a) \text{ or } ((x \geq 3) \text{ and } \text{not}(x < 16))$ ,
- 5)  $(((((a \text{ and } b) \text{ or } c) \text{ and } d) \text{ or } e) \text{ and } f) \text{ or } g)$ .

4.4. Определите значения истинности следующих высказываний:

- 1) Париж расположен на Сене и  $2+3=5$ ;
- 2) 1 - простое число и 2 - простое число;
- 3) 1 - простое число или 2 - простое число;
- 4)  $2 \times 2 = 4$  или белые медведи живут в Африке;
- 5)  $2 \times 2 = 4$  и  $2 \times 2 = 5$  и  $2 \times 2 \geq 4$ .

4.5. Сформулируйте отрицания высказываний:

- 1) Луна - спутник Марса;
- 2) Все простые числа нечетны;
- 3) 32 не делится на 4;
- 4) Сегодня светит солнце и не дует ветер;
- 5) Я поступлю в ПГТУ или в ПГУ;
- 6) Я умею решать квадратные уравнения или прыгать по лужам;
- 7) Я сегодня пойду на лекцию по информатике и не пойду в деканат или я пойду в деканат и на дискотеку;
- 8) Солнце светит да не греет;
- 9) Сегодня мы пойдем в театр и не пойдем в кино.

4.6. Запишите на Паскале логические выражения истинные, если:

- 1)  $x$  лежит на отрезке от 0 до 1;
- 2)  $x$  лежит вне отрезка от 0 до 1;
- 3) трое друзей имеют карманные деньги;
- 4) ни один из трех друзей не имеет карманных денег;
- 5) только один из друзей имеет карманные деньги;
- 6) хотя бы один из трех друзей имеет карманные деньги;
- 7) карманных денег любых двух друзей из трех хватит для покупки трех порций мороженого;
- 8) общее количество денег у трех друзей - величина четная и не кратная 28;
- 9) рост Васи на 2 см больше роста Кати;
- 10) вес Лены на 12 кг меньше веса Толи;
- 11) Павел носит обувь на два размера больше, чем Александр и на три размера меньше, чем Шурик;
- 12)  $N$  рублей можно выдать только одними пятерками;
- 13) количество страниц в книге четное и кратно семи;
- 14) точка лежит на отрезке  $[a; b]$ , но не в его центре;
- 15) в целом трехзначном числе нет цифры 2;
- 16) числа  $a$  и  $b$  имеют разные знаки;
- 17) числа  $a$  и  $b$  имеют одинаковые знаки;

- 18) числа  $a$  и  $b$  оба равны нулю;
- 19) только одно из чисел  $a$  или  $b$  равно нулю;
- 20) хотя бы одно из чисел  $a$  или  $b$  равно нулю;
- 21) числа  $a$  и  $b$  или оба равны нулю, или имеют разные знаки;
- 22) числа  $a$  и  $b$  или оба равны  $c$ , или оба меньше  $c$ . Запишите отрицание

этого высказывания.

4.7. Запишите на русском языке, что обозначает условие

if  $(x+y+z)/3=x$   
 then оператор 1  
 else оператор 2.

4.8. Определите, чему равно  $a$  после выполнения фрагмента программы на Паскале:  $a:=0$ ; if  $a < 0$  then;  $a:=2$ ;

4.9. Прямая линия проходит через точки с координатами  $(3;0)$  и  $(0;9)$ .

1) Напишите два разных условия, истинных, когда обе точки  $A(x_1, y_1)$  и  $B(x_2, y_2)$  лежат по разные стороны от прямой.

2) Запишите, что обозначает условие

$$(-3*x_1+9-y_1)*(-3*x_2+9-y_2)<0 \quad (*)$$

3) Запишите, что обозначает условие

$$(-3*x_1+9-y_1<0) \text{ and } (-3*x_2+9-y_2>0) \text{ or } (-3*x_1+9-y_1>0) \text{ and } (-3*x_2+9-y_2<0) \quad (**)$$

4) Если условия  $(*)$  и  $(**)$  эквивалентны, то укажите достоинства и недостатки каждого из них. Если условия не эквивалентны, то укажите точки, которые дают разные результаты на этих условиях.

5) Запишите, что обозначает условие

$$(-3*x_1+9-y_1)*(-3*x_2+9-y_2)=0$$

6) Запишите, что обозначает условие

$$(-3*x_1+9-y_1)=(-3*x_2+9-y_2)$$

7) Запишите, что обозначает условие

$$(-3*x_1+9-y_1)*(-3*x_2+9-y_2)>0$$

8) Можно ли сказать, что условие  $(-3*x_1+9-y_1)+(-3*x_2+9-y_2)=0$  обозначает, что только одна из точек лежит на прямой?

9) Запишите, что обозначает условие

$$(-3*x_1+9-y_1)*(-3*x_2+9-y_2)\geq 0$$

10) Запишите, что обозначает условие

$$(-3*x_1+9-y_1)*(-3*x_2+9-y_2)\leq 0$$

11) Запишите, что обозначает условие

$$(-3*x_1+9-y_1)*(-3*x_2+9-y_2)<> 0$$

4.10. Равносильны ли утверждения  $x \bmod 2=0$  и  $\text{int}(x/2)=x/2$ ?

4.11. Какое из приведенных выражений вызовет ошибку компиляции и почему, если переменные имеют следующие значения:  $a=2$ ,  $b=2$ ,  $c=3$ ,  $d=\text{false}$ ,  $e=\text{true}$ ?

1)  $e \text{ and not } d \text{ xor } (a=b) \text{ or } (b=c)$ ,

2)  $e \text{ and not } d \text{ xor } a=b \text{ or } b=c$ .



4.12. Опишите типы переменных, которые используются в записи следующих выражений на Паскале:

- 1)  $(e=2)$  and  $s$  or  $(a=2.3)$ ,
- 2)  $(a=a+1)$  xor not  $d$  and not  $(s \leq 7)$ ,
- 3)  $a$  and not  $b$  and  $c$  and not  $d$ .

4.13. Установите, может ли правильная запись условия (логического выражения) на Паскале иметь вид:

- 1)  $(a > a)$  and  $(a > a)$ ;
- 2)  $(a > a)$  and  $(a < a)$ ;
- 3)  $(a > a)$  or  $(a > a)$ ;
- 4)  $(a > a)$  or  $(a < a)$ ;
- 5)  $(a \geq a)$  and  $(a \geq a)$ ;
- 6)  $(a \geq a)$  or  $(a \geq a)$ ;
- 7)  $(a \geq a)$  and  $(a \leq a)$ ;
- 8)  $(a \geq a)$  or  $(a \leq a)$ ;
- 9) true and false;
- 10) true and true;
- 11) false and false;
- 12) true or false;
- 13) true or true;
- 14) false or false;
- 15) true or  $(1/0 > 0)$ ;
- 16)  $(1/0 > 0)$  or true.

4.14. Установите, истинны или ложны следующие условия:

- 1) not  $((6 < 4) \text{ and } (4 > 8))$ ;
- 2)  $(a \leq a+1) \text{ or } (12-3 > 0)$ ;
- 3)  $((2 \leq 2) \text{ and } (3 \geq 3)) \text{ or } (15 > 25)$ ;
- 4)  $(x > 5) \text{ and } (x < x+1)$ ;
- 5)  $a$  and not  $a$ ;
- 6)  $a$  or not  $a$ ;
- 7)  $a$  and true;
- 8)  $a$  and false;
- 9)  $a$  or true;
- 10)  $a$  or false;
- 11) not not  $a$ ;
- 12)  $a$  or  $a$ ;
- 13)  $a$  and  $a$ ;
- 14) not  $a$  and not  $b = \text{not } (a \text{ or } b)$ ;
- 15) not  $a$  or not  $b = \text{not } (a \text{ and } b)$ ;
- 16)  $a \text{ and } (b \text{ or } c) = (a \text{ and } b) \text{ or } (a \text{ and } c)$ ;
- 17)  $a \text{ or } (b \text{ and } c) = (a \text{ or } b) \text{ and } (a \text{ or } c)$ ;
- 18)  $a \leq b = \text{not } a \text{ or } b$ ;
- 19)  $a \text{ and } b = (a < \text{true}) < b$ ;
- 20) not  $a = a < \text{true}$ .

4.15. Объясните ошибки в записях:

- 1) 1 and 0;
- 2) true+false;
- 3) true<0;
- 4) not 2=5;
- 5) x>0 or y=4;
- 6) not not b or or d;
- 7) true>'true'.

4.16. Вычислите значения выражений:

- |                               |                       |
|-------------------------------|-----------------------|
| 1) false<true;                |                       |
| 2) ord(false)=1;              |                       |
| 3) pred(true);                |                       |
| 4) ord(succ(false))>0;        |                       |
| 5) not(pred(c) or (ord(c)=1)) | при c=true;           |
| 6) (p<true)=(q=false)         | при p=q=true;         |
| 7) a and b>a or b             | при a=false, b=true;  |
| 8) x<2                        | при x=1;              |
| 9) not a or odd(x)            | при a=true, x=1;      |
| 10) ord(a)<>x                 | при a=true, x=1.      |
| 11) not odd(n)                | при n=0;              |
| 12) t and (p mod 3=0)         | при t=true, p=101010; |
| 13) (x*y<>0) and (y>x)        | при x=2, y=1;         |
| 14) (x*y<>0) or (y>x)         | при x=2, y=1;         |
| 15) a or not b                | при a=false, b=true.  |

4.17. Нарисуйте на плоскости (x,y) область, в которой и только в которой истинно условие:

- 1) (y>=x) and (y+x>=0) and (y<1);
- 2) (sqr(x)+sqr(y)<1) or (y>0) and (abs(x)<=1);
- 3) (trunc(y)=0) and (round(x)=0);
- 4) (abs(x)<=1)>(abs(y)>=1);
- 5) (sqr(x)+sqr(y)<=4)=(y<=x).

4.18. Запишите на русском языке условие, заданное на Паскале:

- 1) (x=y) and (x=z);
- 2) odd(n)=odd(k);
- 3) ord(a)+ord(b)=100;
- 4) (a<=x) and (x<b);
- 5) abs(x-0.5)<3;
- 6) (x=y) and (x<>z) and (y<>z) or (x=z) and (x<>y) and (y<>z) or (z=y) and (x<>z) and (y<>x);
- 7) k mod 10 =5;
- 8) m mod n =0;
- 9) x\*y<0.

4.19. Определите тип и, если возможно, значения следующих выражений:

var p,q,r,s:Boolean; k:integer;

- 1) sqr(2);
- 2) sqr(2.0);

- 3) `ord('я')-ord('a');`
- 4) `trunc(-99.9);`
- 5) `-round(99.9);`
- 6) `-round(-99.9);`
- 7) `not(p and q) = not (not p and not q);`
- 8) `10 div 3;`
- 9) `10/3;`
- 10) `1266 div 3 mod 5;`
- 11) `(p and (q and not q)) or not (r or (s or not s));`
- 12) `(round(-65.3)<trunc(-65.3)) and p;`
- 13) `odd(k) or odd(k+1).`

## 5. Строковый тип данных

5.1. Как записываются строковые константы?

5.2. Как описываются строковые переменные?

5.3. Перечислите операции и функции, разрешенные для данных строкового типа.

5.4. Даны три записи на Паскале: `101`, `101.0`, `'101'`. Запишите одним словом, в чем сходство этих записей, и одним словом, в чем разница между ними.

5.5. Типы данных заданы константами: `2`, `2.0`, `'2.0'`. Укажите знак операции, который можно применить для обозначения операции с данными любого из этих типов.

5.6. Запишите условие истинное, если:

- 1) в записи фамилии «Соколенко» встречается буква «л»;
- 2) в записи фамилии «Соколенко» не встречается буква «л»;
- 3) в записи фамилии «Соколенко» встречается буква «я»;
- 4) в записи фамилии «Соколенко» не встречается буква «я»;
- 5) в записи фамилии «Соколенко» встречается буква «н», но не встречается буква «у»;
- 6) слово *a* длиннее слова *b*;
- 7) из букв фамилии «Соколенко» можно образовать слово «лоск»;
- 8) из букв фамилии «Соколенко» можно образовать слово «локон»;
- 9) из букв фамилии «Соколенко» можно образовать слово «кокон»;
- 10) из букв фамилии «Соколенко» нельзя образовать слово «коляска»;
- 11) из букв фамилии *A* нельзя образовать слово *B*.

5.7. Выполняется ли коммутативный закон для сцепления? А ассоциативный?

5.8. Сколько начальных позиций в строке длиной *N* может быть у подстроки длиной *M* ( $N > M$ )?

5.9. Сколько начальных позиций может быть у пустой подстроки в непустой строке?

5.10. Какое значение вернет процедура `val('2e5',i,k)`?

5.11. Какое значение вернет процедура `val('+2e-5',i,k)`?

5.12. Какое значение вернет процедура `val('+2-5',i,k)`?

5.13. Укажите оператор Паскаля, использование в котором одной из следующих конструкций: 1) '#','#' 2) '##' 3) '#'+#' даст одинаковые результаты. Опишите правила выполнения этого оператора. Запишите названия каждой конструкции. Приведите другие варианты, выдающие такой же результат.

5.14. Пусть  $a, b, x$  - строки и  $a$  - начальная часть  $b$ .

1) Найдите  $x$  из условия:  $a+x=b$ ;

2) Найдите  $x$  из условия, если  $a$  - конец  $b$ :  $x+a=b$ .

3) Опишите все стандартные функции работы со строками, которые использовались при решении пунктов 1 и 2.

5.15. 1) Пусть  $x$  - строка. Укажите, для каких значений  $x$  справедливо равенство  $x='x'$ .

2) Пусть  $x$  и  $y$  - строки, причем длина  $x$  равна 1. Укажите условие, при котором  $x+y=y+x$ .

3) Опишите все стандартные функции, которые использовались при решении пунктов 1 и 2.

5.16. 1) Пусть  $a$  - строка, "+" - сцепление. Найдите строку  $x$ , удовлетворяющую условию  $a+x=a$ .

2) Пусть  $a, b, x$  - строки и  $a$  - начальная часть  $b$ . Найдите  $x$  из условия  $a+x=b+a$ .

3) Описать все стандартные функции, которые использовались для решения пунктов 1 и 2.

5.17. Укажите, какие из указанных ниже операций допустимы в Паскале:

```
var a1,a2:string; b1,b2:string[80]; c:string[5];
```

1)  $a1=a2$ ;

2)  $b2>b1$ ;

3)  $a1\leq b2$ ;

4)  $c=b1$ ;

5)  $c:=a2$ ;

6)  $a2:='b1+b2'$ ;

7)  $a1:=a2$ ;

8)  $a2:=b2$ ;

9)  $c>='12345'$ ;

10)  $a1<>'**'$ ;

11)  $c:=a1+b2$ .

5.18. Укажите, какие из указанных ниже выражений допустимы в Паскале:

```
var a1,a2:string; b1,b2:string[80]; c:string[5];
```

1)  $\text{copy}(a1,5,3)+\text{length}(a2)$ ;

2)  $\text{length}(b2)+\text{pos}('aa',b1)$ ;

3)  $\text{pos}(c,a1)+\text{length}(\text{copy}(c,6,2))$ ;

4)  $b1+\text{copy}(b2,65,20)+\text{copy}(\text{pos}('a',c),1,7))$ ;

5)  $\text{pos}(\text{copy}(a1,4,6),\text{delete}(a2,2,1))+\text{length}(b2)$ ;

6)  $\text{insert}(\text{copy}(c,2,3),a2+b2,3)$ ;

7)  $\text{delete}(\text{copy}(a2,5,6),3,4)$ ;

8)  $\text{insert}(\text{copy}(a1,5,5),a2,3)$ ;

9)  $\text{insert}(a1,a2,\text{length}(a2))$ ;

10) insert(b1,b2,length(b2)+1);

11) insert(b1,b2,length(b1)); .

5.19. Установите значения следующих выражений:

1) concat('кро',a,b)

при a='код', b='ил';

2) copy(s,2,3)

при s='крокодил';

3) copy(s,4,2)+s[2]+s[7]+s[6]+s[5]+s[2]

при s='крокодил';

4) pos('око',s)

при s='крокодил';

5) pos('я','крокодил');

6) pos('а','ёеуыаоэяию');

7) pos('б','ёеуыаоэяию');

8) length('зеленый'+ ' '+s)

при s='крокодил';

9) insert('чо',w,5)

при w='моряк';

10) insert('Н',d,3)

при d='око';

11) delete(w,5,2)

при w='морячок';

12) str(123,s);

13) pos('а','аяоёуыиэе')>0;

14) pos('б','аяоёуыиэеАЯОЁУЮЫИЭЕ')>0;

15) copy(s,pos(' ',s)+1,length(s)-pos(' ',s))

при s='Вася Кук – информатик';

16) delete(s,length(s),1)

при s='123456789';

17) copy(s,i,j-i+1)

при s='123456789', i=3, j=7;

18) insert(b1,b2,length(b1))

при b1='1', b2='22';

19) insert(b1,b2,length(b1))

при b1='11'; b2='22';

20) insert(b1,b2,length(b1))

при b1='111', b2='22';

21) insert(b1,b2,length(b2))

при b1='1', b2='22';

22) insert(b1,b2,length(b2)-1)

при b1='1', b2='22';

23) insert(b1,b2,length(b2)+1)

при b1='1', b2='22'.

24) delete(s,1,length(s))

при s='11111';

25) delete(s,1,0)

при s='11111';

26) delete(s,1,-2)

при s='11111';

27) delete(s,1,length(s div 2))

при s='11111';

28) delete(s,1,length(s mod 2))

при s='11111' и при s='2222';

29) delete(s,length(s) div 2,1)

при s='12345';

30) delete(s,length(s) mod 2,1)

при s='12345';

31) delete(s,length(s) mod 2,1)

при s='123456';

32) delete(s,-100,1)

при s='12345'.

5.20. Проверьте, является ли заданный символ строчной гласной буквой русского алфавита.

5.21. Проверьте, является ли заданный символ согласной буквой русского алфавита (заглавной или строчной).

5.22. Проверьте, является ли заданный символ цифрой.

5.23. Пусть объявлена переменная var a:string[11]. Какое значение будет содержать эта переменная после присваивания a:='сумей'+',догадайся'?

5.24. Подставьте вместо знаков вопроса буквы русского алфавита так, чтобы получить осмысленное слово и не нарушить истинность логического выражения:

1) 'конфета' < 'конф??' = true;                      2) 'ко????' > 'конфета' = false.

5.25. Из слова 'понедельник', используя функции строковых данных, получите слово 'плодик'.

5.26. Какие из следующих строк можно преобразовать в числа:

- 1) '1e+1' ,
- 2) '-1.256' ,
- 3) '+295.689e6' ,
- 4) '+1.2e-3' ,
- 5) '123456789012345' ,
- 6) '2,7e+2' ,
- 7) 'II' ,
- 8) '+2-3' ?

5.27. Пусть строка состоит из слов, разделенных запятыми, например, слово1,слово2,слово3. Запишите оператор, удаляющий второе слово исходной строки.

5.28. Из указанных фрагментов программ выберите эквивалентные, т.е. выполняющие одну и ту же работу. Назовите также все операторы и типы данных, используемые в приведенных фрагментах.

- 1) for i:=1 to length(s) do writeln(copy(s,i,i));
- 2) for i:=length(s) downto 1 do writeln(copy(s,i,i));
- 3) for i:=length(s) downto 1 do writeln(copy(s,i,length(s)-i+1));
- 4) for i:=length(s) downto 1 do writeln(copy(s,length(s)-i+1,i));
- 5) for i:=1 to length(s) do writeln(copy(s,i,length(s)-i+1));
- 6) for i:=1 to length(s) do writeln(copy(s,length(s)-i+1,i)).

5.29. Установите, какую задачу решает приведенный ниже фрагмент:

- 1) s:="";  
for i:=1 to length(t) do  
if pos(copy(t,i,1),'aeiouy')>0  
then s:=s+copy(t,i,1);
- 2) s:="";  
for i:=length(t) downto 1 do  
s:=copy(t,i,1)+s.

## **6. Программы на Паскале.**

### **Ввод, вывод, присваивание**

6.1. Что такое «комментарий»? Как он устроен и для чего используется?

6.2. Правильно ли употреблены комментарии?

{ Этот фрагмент Паскаль-программы содержит очень много  
к о м м е н т а р и е в }

a{переменная левой части}:=a{выражение} + b{правой} - 16{части};  
writeln(a,{разделяем выражения}b)

6.3. Как задаются правила построения различных конструкций Паскаля?

6.4. Из каких частей состоит программа на Паскале?

6.5. Обязательно ли необходимы описания в Паскаль-программе?

6.6. Для чего описывают константы?

6.7. Для чего описывают типы?

6.8. Для чего описывают переменные?

6.9. Верно ли, что в Паскаль-программе нужно описывать все переменные?

6.10. Какую информацию извлекает транслятор из описания переменных?

Как он её использует?

6.11. Какие значения имеют переменные в начале выполнения программы?

6.12. Можно ли менять значения констант в процессе выполнения программы?

6.13. В описании констант их типы не указываются. Как определяет типы констант транслятор?

6.14. Почему не надо описывать константы: maxint, true, false, pi?

6.15. Для чего описывают процедуры и функции?

6.16. Как делаются описания констант, типов, переменных, процедур и функций?

6.17. Найдите ошибки в программе:

```
programm 101;  
{ эта программа печатает число недель в году {  
var s; integer;  
begin  
s:=56;  
write(В году s недель); end
```

6.18. Найдите ошибки в программе:

```
program 102;  
constanta dwa=2;  
tri=dwa+1;  
пять=tri + dwa;  
var info,comma:integer,  
tigr,lev,volk:real;  
serg:olga;  
type olga=real;  
begin readln(serg);  
write('Введите имя ');  
writeln(serg=olga)  
end
```

6.19. Что будет напечатано после выполнения программы, если программист введет с клавиатуры числа: 1 2 3 4 5<- ?

```
program abba;  
var a,b,c:integer;  
begin read(a,b,a,c,b);  
write(a,b,c)  
end.
```

6.20. Что будет напечатано после выполнения программы?

```
program print;  
begin write(1);write(2,3);writeln(4);
```

```

write(5);writeln(6,7);writeln;
write(8)
end.

```

6.21. Что будет напечатано?

```

write(1);writeln(2,3);write(4);writeln(5,6);write(7);writeln(8,9).

```

6.22. Как изменится результат работы, если в приведенном фрагменте поменять местами второй и третий операторы присваивания?

- 1) a := b;
- 2) b := c;
- 3) c := a.

6.23. Как изменится результат работы, если в приведенном фрагменте третий оператор присваивания поставить на первое место?

- 1) a := b;
- 2) b := c;
- 3) c := a.

6.24. Какие из записей: «a:=a+1» и «a=a+1» ошибочны в Паскале? Почему?

6.25. Как напечатать фразу: «Хотите купить утку?»

6.26. Напишите программу, печатающую имя того, кто её исполняет.

6.27. Каков будет результат работы программы?

```

program wr(input,output);
var i:integer;
begin i:=maxint;
      writeln(i,' ',i+1,' ',i+2);
      readln;
end.

```

6.28. В программе используется оператор i:=i+33000, где var i:integer. Ошибочен ли этот оператор?

6.29. Исправьте ошибки в программе:

```

program temp;
const tri=3,
      dba=2,
      odin=1;
var   g,h : real;
      tax,rate : real;
end
      g = e21;
      tax = rate * g;
begin.

```

6.30. Исправьте ошибки в программе:

```

Program Ой-ей-ей(input,output);
const b=Ай-яй-яй;
      x=10;
var   age:integer;
      name:string;

```



```

begin write("Введите, пожалуйста, свое имя ");
      readln(name);
      write("Прекрасно, ",name," , сколько Вам лет? ");
      readln(age);
      xr:=age+x;
      write(name,"! Вам должно быть, по крайней мере ",xr," лет")
end.

```

6.31. Пусть  $x=2$ ,  $y=3$ . Напишите оператор печати фразы «Сумма  $2+3=5$ », используя значения переменных  $x$  и  $y$ .

6.32. Известно, что  $a \bmod b$  не равен нулю. Какое число нужно добавить к переменной  $a$ , чтобы она стала кратной  $b$ ?

6.33. Даны значения переменных:  $a=1$ ,  $b=5$ . Какими будут их значения после выполнения последовательности операторов?  $a:=b$ ;  $b:=a$

6.34. Какими значениями будут обладать переменные  $x$  и  $y$  после выполнения последовательности операторов?

а)  $x:=15 \operatorname{div} (8 \bmod 3)$ ;  
 $y:=17 \bmod x*5-19 \bmod 5*2$ ;

б)  $x:=2*5 \operatorname{div} 3 \bmod 2$ ;  
 $y:=2*5 \operatorname{div} (3 \bmod 2)$ ;  
 $x:=x*y$ ;  
 $y:=y*y$ .

6.35. Значение переменной  $a$  - трехзначное натуральное число. Результат - двухзначное число  $b$ , которое получается из  $a$ , если вычеркнуть среднюю цифру. Напишите операторы для получения числа  $b$ .

6.36. Укажите ошибки в записи оператора ввода:

```

read(a,b,ab);
rEaD(i);
read(begin);
read{возможно ли такое?}(a,b);
READ(a{a такое возможно?},b);
read({может быть можно так?});
re{или так?}ad(a,b);
read((a,b));
read((a),b);
read(a,a,a);
read(a,1,b);
read('Введите число ',a);
read( a,
      b,
      c);

```

г  
е  
а

```
d
(
a
,
b
).
```

6.37. Что получится в результате выполнения следующей программы, если пользователь ввел в буфер ввода числа 1 2 3↵?

```
program test2;
{что же получится?}
var a,b,c:integer;
begin write('Введите три целых числа ');
      readln(a,b,a);
      c:=a+b;
      write('a+b=',c)

end.
```

Укажите оператор, который можно убрать из программы так, что результат ее работы не изменится.

6.38. Запишите самый короткий оператор Паскаля.

6.39. Запишите самую короткую программу на Паскале.

6.40. Установите формулу, вычисляемую программой:

```
program exampl;
begin a,b,c:integer; {коэффициенты}
      d,e:integer; {промежуточные переменные}
      x:integer; {аргумент}
      r:integer; {результат}

begin write('Введите коэффициенты ');
      readln(a,b,c);
      write('Введите значение аргумента ');
      readln(x);
      d:=a*sqr(x);
      e:=b*x;      {*}
      r:=d+e+c;
      write('результат=',r:5)

end.
```

Что будет получено, если в операторе, отмеченным звездочкой, вместо операции умножения (\*) использовать операцию деления (/)?

6.41. Что будет напечатано в результате работы программы:

```
program ex;
var  a,b,c,d:integer; {исходные данные}
     a1,b1,c1,d1:integer; {копии исходных данных}
     r:integer; {для временного хранения значения}
begin write('Введите исходные данные: четыре целых числа ');
      readln(a,b,c,d);
      a1:=a; b1:=b; c1:=c; d1:=d;
```

```

r:=a;
a:=b;
b:=c;
c:=d;
d:=r;
a1:=a1+b1;
b1:=a1-b1;
a1:=a1-b1;
b1:=b1+c1;
c1:=b1-c1;
b1:=b1-c1;
c1:=c1+d1;
d1:=c1-d1;
c1:=c1-d1;
write((a=a1) and (b=b1) and (c=c1) and (d=d1))

```

end.

6.42. Что будет напечатано следующей программой, если для ввода были заданы числа: 5 4 3?

```

program primer; {schlecht arbeit} var fir:integer; funf,a,b:real; begin
read(a,fir,b); funf:=5*(a+b)-fir; fir:=a*funf; write(funf,' ',fir); {какова
программка? вам нравится?} b:=sqr(abs(funf-fir)*sin(a)-3); writeln(a+b-
a*b+(funf-fir)); a:=abs(b-a+funf-fir)*a+b-fir*funf-sin(a+cos(b-sqrt(funf)
+2)); writeln('результат=',a,' ',b,' ',funf,' ',fir,' ',a+b,' ',funf+fir,' ',a-b-fir-
fir);; end.

```

6.43. Найдите ошибки в каждой из следующих программ:

1) program A;  
const d=5;  
begin d:=sqr(d);  
writeln('d\*\*2=',d)  
end.

2) program B;  
const k=true;  
var x:real;  
begin read(x);  
writeln(odd(x)=k)  
end.

3) program C;  
var a,b,c:integer;  
begin read(a,b);  
writeln((a+b+c)/3)  
end.

4) program D;  
var x:real;  
begin read(x);  
y:=sqr(x)+1;

```

        writeln(y)
    end.
5) program E;
    const B:=2.5;
    var a,b,c:real;
    begin read(a,c);
        writeln(a*c>b)
    end.
6) program F;
    var a,b:integer;
    begin read(A);
        d:=odd(pi*0) and b>a;
        writeln(d)
    end.

7) program G;
    var a,b:integer;
        r:integer;
    begin readln(a,b);
        r:=a/b;
        write(r)
    end.
8) program H;
    var a,b:real;
        r,p:integer;
    begin readln(a,b);
        r:=a div b;
        p:=a mod b;
        write(r,' ',p)
    end.
9) program I;
    var a,b:integer;
        r:real;
    begin readln(a,b);
        r:=a div b;
        write(r)
    end.

```

6.44. Каким образом можно вводить значения логических переменных?

6.45. Задача о расчётливой Дорис. На вопрос, пойдёт ли она на вечеринку, Дорис ответила: «Мне нужен кавалер на этот вечер. Хорошо, если там окажется Боб или Карл. Только не оба сразу, я не хочу оказаться между двух огней. Впрочем, если там будет Анна, то Боб на меня и не посмотрит».

Ниже приводится программа, которая, по мнению авторов, позволяет определить, пойдёт ли Дорис на вечеринку, если туда пойдут её друзья. Известно, что если кто-то идёт на вечеринку, то для него вводится символ  $T$ , в

противном случае –  $F$ . Установите, совпадет ли ответ программы с мнением Дорис. Какие претензии можно высказать по поводу этой программы?

```
program Doris;      {задача о расчётливой Дорис}
var   A{nna}, B{ob}, C{arl}, D{oris} : boolean;
      s:string;
begin write('Пойдет ли Анна на вечеринку? ');
      readln(s);
      a:=s[1]='T';
      write('Пойдет ли Боб на вечеринку? ');
      readln(s);
      b:=s[1]='T';
      write('Пойдет ли Карл на вечеринку? ');
      readln(s);
      c:=s[1]='T';
      d:=not a and b xor c;
      write('Пойдет ли Дорис на вечеринку? ',d:5,'!');
end.
```

6.46. Что будет напечатано следующими операторами?

```
var d : boolean;
    d:=true;
    writeln('1) ',d);
    writeln('2) ',d=false);
    writeln('3) ',d<true);
    writeln('4) ',not d or d);
    writeln('5) ',not d>false);
    writeln('6) ',not (d and not d)=d or not d).
```

6.47. Найдите ошибки в программах:

```
1) i:integer;
   begin write('задайте целое число ');
       readln(i);
       write(i)
   end;
2) var i:real;
   begin i:=1;
       writeln('i=',j)
   end.
```

6.48. Всегда ли выполнение последовательностей операторов

```
read(x,y); z:=sqr(x)+1; write(x/z,' ',y/z,' ',x)
```

и

```
read(x,y); write(x/sqr(x)+1,' ',y/sqr(y)+1,' ',x)
```

при одних и тех же исходных данных приведет к выводу одних и тех же чисел?

6.49. Какие скобки в приведенных выражениях можно снять, не изменив значения этих выражений? Предложите алгоритм проверки правильности расстановки скобок в выражении:

$((a+b)/c),$

```

((a+(b/c))-((2*d)/(2+d))),
not ( not (a<b) or (c>=d) ),
((a+b)-(((c+d)+(2*e))))),
((a+b)+c)+((c+d)*(2*e)),
(((a-b)-c)-d),
(a-(b-(c-d))).

```

6.50. В каком порядке и какие числа должны вводиться, если после выполнения операторов Паскаля было напечатано: 1 2 3 ?

```

read(a, b, c, a, b);
write(c, b, a).

```

6.51. Сколько строк выходных данных будет напечатано следующим фрагментом программы:

```

x:=1;
while x<=5 do
begin y:=10;
  while y>2 do
  begin writeln(x*y);
    y:=y-0.5
  end;
  x:=x+0.5
end.

```

6.52. Сколько строк выходных данных будет напечатано следующим фрагментом программы:

```

x:=1;
while x<=5 do
begin y:=20;
  while y>=7 do
  begin writeln(x*y);
    y:=y-1
  end;
  x:=x+0.5
end.

```

6.53. Какую задачу решает приведенный фрагмент:

```

q:=0; r:=x;
while r>=y do
begin r:=r-y;
  q:=1+q
end.

```

6.54. Какую задачу решает приведенный фрагмент:

```

x:=x0; y:=y0; z:=0;
while x<> 0 do
begin if x mod 2 <> 0
  then z:=z+y;
  y:=y*2;

```

```

        x:=x div 2
    end.
6.55. Какую задачу решает приведенный фрагмент:
    s:=0; i:=a;
    while i>=b do
    begin p:=1;
        for j:=i downto 2 do p:=p*j;
        s:=p+s;
        i:=i-2
    end.

```

## 7. Условный оператор, составной оператор, оператор выбора.

### Типы: перечислимый и диапазон

7.1. Запишите как можно проще на Паскале:

ЕСЛИ  $a$  больше  $b$

ТО ничего не делать

ИНАЧЕ ЕСЛИ  $2d$  меньше  $3\sin(x)$

ТО ЕСЛИ  $a$  больше двух с половиной умноженных на  
десять в минус третьей степени

ТО ничего не делать

ИНАЧЕ  $c$  присвоить 32

ВСЕ

ИНАЧЕ ничего не делать

ВСЕ

ВСЕ.

7.2. Нарисовать графики функций, заданные фрагментами программ:

1) if  $x < -1$

then  $y := 1/\sqrt{x}$

else if  $x \leq 2$

then  $y := \sqrt{x}$

else  $y := 4$ ;

2) if  $x < -0.5$

then  $y := 1/\text{abs}(x)$

else if  $x < 1$

then  $y := 2$

else  $y := 1/(x-0.5)$ ;

3) if  $\text{abs}(x) > 2$

then  $y := \sqrt{x}$

else if  $x < 0$

then  $y := -2*x$

else if  $x \geq 1$

then  $y := 4$

else  $y := 4*\sqrt{x}$ ;

4) if  $\sqrt{x} > 2$

then if  $x > 2$

```

        then y:=x*sqr(x)
        else y:=8
    else y:=8*sqr(x);
5) if x>2
    then if x>3
        then y:=x-1
        else y:=2
    else y:=x;

```

```

6) t:=frac(x);
    if t<0.25
    then y:=t
    else if t<0.75
        then y:=-t+0.5
        else y:=t-1;

```

```

7) t:=x-int(x/4.5)*4.5;
    if x<0.75
    then y:=x
    else if x<1.5
        then y:=0.75
        else if x<3
            then y:=-x+2.25
            else if x<3.75
                then y:=-0.75
                else y:=x-4.5.

```

7.3. Упростите запись условного оператора в указанном фрагменте:

```

if a<=5
then c:=1
else if a>5
    then c:=2
    else c:=3.

```

7.4. Определите, каким было значение переменной  $a$  до выполнения фрагмента программы, если после его выполнения  $a=3$ .

```

a:=? ;
if a<5
then a:=1
else if a>5
    then a:=2
    else a:=3.

```

7.5. В предложенной программе подчеркните все операторы присваивания и восстановите описания типов переменных.

```

program test1;
{подчеркните присваивания и восстановите описания переменных}
var    a:
        b:

```



c:  
d:  
e:  
f:  
g:  
h:  
k:

```
begin write('Введите два числа ');
      readln(a,b,c);
      k:=a+b/c;
      if a<b
      then if b<c
            then d:=c=e
            else write('d:=c<>e')
      else for f:=1 to 10 do g:=g+k*f;
            {h:=g+a+b+c+k}
      write(g,' ',h)
end.
```

7.6. Определите значение  $a$  после выполнения фрагмента программ, если в одном случае  $a=0$ , а в другом  $a=6$ :

- 1) if  $a < 0$  then ;  $a:=2$ ;
- 2) if  $a < 4$  then begin  $a:=a+2$ ;  $a:=a+3$  end;
- 3) if  $a < 4$  then  $a:=a+2$ ;  $a:=a+3$ ;
- 4) if  $a < 4$  then  $a:=a+2$  else  $a:=a+3$ ;
- 5) if  $a < 4$  then else begin  $a:=a+2$ ;  $a:=a+3$  end.

7.7. Известно, что в составном операторе количество begin должно строго соответствовать количеству end. Для каких элементов условного оператора выполняется подобное же соотношение?

7.8. Может ли в записи программы на Паскале количество ключевых слов else быть больше количества ключевых слов then? А меньше?

7.9. Могут ли в условном операторе стоять рядом две фразы then? А две фразы else?

7.10. Являются ли составными операторами:

- 1) begin  $a:=0$  end;
- 2) begin begin  $a:=0$  end end;
- 3) begin ; end;
- 4) begin  $x:=0$ ;;;;;; $x:=0$  end;
- 5) begin begin writeln('Бывает же такое!') end;
- 6) if  $x < y$   
 then begin  
 else end;
- 7) begin {begin } end;
- 8) begin { begin end};
- 9) begin {  $x:=7$  end};

- 10) begin begin x:=1; y:=2 end x:=3 end;
- 11) begin begin x:=1; y:=2 end; x:=3 end;
- 12) begin if x>1 then begin end; if then else end;
- 13) begin b:=e; begin write('begin') end end;
- 14) begin b:=e; begin write('begin'); end end;
- 15) begin b:=e; begin write('begin a:=2; end') end; end;
- 16) begin begin begin end begin end end begin end end.

7.11. Являются ли условными операторами следующие последовательности СИМВОЛОВ:

- 1) if x<y then x:=0 else y:=0;
- 2) if x>y then x:=0 else read(y);
- 3) if x>=y then x:=0;y:=0 else write(y);
- 4) if a<b then 100 else z:=25;
- 5) if a=b then begin; c:=3 end else c:=1;
- 6) if f or false then a:=2\*f else a:=c-2;
- 7) if c<2 or a>2 then read(a) else read(c);
- 8) if a+b then write('a') else write('b');
- 9) if a and true then begin c:=1 else c:=2 end;
- 10) if (1<=x) and (x<=9) then c:=2; else c:=3;
- 11) if 2\*a-3>0 then (a:=0) and (b:=0) and (c:=0);
- 12) if t<z else a:=4;
- 13) a:=if c>d then d else c;
- 14) if {это условие} a<b {если условие истинно} then a:=b else {если условие ложно} b:=a;
- 15) if c:=0 then begin read(c);c:=0 end else begin write(c);c:=0 end;
- 16) if a=2 then a:=a+1 else P(x,y);
- 17) if x<y then; z:=true else z:=false;
- 18) if if a=b then true else false then a:=1 else a:=1;
- 19) if a and b or c then else;
- 20) if a or not a then if b and not b then else else if d=0 then write(1) else write(2).

7.12. Объясните, как будут выполняться фрагменты программ:

- 1) if a<b then a:=b else a:=b;
- 2) if false then write('a') else write('b');
- 3) if true then write('a') else write('b');
- 4) if 2\*a<=3\*b then a:=a+b-14 else a:=b-14+a;
- 5) if a<b then a:=b else a:=a;
- 6) if a>b then a:=a else a:=b;
- 7) if a mod 2<>3 then a:=a+1;
- 8) if (a mod 2=1) or (a mod 2=0) then if a<0 then a:=-a;
- 9) if a<0 then b:=a else b:=-a.

Замените приведенные фрагменты, где возможно, более короткими операторами Паскаля.

7.13. Запишите одним условным оператором:

- 1) if k>5 then c:=1;

```

    if k>5 then d:=2;
    if k>5 then e:=3;
    if k<=5 then write(c+d);
    if k<=5 then write(d+e);
2) if a>b then c:=1;
    if a>b then d:=2;
    if a<=b then c:=3;
    if a<=b then d:=4;
3) if x<y then x:=1 else y:=2;
    if x<y then y:=0.5 else x:=1.5;
    if x>=y then z:=0.25 else z:=2.25;
    if x>=y then d:=5;
4) if a-b<0 then else c:=1;
    if a-b<0 then else d:=2;
    if a>=b then else c:=3;
    if a>=b then else d:=4.

```

7.14. Какую работу выполняет оператор (сформулируйте предложением русского языка): `if x>y then y:=2*y else x:=2*x`.

7.15. Постройте тесты для проверки работоспособности программ:

```

1) if a>b
    then if b<0
        then c:=a
        else c:=b
    else c:=0;
2) if a>b
    then if a>c
        then if a>d
            then c:=1
            else c:=2
        else c:=3
    else if b>c
        then if b>d
            then c:=4
            else c:=5
        else c:=6.

```

7.16. Какое значение имела переменная *a*, если переменной *c* было присвоено значение 3?

```

    if a<5
    then c:=1
    else if a=5
        then c:=2
        else c:=3.

```

7.17. Чему равнялось *b*, если было напечатано \$?

```

    if a>b
    then write('#')

```

```

else if a<b
    then write('@')
    else write('$').

```

7.18. Чему было равно исходное значение  $d$ , если после выполнения программы ничего не было напечатано? Нарисуйте блок-схему. Разработайте тесты для прохождения каждой ветки программы хотя бы по одному разу.

```

if c>d
then if 2*d>c
    then write(1)
    else begin if 2*d<c
        then write(2)
    end
else write(3).

```

7.19. Нарисуйте блок-схемы и объясните употребление условного оператора. Перепишите фрагменты программ со сдвигами. Какой вариант записи: исходный или со сдвигами более нагляден?

1) if a=b then if c=d then w:=x else w:=y else w:=z;

2) if a=b then if c=d then w:=x else w:=y;

3) if a=b then w:=x else if c=d then w:=y else w:=z;

4) if a=b then begin if c=d then w:=a else w:=b; x:=c end else if e=f then begin if c=d then k:=3 end else w:=p;

5) if a=b then if c=d then w:=x else else w:=y.

7.20. В чем проявится разница в напечатанных значениях в случае  $a=1$  и  $a=2$  для фрагмента: if a<2 then write(1) else write('1').

7.21. Что будет напечатано?

```

var  a,b:integer;
     aa,bb,cc:boolean;
begin a:=3; b:=5;
      aa:=a<b; bb:=a>c; cc:=a;
      if aa then write('a');
      if bb then write('b');
      if cc then write('c')
end.

```

7.22. Подсчитайте количество пустых операторов в записях:

1) begin k:=0;;d:=c end;

2) begin end

3) begin begin ; end;end;

4) if x>0 then x:=2 else; y:=x+1;

5) if odd(k) then else k:=0;

6) begin x:=2;y:=5 end;

7) begin a:=true;;b:=b or a end;

8) begin x:=0; end;

9) while x>0 do; x:=x-2.

7.23. Найдите ошибку в программе, определяющей вид треугольника (равносторонний, равнобедренный или разносторонний):

```

if a=b
then  if a=c
      then write('равносторонний')
      else write('равнобедренный')
else  if b=c
      then write('равнобедренный')
      else write('разносторонний').

```

7.24. Запишите на Паскале следующие условные операторы (выводится значение ДА, если условие выполнено. В противном случае выводится НЕТ):

- 1) целые числа  $x$  и  $y$  имеют одинаковые знаки;
- 2) целые числа  $x$  и  $y$  имеют разные знаки;
- 3) точка  $M$  с координатами  $(x,y)$  лежит внутри окружности единичного радиуса с центром в начале координат;
- 4) точка  $M$  с координатами  $(x,y)$  лежит выше биссектрисы первого и третьего координатных углов;
- 5) точка  $M$  с координатами  $(x,y)$  лежит ниже биссектрисы второго и четвертого координатных углов;
- 6) расстояние между точками  $x$  и  $y$  меньше 0,1;
- 7) значение переменной  $x$  лежит на отрезке  $[a;b]$ ;
- 8) значение переменной  $x$  лежит вне отрезка  $[a;b]$ ;
- 9) значение переменной  $x$  лежит вне отрезков  $[a;b]$  и  $[c;d]$ ;
- 10) точка  $M$  с координатами  $(x,y)$  лежит внутри параболы:  $ax^2+bx+c$ ;
- 11) точка  $M$  с координатами  $(x,y)$  лежит под параболой:  $ax^2+bx+c$ ;
- 12) точка  $M$  с координатами  $(x,y)$  лежит внутри квадрата, заданного координатами левого верхнего -  $(x1,y1)$  и правого нижнего угла -  $(x2,y2)$ ;
- 13) целое число  $x$  является четным;
- 14) целые числа  $x$  и  $y$  имеют одинаковую четность;
- 15) целое число  $a$  делится на целое число  $b$ ;
- 16) целое число  $a$  является полным квадратом;
- 17) квадратное уравнение  $ax^2+bx+c$  имеет действительные корни;
- 18) каждая из трех целых переменных  $x, y, z$  положительна;
- 19) хотя бы одна из трех целых переменных  $x, y, z$  положительна;
- 20) ни одна из трех целых переменных  $x, y, z$  не положительна;
- 21) только одна из трех целых переменных  $x, y, z$  положительна;
- 22) не больше двух из трех целых переменных  $x, y, z$  положительны;
- 23) только две из трех целых переменных  $x, y, z$  положительны;
- 24) целых переменных  $x, y, z$  равны между собой;
- 25) среди трех целых переменных  $x, y, z$  имеются две равных;
- 26) среди трех целых переменных  $x, y, z$  имеется не более двух равных;
- 27) среди трех целых переменных  $x, y, z$  имеется не менее двух равных;
- 28) каждое из четырех целых чисел равно 5;
- 29) точка  $M$  с координатами  $(x,y)$  лежит внутри параболы  $2x^2+x+1$ , но вне окружности с центром  $(1;1)$  и радиусом 18;
- 30) точка  $M$  с координатами  $(x,y)$  лежит вне параболы  $2x^2+x+1$ , но внутри окружности с центром  $(1;1)$  и радиусом 18.

7.25. Определите, какими будут значения переменных  $p$  и  $d$  после выполнения операторов:

```
p:=true; d:=1;
case k mod 10 of
3,2,7,5: d:=k;
1: ;
4,8: begin p:=false; d:=2 end;
9,6: begin p:=false; d:=3 end;
end {of case}
```

если целая переменная  $k$  имеет значение: 1) 6; 2) 235; 3) 71; 4) 100.

7.26. Найдите ошибки в операторах:

```
var u,w:'a'..'z';
case u of
'a'..'w': w:=succ(w);
'u','y': u:='g'; w:=u
end.
```

7.27. Найдите ошибки в программе:

```
type sezon=(zima,vesna,leto,osen);
var x,y:sezon;
t:(teplo,xolodno);
procedure oxota(s:sezon;var p:(teplo,xolodno));
begin if (ord(s)=0) or (ord(s)=1)
then p:=xolodno
else p:=teplo
end;
begin {1} x:=vesna;
{2} y:=x;
{3} t:=teplo;
{4} y:=t;
{5} t:=xolodok;
{6} read(x,t);
{7} write(vesna);
{8} writeln(t);
{9} if succ(x)=leto
then write(true)
else write(false);
{10} case y of
zima..leto: write('это - не осень');
osen: write('а вот это осень!');
end;
{11} case t of
pred(xolodno):write('не холодно');
succ(teplo):write('не тепло');
end;
{12} for x:=zima to osen do write(x<y);
```

```

    {13} for t:=xolodno to teplo do write(ord(t));
    {14} while ord(zima)+ord(vesna)<ord(osen) do write('Большой
ПРИВЕТ!');
    {15} oxota(x,t);
end.

```

7.28. Установите, какие значения будут напечатаны:

```

type sezon=(zima,vesna,leto,osen);
var x,y:sezon;
    t:(teplo,xolodno);
begin write(vesna<leto);
      write(zima<=leto);
      write(osen<zima);
      write(vesna<>teplo);
      write(ord(succ(vesna)), ' ',ord(pred(xolodno)));
end.

```

7.29. Укажите ошибки в следующих описаниях:

```

type
  {1} bukva=('a','b','c','d','e');
  {2} glas=(a,e,i,o,u);
  {3} soglas=(b..d,f,g);
  {4} kart=(6,7,8,9,10,va,da,ko,tu);
  {5} root=(1.00,1.41,1.73,2.00);
  {6} ab=(a1,a2,a3,a4,a5);
  {7} a3=(aa1,aa2,aa3);
  {8} bb=(a1,b2,c3,d4,e5,f6);
  {9} delenie=(div,mod);
  {10} logical=Boolean.

```

7.30. Имеются описания:

```

type digit='0'..'9';
var   d:digit;
      k:0..9;
      n:integer;

```

Ответьте на следующие вопросы:

1) Каковы значения типа digit? Каков базовый тип этого диапазона? Допустимы ли присваивания: а) d:='7'; б) d:='a'; в) d:=7?

2) Какие значения может принимать переменная k? Каков её базовый тип? Допустимы ли присваивания: а) k:=5; б) k:=10; в) k:=-0; д) k:='5'; е) k:=ord(d); г) k:=ord(d)-ord('0')?

3) Верно ли, что к значениям ограниченного типа можно применять те же операции, что и к значениям базового типа? Есть ли ошибки в операторе:

```
if k+n>7*k then k:=abs(n) mod 10 else d:=chr(k+ord('0'))?
```

4) Можно ли в Паскале определить ограниченный тип, состоящий из простых чисел?

7.31. Укажите ошибки в следующих описаниях:

```
const n=180;
```

```

type  chifra='0'..'9';
      bukva=a..z;
      ugol=-n..n;
      otrezok=0..n-1;
      period=-pi..pi;
      plus='+'..'+';
      nedelja=(vs,pn,vt,sr,ch,pt,sb);
      budni=pn..pt;
      vixodnoy=sb..vs.

```

7.32. Установите, какую задачу решает приведенный ниже фрагмент:

```

type  month=(jan,feb,mar,apr,may,jun,jul,aug,sep,oct,nov,dec);
      day=1..31;
var d1,d2:day; m1,m2:month; t:Boolean;
begin t:=(m1<m2)or(m1=m2)and(d1<d2) end.

```

Присвойте переменной *m1*:

- 1) название месяца, следующего за месяцем *m2* (с учётом того, что за декабрём идёт январь);
- 2) название *k*-го месяца после месяца *m2*;
- 3) название *n*-го месяца года.

7.33. Установите, имеются ли ошибки в следующем фрагменте программы (объясните Ваше решение):

```

type Land=(Osterreich,Deutschland);
      Hauptstadt=(Wien,Berlin);
var a:Land;b:Hauptstadt;
begin a:=Osterreich;
      case a of
        Osterreich  :b:=Wien;
        Deutschland:b:=Berlin;
      end;
end.

```

7.34. Установите, какую работу выполняет программа:

```

uses crt,graph;
var  mode :integer;
      driver :integer;
      ch    :char;
      x,y,r :integer;

```

```

procedure bip; {Звуковой сигнал}
begin sound(1000); delay(1000); nosound end;

```

```

begin driver:=detect;
      initgraph(driver,mode,'c:\tp7\bgi');
      setbkcolor(3);
      setcolor(12);
      setlinestyle(0,0,3);

```



```

x:=getmaxx div 2;
y:=getmaxy div 2;
r:=20;
ch:=' ';
while ch<>#13 do
begin cleardevice;
  circle(x,y,r);
  ch:=readkey;
  if ch=#0 then ch:=readkey;
  case ch of
    #72:if y-5-r>=0 then y:=y-5 else bip;
    #80:if y+5+r<=getmaxy then y:=y+5 else bip;
    #75:if x-5-r>=0 then x:=x-5 else bip;
    #77:if x+5+r<=getmaxx then x:=x+5 else bip;
    '<':if r-5>0 then r:=r-5 else bip;
    '>':if r+5<getmaxx div 2 then r:=r+5 else bip;
  end;
end;
setcolor(5);
outtextxy(1,340,'нажмите Enter для выхода ...');
readln;
closegraph;
end.

```

## 8. Циклы

8.1. Выполните трассировку и ответьте на вопросы по заданному фрагменту программы:

```

for i:=1 to n do
begin j:=2;
  while j<=m do
  begin write(i+j);
    j:=j+2
  end;
  writeln
end.

```

- 1) Сколько раз будет выполнен оператор write(i+j)?
- 2) Сколько строчек будет напечатано?

8.2. Определите, сколько раз будет выполнено тело цикла в приведенном фрагменте программы:

```

s:=0; i:=1; read(n);
while i<=n do;
begin s:=s+i;
  i:=i+2;
end;

```

8.3. Задан фрагмент программы. Выполните трассировку и ответьте на вопрос: «Сколько раз будет выполнено тело цикла и какими будут значения переменных  $a, b$  и  $s$  после завершения этого фрагмента?»

```
a:=1; b:=1;
while a+b<8 do
begin a:=a+1;
      b:=b+2
end;
s:=a+b;
```

8.4. Нарисуйте блок-схему алгоритма, содержащего в цикле два идущих подряд условных блока.

8.5. Нарисуйте блок-схему алгоритма, содержащего в цикле два вложенных друг в друга условных блока.

8.6. Запишите оператор, эквивалентный условному с использованием любого оператора цикла.

8.7. Установите, какую задачу решают предложенные фрагменты программ (везде  $n$  – натуральное число). Ответьте на вопросы.

```
1) s:=0;
   i:=n;
   while i>0 do
   begin s:=s+i;
        i:=i-1
   end.
```

Запишите данный фрагмент с помощью операторов for... и repeat ... until.

```
2) s:=0;
   i:=1;
   j:=n;
   while i<=j do
   begin s:=s+i+j;
        i:=i+1;
        j:=j-1
   end;
   if n mod 2=1
   then s:=s-n mod 2+1.
```

В каких случаях выполнится условный оператор, записанный после цикла?

```
3) s:=0; i:=56; j:=1;
   while i<=56+2*(n-1) do
   begin s:=s+j;
        j:=j+1;
        i:=i+2
   end.
```

Можно ли в последнем фрагменте переменную  $j$  выразить через переменную  $i$  и убрать  $j$  из программы?

8.8. Правильно ли записаны на Паскале операторы цикла? Если «Да», то подсчитайте, сколько раз выполняется тело цикла. Если «Нет», объясните ошибки.

- 1) while true do i:=i+1;
- 2) for z:=1 to 2\*z do i:=i+1;
- 3) while false do i:=i-1;
- 4) for u:=2\*a downto -1 do i:=i-1;
- 5) for i:=j to j+1 do s:=s+i;
- 6) k:=0;  
for i:=2 downto k do k:=k\*2;
- 7) j:=56;  
for i:=-4 to j div 7 do j:=j-2;
- 8) j:=56;  
i:=-4;  
while i<=j div 7 do  
begin j:=j-2;  
i:=i-1  
end;
- 9) for i:=1 to n do a[i]:=2\*i+6;  
for a[1]:=1 to n do write(a[a[1]], '');
- 10) i:=5; while i>0 do i:=i+1;
- 11) i:=5; while i<=0 do i:=i-1;
- 12) i:=5; while i>=0 do i:=i-1;
- 13) i:=5; while i<=0 do i:=i+1;
- 14) i:=90; repeat i:=i-1 until i>0;
- 15) i:=90; repeat i:=i-1 until i<=0;
- 16) i:=9; repeat i:=i+1 until i>10;
- 17) i:=9; repeat i:=i+1 until i<=10;
- 18) i:=9; repeat i:=i-1 until false;
- 19) i:=9; repeat i:=i+1 until true;
- 20) i:=9; repeat i:=i\*2 until false and not false;
- 21) i:=9; repeat i:=i/2 until true or false;
- 22) i:=9; repeat i:=i/2 until true and false;
- 23) i:=9; repeat i:=i\*2 until not (i<9) and (i>=9);
- 24) i:=9; while (i<=100) and (i>100) do i:=i+1;
- 25) i:=9; while (i<=100) do i:=i div 2;
- 26) for i:=9 downto i div 2 do i:=i div 2;
- 27) while read(x) do write(x);
- 28) for i:=-1 to -10 do write(i+1);
- 29) i:=1; while i<2 do i:=j;
- 30) while x<1 do x:=x>0;
- 31) k:=0; for i:=1 to 8 do; k:=k+i;
- 32) g:=9; for i:=1.0 downto -1.0 do g:=g\*2;
- 33) g:=1; for i:=1 downto -16 do g:=g\*2;
- 34) for i,j:=1 to n do write('П Р И В Е Т !! ');

```

35) for i+j:=1 to n do read(a,b,c);
36) for k:=1 to 2*k do write('Ч Е П У Х А ');
37) x:=for i:=1 to 7 do write('Ну и что..?');
38) for if:=1 to 10 do write('then 1 else 2');
39) for x>0 to 35 do readln;
40) repeat write('ВСЕМ ПРИВЕТ!') until false;
41) x:=1; repeat x:=x+1 until 100;
42) repeat true until true;
43) x:=0; repeat x:=x+1 until x<0;
44) repeat for i:=1 until 100 do readln(n).

```

8.9. Установите, чему равен  $x$ :

```

1) x:=0;
   for i:=-5 to 0 do
       n:=i*i;
       x:=x+n;
2) n:=5;
   x:=0;
   for i:=1 to n do x:=i;
   writeln(x);
3) for i:=1 to 5 do
   for j:=1 to 5 do write('x');

```

Исправьте этот фрагмент так, чтобы вывод составил квадрат размером  $5*5$ ;

```

4) n:=0;
   while n<=10 do
   begin x:=n*n;
        writeln(n:6,' ',x:6)
   end;

```

Как нужно изменить этот фрагмент, чтобы были выведены числа от 1 до 10 и их квадраты. Как ту же задачу можно решить с помощью оператора for?

```

5) x:=0;
   while x<5 do x:=x+1;
6) x:=0;i:=0;
   while i<5 do i:=i+1; s:=s+1/i;
7) x:=0;i:=1;
   while i>1 do begin x:=x+1/i; i:=i-1 end;
8) x:=0 i:=1;
   repeat x:=x+1/i; i:=i-1 until i<=1;
9) x:=1; n:=1;
   for i:=2 to n do x:=x+1/i.

```

8.10. При каких целых значениях  $k$  следующие фрагменты программ вычисляют одно и то же значение  $y$  ( $x$  и  $y$  – real)?

```

1) y:=0;
   x:=(k+6)/2;
   while x<=(2*k+16)/2 do
   begin y:=y+abs(x);

```

```

        x:=x+1
    end;
    write(y:2:4);
2) y:=0;
   x:=(2*k+16)/2;
   while x>=(k+6)/2 do
   begin y:=y+abs(x);
        x:=x-1
   end;
   write(y:2:4).

```

8.11. Найдите ошибки во фрагментах программ:

- 1) var k,i:integer; x,y:real;  
begin y:=0;for x:=0.1 to 0.9 do y:=y+sin(x) end;
- 2) var k,i:integer; x,y:real;  
begin k:=81; y:=1 for i:=1 to sqrt(k) do y:=2\*y end;
- 3) var k,i:integer; x,y:real;  
begin k:=0; for i:=1 to 9 do k:=k+sqr(i); k:=k\*i end;
- 4) var k,i:integer; x,y:real;  
begin k:=1; for i:=1 to 64 do begin i:=2\*i; k:=k+1 end end.

8.12. Дана последовательность операторов:

```

a:=1; b:=1;
while a+b<8 do
begin a:=a+1;
      b:=b+2
end;
s:=a+b;

```

Сколько раз будет выполнен цикл и какими будут значения переменных  $a$ ,  $b$  и  $s$  после завершения этой последовательности операторов?

8.13. Какими будут значения переменных  $p$  и  $s$  после выполнения операторов:

```

p:=a; s:=0;
while p>0 do
begin s:=s+p mod 10;
      p:=p div 10
end.

```

Определите значение результата  $p$  при следующих значениях исходного данного: а)  $a=23$ ; б)  $a=32$ ; в)  $a=109$ .

8.14. Дана последовательность операторов, вычисляющих факториал  $f$  числа  $n$ :

```

k:=1; f:=0;
while k<n do
    f:=f*k;
    k:=k-1,

```

которая содержит пять ошибок. Найдите ошибки, если факториал числа  $n$  равен произведению всех чисел от 1 до  $n$ .

8.15. Укажите особенность следующего цикла: `while a<b do c:=a+b.`

8.16. Запишите фрагменты решения следующих задач на Паскале (укажите сходные черты и различия в решениях):

- 1)  $1+2+3+\dots+n$ ;
- 2)  $5+6+7+\dots+(n+5)$ ;
- 3)  $1+3+5+\dots+(2n-1)$ ;
- 4)  $2+4+6+\dots+2n$ ;
- 5)  $1+4+7+\dots+(3n-2)$ ;
- 6)  $1 \cdot 2 \cdot 3 \cdot \dots \cdot n$ ;
- 7)  $2 \cdot 4 \cdot 6 \cdot \dots \cdot 2n$ ;
- 8)  $\sin(1)+\sin(2)+\sin(3)+\dots+\sin(n)$ ;
- 9)  $n+(n-1)+\dots+2+1$ ;
- 10) сумму натуральных чисел отрезка  $[a;b]$ ;
- 11) сумму четных чисел отрезка  $[a;b]$ ;
- 12)  $\cos(n)+\cos(n-1)+\dots+\cos(2)+\cos(1)$ ;
- 13)  $\frac{1}{2} + \frac{1}{4} + \frac{1}{6} + \dots + \frac{1}{2n}$ ;
- 14)  $\frac{1}{2n-1} \cdot \frac{1}{2n-3} \cdot \frac{1}{2n-5} \cdot \dots \cdot \frac{1}{3} \cdot \frac{1}{1}$ ;
- 15)  $k \cdot (k+1) \cdot (k+2) \cdot \dots \cdot 2k$ ;
- 16)  $(n+m) \cdot (n+2m) \cdot (n+3m) \cdot \dots \cdot (n+m^2)$ ;
- 17)  $a+a \cdot k+a \cdot 2k+a \cdot 3k+\dots+a \cdot k^2$ ;
- 18)  $2+2^2+2^3+\dots+2^n$ ;
- 19)  $a/(a+1)/(a+2)/(a+3)/\dots/(a+n)$ .

8.17. Запишите каждое решение задания 8.16 всеми возможными операторами цикла на Паскале.

8.18. Запишите фрагменты решения следующих задач на Паскале (укажите сходные черты и различия в решениях):

- 1)  $(\dots(x+a)x+a)x+a+\dots+a)x+a$ ;
- 2)  $(\dots(n-p)n-p)n-p-\dots-p)n-p$ ;
- 3)  $(\dots(x+a)^9+a)^9+\dots+a)^9+a$ ;
- 4)  $(\dots(x+1)x+2)x+3)x+4)x+\dots+n$ ;
- 5)  $(\dots(y+n)y+(n-1))y+\dots+2)y+1$ ;
- 6)  $\cos(1+\cos(2+\cos(3+\dots+\cos(n)\dots)))$ ;
- 7)  $\sin(n+\sin(n-1+\dots+\sin(2+\sin(1)\dots)))$ ;
- 8)  $\sqrt{3+\sqrt{6+\sqrt{9+\dots+\sqrt{99}}}}$ ;

$$1 + \frac{2x}{2 + \frac{3x}{3 + \frac{4x}{4 + \frac{5x}{0 + \frac{nx}{n+1}}}}}$$

9)

8.19. Сколько раз будут выполнены операторы из тела циклов в следующих фрагментах программ:

- 1) for k:=-1 to 1 do writeln;
- 2) for k:=10 to 20 do read(n[k]);
- 3) for k:=20 to 10 do i:=i+k;
- 4) k:=5; r:=15;  
for i:=k+1 to r-1 do k:=k+r;
- 5) k:=r;  
for i:=k to r do write('УРА!').

8.20. Определите значение переменной *s* после выполнения следующих операторов:

```
s:=0; n:=10;
for i:=2 to n do s:=s+100 div i.
```

8.21. Установите, какую задачу решает каждый из предложенных фрагментов программ:

- 1) readln(n);  
s:=0;  
while n>0 do  
begin s:=s\*10+n mod 10;  
n:=n div 10  
end;
- 2) s:=0;  
for i:=1 to n do  
for j:=i+1 to n do s:=s-a[j,i];
- 3) s:=0;  
for i:=3 downto -2 do  
if i mod 2=0  
then s:=s-i  
else s:=s+i;
- 4) i:=p; j:=q;  
while i<j do  
begin r:=a[i]; a[i]:=a[j]; a[j]:=r;  
i:=i+1;  
j:=j-1  
end;

```

5) p:=1; i:=1; y:=x;
   while i<=n do
     begin y:=sin(y);
          p:=p*y;
          i:=i+2
     end;
6) s:=0;
   for i:=n downto 1 do
     begin p:=1;
          for j:=1 to i do p:=p*j;
          s:=s+p
     end;
7) i:=1; j:=n;
   while i<=j do
     begin read(a[i],a[j]);
          i:=i+1;
          j:=j-1
     end.

```

8) Укажите, как нужно расположить числа для ввода, чтобы после исполнения представленной ниже программой результат был бы упорядочен по возрастанию:

```

i:=1; j:=n;
while i<=j do
  if (i+j) mod 2=1
  then begin read(a[i]);i:=i+1 end
  else begin read(a[j]);j:=j-1 end.

```

8.22. Установите, являются ли оператором цикла следующие записи:

- 1) while x<0 do x:=x+0.5;
- 2) while x<0 do x:=x-100;
- 3) while 0<x<1 do y:=sqr(y+0.01);
- 4) while a>0 do y:=2\*y;
- 5) while a>b do a:=a-1; b:=b+1;
- 6) while true do write('a');
- 7) while false do write('b').

8.23. Может ли завершиться выполнение оператора цикла, начало которого выглядит так: while abs(x)+1>0.793 do ...?

8.24. Установите, для каких начальных значений переменной x завершится выполнение операторов цикла:

- 1) while x<1.3 do x:=sqr(x);
- 2) while abs(x)>=1 do x:=x-1;
- 3) while 2\*x>x do x:=x-1;
- 4) while sqr(x)>=0 do x:=sin(x)+1.315.

8.25. Пусть нужно получить значения функции  $\sin x$  для  $x=1, 1.1, 1.2, \dots, 2$ . Установите, какая программа решения этой задачи из двух приведенных ниже лучше (более надежна) и почему:



```

{программа 1}
var  x:real;
begin x:=1;
      while x<=2 do
      begin writeln(sin(x));
            x:=x+0.1
      end
end;
{программа 2}
var  x:real;
i:integer;
begin for i:=1 to 11 do
      writeln(sin(1+0.1*(i-1)));
end.

```

8.26. Студент решал задачу: «Натуральные числа вводятся в порядке возрастания до появления нуля (признак окончания ввода). Нужно напечатать факториалы всех введенных чисел. Программа должна быть эффективной, т.е. не повторять вычислений». Ниже приводятся два решения этой задачи – студента и преподавателя. Определите, какое решение удовлетворяет всем условиям задачи.

```

{Решение студента}
read(a);
while a<>0 do
begin p:=1;
      for i:=1 to a do p:=p*i;
      writeln(p);
      read(a)
end;
{Решение преподавателя}
read(a);
p:=1; k:=1;
while a<>0 do
begin for i:=k to a do p:=p*i;
      write(p);
      k:=a+1;
      read(a)
end.

```

8.27. Студент решал задачу: «Натуральные числа вводятся в порядке убывания до появления нуля (признак окончания ввода). Нужно найти сумму степеней 3, если введенные числа используются как показатели. Программа должна быть эффективной, т.е. не повторять вычислений, например, при вводе 9 7 6 4 0 должен быть получен ответ:  $3^9+3^7+3^6+3^4$ ». Ниже приводятся два решения этой задачи – студента и преподавателя. Определите, какое решение удовлетворяет всем условиям задачи.

```

{Решение студента}

```

```

s:=0;
read(a);
while a<>0 do
begin p:=1;
    for i:=1 to a do p:=p*3;
    s:=s+p;
    read(a);
end;
{Решение преподавателя}
read(a);
if a=0 then write('сумма=',0);
else begin    s:=0;
              p:=1;
              for i:=1 to a do p:=p*3;
              writeln(3,'^',a,'=',p);
              s:=s+p;
              b:=a;
              p1:=p;
              read(a);
              while a<>0 do
              begin p:=1;
                  for i:=1 to b-a do p:=p*3;
                  s:=s+p1 div p;
                  writeln(3,'^',a,'=',p1 div p);
                  p1:=p1 div p;
                  b:=a;
                  read(a);
              end
            end.

```

8.28. Василий решал задачу: «Найдите сумму  $s=1+2-3+4+5-6+7+8-9+\dots-99$ ». Далее приводится его решение этой задачи. Верно ли оно?

```

s1:=0; for i:=1 to 99 do s1:=s1+i;
s2:=0; i:=3;
while i<=99 do
begin s2:=s2+i;
    i:=i+3
end;
s:=s1-s2.

```

8.29. Решалась задача: «Найдите номер последнего по порядку элемента, совпадающего по величине с максимальным. Например, в массиве: 12 3 4 5 3 12 3 4 12 12 3 должен быть получен ответ 10». Студент представил приведенное ниже решение. Преподаватель утверждает, что это решение не верное. Установите, кто прав.

```

max:=1;
for i:=2 to n do

```

```

    if a[max]<a[i]
    then max:=i.

```

8.30. Другой студент представил решение задачи 8.29 в виде:

```

max:=1;
for i:=n downto 2 do
    if a[max]<a[i]
    then max:=i;

```

и получил отличную оценку. Прав ли преподаватель в этом случае?

8.31. Помогите исправить программу из 8.29, чтобы студент получил отличную оценку и было бы внесено минимальное количество изменений.

8.32. Установите, является ли приведенный фрагмент решением следующей задачи: «Найдите номер последнего по порядку минимального элемента массива»?

```

i:=1; j:=n;
while i<=j do      {**}
    if a[i]>=a[j] {*}
    then i:=i+1
    else j:=j-1;
    write(i).

```

Лаборант предлагает в строке, отмеченной звёздочкой, изменить знак сравнения на < (строго меньше), чтобы получить решение предложенной задачи. Прав ли он? Что получится, если в строке, отмеченной двумя звёздочками, изменить знак сравнения на < (строго меньше)?

8.33. Автор нижеследующей программы утверждает, что она моделирует «бегущую строку». Прав ли автор? Как можно изменить программу, чтобы строка пробегала от правой границы экрана до левой?

```

uses crt;
type stroka=string[160];
var vrod:stroka;
procedure gostring(x,y:byte;inst:stroka);
var st1:stroka;i:byte;
procedure s;
begin sound(1000);delay(50);nosound end;
begin st1:=' ';
    clrscr;
    st1:=st1+inst;
    for i:=1 to length(st1) do
    begin delete(st1,1,1);
        gotoxy(x,y);
        write(st1);
        s;
        delay(2000);
        delline;
    end;
end;

```

```
begin gostring(10,10,'Включите принтер!!!');
readln;
end.
```

8.34. Установите, что будет выведено на экран в результате выполнения следующей программы:

```
var n,x:integer;
begin n:=1;
  while n<=10 do
    begin x:=n*n;
      writeln(n:6,' ',x:6)
    end;
  end.
```

8.35. Измените программу из 8.34 так, чтобы она выводила в столбик числа от 1 до 10 и их квадраты.

8.36. Выполните трассировку и установите, какую задачу решает следующий фрагмент программы:

```
while x<>y do
begin while x>y do x:=x-y;
  while y>x do y:=y-x;
end.
```

8.37. Решалась задача: «Последовательность из  $n$  элементов задана формулой общего члена  $a_i = \sin(i + i/n)$ . Напишите программу, определяющую номер первого элемента последовательности большего  $Z$ ». Ниже предложены три варианта решения этой задачи. Объясните, какое решение Вам больше нравится и почему?

```
{Решение 1}
var  n:integer;    {количество элементов}
     z:real;       {искомый элемент}
     i:integer;    {номер элемента}
     f:boolean;    {признак окончания поиска: true - найдено}
begin write('Введите n и z');
  readln(n,z);
  i:=1;
  f:=false;
  while (i<=n) and not f do
    if sin(i+i/n)>z
    then f:=true
    else i:=i+1;
  if f
  then write('номер найденного элемента – ',i)
  else write('нужного элемента в последовательности НЕТ');
```

```
{Решение 2}
var  n:integer;    {количество элементов}
     z:real;       {искомый элемент}
     i:integer;    {номер элемента}
```

```

begin write('Введите n и z');
      readln(n,z);
      for i:=1 to n do
        if sin(i+i/n)>z
          then i:=n+1
          else i:=i+1;
      if i=n+1
        then write('последовательность содержит нужный элемент')
        else write('нужного элемента в последовательности НЕТ');
{Решение 3}
var   n:integer;   {количество элементов}
      z:real;      {искомый элемент}
      i:integer;   {номер текущего элемента}
      j:integer;   {номер искомого элемента}
begin write('Введите n и z');
      readln(n,z);
      i:=1;
      j:=0;
      while i<=n do
        if sin(i+i/n)>z
          then begin j:=i;i:=n+1 end
          else i:=i+1;
      if i=n+1
        then write('номер найденного элемента ',j)
        else write('нужного элемента в последовательности НЕТ');

```

8.38. 1) Исправьте решение 2 в задаче 8.37 так, чтобы получалось верное решение.

2) Исправьте все решения в задаче 8.37 так, чтобы перебор элементов осуществлялся от последнего элемента к первому.

3) Исправьте все решения в задаче 8.37 так, чтобы поиск нужного элемента осуществлялся только для элементов с номерами от  $p$  до  $q$ .

4) Укажите, для какого решения задачи 8.37 потребовалось вносить меньшее количество исправлений при выполнении пунктов 2 и 3 данной задачи. Сделайте выводы.

8.39. Установите, какую задачу решает приведенный ниже фрагмент программы:

```

s:=0;
while n>0 do
begin s:=s+n mod 10;
      n:=n div 10 end.

```

8.40. Исправьте программу так, чтобы избавиться от вложенного цикла:

```

s:=0;
for i:=1 to n do
begin p:=1;
      for j:=1 to i do p:=p*j;

```

```

    s:=s+p
end.
for i:=1 to n do b[i]:=a[n-i+1].

```

8.41. Найдите эквивалентные последовательности операторов:

- 1) for i:=1 to n do P(i);
- 2) i:=1; while i<=n do begin P(i); inc(i) end;
- 3) i:=1; repeat P(i); inc(i) until i>n;
- 4) for i:=1 to n do P(i); i:=n+1;
- 5) i:=0; while i<n do begin inc(i); P(i) end;
- 6) i:=0; repeat inc(i); P(i) until i>=n;
- 7) i:=n; while i>0 do begin P(i); dec(i) end.

8.42. Установите, какую задачу решает фрагмент программы. Изменится ли результат работы программы, если отмеченную комментарием точку с запятой заменить символом else? Почему? Перепишите фрагмент с использованием оператора while:

```

max:=a[1]; min:=a[1];
for i:=2 to n do
begin if max<a[i]
    then max:=a[i] ; {←эту точку с запятой замените else}
    if min>a[i]
    then min:=a[i]
end.

```

8.43. Установите, какую задачу решает приведённый фрагмент программы. Перепишите его с использованием оператора while:

```

s:=1; max:=a[1];
for i:=n downto 2 do
    if max<a[i]
    then begin max:=a[i]; s:=1 end
    else if max=a[i] then s:=s+1.

```

8.44. Какую задачу решает приведенный фрагмент программы? Каким соотношениям должны удовлетворять переменные  $a$ ,  $b$ ,  $c$ ,  $d$ , чтобы задача могла быть решена? Приведите все возможные соотношения.

```

s:=0; p:=1;
readln(a,b,c,d);
x:=a;
while x<=d do
begin if x<c
    then s:=s+x
    else if x>b
        then p:=p*x;
    x:=x+1
end.

```

8.45. Установите, что делает фрагмент программы. Перепишите его, используя оператор цикла с постусловием:

```

s:=0;

```

```

readln(a);
while a<>0 do
begin if a mod b=0
      then s:=s-a;
      read(a)
end.

```

8.46. Какая задача решается в предложенном фрагменте программы. Как изменится этот фрагмент, если эту же задачу нужно будет решить на отрезке  $[a;b]$ ?

```

s:=0; i:=2; while i<=n do begin s:=s+i; i:=i*2 end.

```

## 9. Массивы

9.1. Какие из приведенных описаний одномерных массивов являются неправильными и почему:

- а) var dim:array[-1..1] of real;
- б) type mas=array[char] of char;  
var beta:mas;
- в) type massiv=array['A'..'D'];  
var a,b:massiv;
- г) var vector:array[integer] of char;
- д) var mm:array[false..true] of char;
- е) type ss=array[-20..0] of integer;  
var mas=ss;
- ж) type tr=array[1..n,1..m] of real;  
var x,y,z:tr;
- з) var k:integer;  
pak:array[1..k] of integer;
- и) const n=50;  
type mas=array[0..n-1] of real;  
var p,r,s,t:mas;
- к) type city=(Kiev,Moskva,Perm,Minsk,Orel);  
var a:array[city] of real;
- л) type zif=array[true..false] of (1,2,3,4);  
x:zif;
- м) var ch:array[-2] of boolean;
- н) var sh:array[0..0] of integer;

9.2. Сколько элементов содержит каждый правильно описанный массив в задаче 9.1?

9.3. Какие значения могут принимать элементы каждого правильно описанного массива в задаче 9.1?

9.4. Как обратиться к первому и последнему элементам в каждом правильно описанном массиве в задаче 9.1?

9.5. Опишите регулярный тип ALPHA, содержащий разные литеры, а индексами которого являются двузначные числа?

9.6. Может ли массив содержать один элемент? Если может, то дайте его описание. Если не может, то объясните почему.

9.7. Присвойте нулевые значения всем элементам массива

`a:array['k'..'z'] of real.`

9.8. Какие из приведенных описаний двумерных массивов являются неправильными и почему:

а) `var a:array[1..5,1..7] of real;`

б) `type mas=array['a'..'d',-2..3] of integer;`  
`var arr:mas;`

в) `type mm=array[1..10] of char;`  
`var a:array[2..7] of mm;`

г) `var w:array[5,7] of integer;`

д) `type mask=(loup,lapin,herisson);`  
`a:array[mask] of array[1..5] of char;`

е) `var mass:array[1..5],[1..7] of real;`

9.9. Сколько компонентов содержит каждый из описанных ниже массивов?

а) `var mb:array['a'..'c',2..5] of real;`

б) `var mc:array[false..true] of array[1..5] of char;`

в) `type ww=(m1,m2,m3,m4);`  
`var zar:array[ww] of array[ww] of boolean.`

9.10. Используя описания массивов из задачи 9.9, определите, какие из приведенных обращений к элементам массивов являются неправильными и почему:

а) `mb['b',3],`

б) `mb['b'][6],`

в) `mc[true,2],`

г) `mc[3,false],`

д) `zar[m2,m1],`

е) `zar[true,m3].`

9.11. Если массив описан так:

`type mas=array[1..3] of array [1..5] of integer;`  
`var a,b,c:mas;;`

то возможны ли присваивания

`a[2]:=b[1]; a[i][j]:=c[i,j]?`

9.12. Опишите регулярный тип, объединяющий в себе массивы, элементами которых являются натуральные числа, а индексами - любые литеры.

9.13. Можно ли во время исполнения изменять размер массива (количество элементов в нем)?

9.14. Могут ли элементами некоторого массива быть числа: 1, 1.41, 1.73 и 2?

9.15. Может ли тип элементов массива быть любым?

9.16. Может ли типом индекса массива быть тип `integer` или `real`?

9.17. Дано:

`const n=101;`  
`var a:array[1..n] of real;`

а) напишите операторы для ввода элементов в массив *a*;



б) напишите операторы для заполнения массива *a* случайными числами в интервале от -16 до 73;

в) напишите операторы для вывода элементов массива *a* в строку, разделяя их двумя пробелами;

г) напишите операторы для вывода элементов массива *a* в строку, разделяя их таким количеством пробелов, каков порядковый номер выводимого элемента;

д) напишите операторы вывода элементов массива в обратном порядке по *n* чисел в строке.

9.18. Дано:

```
var a:array [1..50] of real;  
    b:array [0..49] of real.
```

Перепишите элементы массива *a* в массив *b*. Можно ли эту задачу решить с помощью этого оператора *a=b*?

9.19. Чем одномерный массив отличается от двумерного?

9.20. Почему массивы называются структурами данных с прямым доступом?

9.21. Приведите примеры программ, где прямой доступ необходим.

9.22. Чем массивы отличаются от строк?

9.23. Если массив описан так:

```
type mas=array[1..3] of array [1..5] of integer;  
var a,b,c:mas;
```

то возможны ли присваивания *a[2]:=b[1]*; *a[i][j]:=c[i,j]*?

9.24. Сколько неизмененных элементов в массиве останется после выполнения фрагмента программы: *for i:=1 to n do a[i]:=a[i]/a[5]*?

9.25. Установите, какая задача решается в предложенном фрагменте программы:

```
s:=0;  
for i:=1 to n do  
  if (1<=a[i]) and (a[i]<=n)  
    then s:=s+a[a[i]].
```

9.26. Установите, какая задача решается в предложенном фрагменте программы. Назовите все операторы и типы данных, использованные в нем, расскажите как они работают:

```
const n=10;  
var   u,v:array [1..n] of integer;  
      w :array [0..n] of integer;  
      j,k,t:integer;  
begin k:=0;  
  for j:=n downto 1 do  
    begin t:=u[j]+v[j]+k;  
          w[j]:=t mod 10;  
          k:=t div 10;  
    end;  
  w[0]:=k  
end.
```

9.27. Установите, какая задача решается в предложенном фрагменте программы. Назовите все операторы и типы данных, использованные в нем, расскажите как они работают:

```
const n=10;
var   u,v,w:array [1..n] of integer;
      j,k,t:integer;
begin k:=0;
      for j:=n downto 1 do
        begin t:=u[j]-v[j]+k;
              w[j]:=t mod 10;
              k:=t div 10;
        end;
      end.
```

9.28. Задан массив  $A = \begin{pmatrix} 2 & 1 & 3 \\ 3 & 3 & 1 \\ 1 & 2 & 1 \end{pmatrix}$ . Установите, каким будет значение  $C$

после выполнения операторов:

```
for i:=1 to 3 do
  for j:=1 to 3 do
    C[i,j] := A[A[i,j],A[j,i]].
```

Существенен ли порядок, в котором выбираются значения индексов  $i$  и  $j$ ?

9.29. Установите, какая задача решается в предложенном фрагменте программы. Назовите все операторы и типы данных, использованные в нем, расскажите как они работают:

```
for i:=1 to n do
  for j:=1 to m div 2 do
    begin r:=a[i,j]
          a[i,j]:=a[i,n-j+1];
          a[i,n-j+1]:=r
    end.
```

9.30. Установите, какая задача решается в предложенном фрагменте программы. Назовите все операторы и типы данных, использованные в нем, расскажите как они работают:

```
for i:=p to q-(p-q+1) div 2 do
  begin r:=a[i];
        a[i]:=a[q-i+p];
        a[q-i+p]:=r
  end.
```

9.31. Установите, какая задача решается в предложенном фрагменте программы. Назовите все операторы и типы данных, использованные в нем, расскажите как они работают:

```
for i:=1 to n do
  for j:=1 to n do
    a[i,j]:=(i div j) * (j div i).
```

9.32. Установите, какая задача решается в предложенном фрагменте программы. Назовите все операторы и типы данных, использованные в нем, расскажите как они работают:

```

for i:=1 to n-1 do
  for j:=i+1 to n do
    if a[i]>a[j]
    then begin r:=a[i];
           a[i]:=a[j];
           a[j]:=r
        end.

```

9.33. Какую задачу решает предложенный фрагмент алгоритма? Определите значение массива  $A$  после выполнения следующих операторов при  $N=3$  и

$$A = \begin{pmatrix} 5 & 7 & 8 \\ 5 & 5 & 7 \\ 6 & 3 & 6 \end{pmatrix};$$

```

for i:=1 to n-1 do
  for j:=i+1 to n do
    begin r:=a[i,j];
          a[i,j]:=a[j,i];
          a[j,i]:=r
    end.

```

9.34. Какую задачу решает предложенный фрагмент программы? Определите значение массива  $C$  после выполнения следующих операторов при

$$N=3, M=2 \text{ и } A = \begin{pmatrix} 6 & 1 \\ 1 & 0 \\ 6 & 4 \end{pmatrix};$$

```

for j:=1 to m do
  begin s:=0;
        for i:=1 to n do s:=s+a[i,j];
        c[j]:=s
  end.

```

9.35. Какую задачу решает предложенный фрагмент программы? Определите значение массива  $C$  после выполнения следующих операторов для

$$\text{заданных массивов } A = \begin{pmatrix} 2 & 4 & 6 \\ 7 & 1 & 1 \\ 9 & 3 & 5 \end{pmatrix} \text{ и } B = \begin{pmatrix} 0 & 4 & 6 \\ 3 & 2 & 1 \\ 4 & 3 & 5 \end{pmatrix};$$

```

for i:=1 to n do
  for j:=1 to n do
    if a[i,j]=b[i,j]
    then c[i,j]:=1
    else c[i,j]:=0.

```

9.36. Какую задачу решает предложенный фрагмент программы? Определите значение массива  $A$  после выполнения следующих операторов при  $N=8$ ,  $K=2$  и  $A=(4, -3, 5, -2, 3, 10, 9, 0)$ :

```
n:=n-1;
for i:=k to n do a[i]:=a[i+1].
```

9.37. Какую задачу решает предложенный фрагмент программы? Определите значение массива  $A$  после выполнения следующих операторов при  $M=3$ ,  $N=5$ ,  $D=6$  и  $A=(5, 2, -8, 1, -3)$ :

```
i:=m;
while i<=n-1 do
begin a[i]:=a[i+1];
      i:=i+1
end;
a[n]:=d.
```

9.38. Какую задачу решает предложенный фрагмент программы? Определите значение массива  $B$  после выполнения следующих операторов при  $N=7$  и  $A=(1, 6, 7, 3, -2, 8, 9)$ :

```
for i:=1 to n do b[i]:=a[n-i+1].
```

9.39. Какую задачу решает предложенный фрагмент программы? Определите значение массива  $P$  после выполнения следующих операторов при

$M=3$ ,  $N=3$  и  $A = \begin{pmatrix} 5 & 2 & 6 \\ 5 & 3 & 7 \\ 3 & 2 & 4 \end{pmatrix}$ :

```
for i:=1 to n do
begin p[i]:=0;
  for j:=1 to m do
    if a[i,j]=5
    then p[i]:=p[i]+1
  end.
end.
```

9.40. Какую задачу решает предложенный фрагмент программы? Определите значение массива  $C$  после выполнения следующих операторов при  $M=9$  и  $C=(4, 5, 3, 2, 2, 3, 5, 8, 7)$ :

```
k:=trunc(m/2);
for i:=1 to k do
begin q:=c[i];
  c[i]:=c[m-i+1];
  c[m-i+1]:=q
end.
```

9.41. Какую задачу решает предложенный фрагмент программы? Определите значение массива  $A$  после выполнения следующих операторов при  $N=8$ ,  $K=3$  и  $A=(3, -1, 9, -3, 4, 12, 3, 7)$ :

```
n:=n-1;
for i:=k to n do a[i]:=a[i+1].
```

9.42. Какую задачу решает предложенный фрагмент программы? Определите значение массива  $A$  после выполнения следующих операторов при  $M=5$ ,  $N=3$ ,  $D=0$  и  $A=(0, 4, -3, 5, -1)$ :

```
i:=m;
while i>=n do
begin a[i+1]:=a[i];
      i:=i-1
end;
a[n]:=d;
m:=m+1.
```

9.43. Какую задачу решает предложенный фрагмент программы? Определите значение массива  $S$  после выполнения следующих операторов при

$$A = \begin{pmatrix} 6 & 7 & 8 & 9 & 2 \\ 0 & 6 & 4 & 8 & 1 \\ 7 & 8 & 6 & 3 & 2 \end{pmatrix}$$

$N=5$  и :

```
for i:=1 to n do s[i]:=a[i,i].
```

9.44. Какую задачу решает предложенный фрагмент программы? Определите значение переменной  $S$  после выполнения следующих операторов при  $N=6$ ,  $T=5$  и  $A=(-4, 0, 3, 9, 5, 2)$ :

```
s:=0;
k:=0;
for i:=1 to n do
  if a[i]>t
  then begin s:=s+a[i];
           k:=k+1
        end;
if k>0 then s:=s/k.
```

9.45. Какую задачу решает предложенный фрагмент программы? Определите значение массива  $C$  после выполнения следующих операторов для

заданных массивов  $N=3$ ,  $A = \begin{pmatrix} 4 & 2 & 1 \\ 3 & 3 & 5 \\ 5 & 2 & 4 \end{pmatrix}$  и  $B = \begin{pmatrix} 5 & 3 & 1 \\ 1 & 3 & 2 \\ -5 & 3 & 0 \end{pmatrix}$ :

```
for i:=1 to n do
  for j:=1 to n do
    if a[i,j]>b[i,j]
    then c[i,j]:=a[i,j]-b[i,j]
    else c[i,j]:=b[i,j]-a[i,j].
```

9.46. Сформулируйте задачу, решаемую в предложенном фрагменте программы, где  $a[i,j]$  – элемент массива размерности  $N \times M$ . Определите

$$A = \begin{pmatrix} 6 & -8 & 5 & 0 & 1 \\ -4 & 6 & 4 & 3 & 8 \end{pmatrix}$$

значение массива  $A$  при  $N=4, M=5, K=3$ ,

:

for i:=1 to n do

for j:=k to m-1 do  $a[i,j]:=a[i,j+1]$ .

9.47. Сформулируйте задачу, решаемую в предложенном фрагменте программы, где  $a[i,j]$  – элемент массива размерности  $N \times N$ . Определите

$$A = \begin{pmatrix} -3 & 2 & 1 & 3 & -2 \\ 1 & -4 & 0 & 0 & 2 \\ -4 & 6 & 4 & 3 & 8 \\ 3 & 7 & 0 & 1 & 4 \end{pmatrix}$$

значение переменной  $P$  при  $N=5$ ,

:

p:=1;

for i:=1 to n do

if  $a[i,n-i+1]>0$  then  $p:=p*a[i,n-i+1]$ .

9.48. Сформулируйте задачу, решаемую в предложенном фрагменте программы, где  $a[i,j]$  – элемент массива размерности  $N \times N$ . Определите

$$A = \begin{pmatrix} -4 & 0 & 0 & 2 \\ 1 & 4 & 3 & 4 \end{pmatrix}$$

значения переменных  $K$  и  $P$  при  $N=4$ ,

:

p:=1;k:=0;

for i:=1 to n do

for j:=n-i+1 to n do

if  $a[i,j]>0$

then begin  $p:=p*a[i,j]$ ;

k:=k+1 end.

9.49. Сформулируйте задачу, решаемую в предложенном фрагменте программы, где  $a[i,j]$  – элемент массива размерности  $N \times N$ . Определите значение переменной  $S$  для исходных данных задачи 9.48:

i:=n;

s:=0;

while i>=1 do

begin if  $a[i,i]>0$

then  $s:=s+a[i,i]$ ;

i:=i-1

end.

9.50. Сформулируйте задачу, решаемую в предложенном фрагменте программы, где  $a[i,j]$  – элемент массива размерности  $N \times M$ . Определите

$$A = \begin{pmatrix} -7 & 6 & 5 & 1 & 1 \\ 3 & 5 & 9 & 3 & 2 \end{pmatrix}$$

значение массива  $R$  при  $N=5, M=4$ , :

```
for i:=1 to n do
begin r[i]:=1;
  for j:=1 to m do r[i]:=r[i]*a[i,j];
  r[i]:=exp(1/m*ln(r[i]))
end.
```

9.51. Сформулируйте задачу, решаемую в предложенном фрагменте программы, где  $a[i,j]$  – элемент массива размерности  $N \times M$ . Определите

$$A = \begin{pmatrix} 5 & 4 & 3 & 5 & 4 \\ 4 & 5 & 7 & 8 & 5 \end{pmatrix}$$

значение массива  $A$  при  $N=5, M=4$ , :

```
for i:=1 to n do
begin f:=a[i,m-1];
  a[i,m-1]:=a[m,i-1];
  a[m,i-1]:=f
end.
```

9.52. Сформулируйте задачу, решаемую в предложенном фрагменте программы, где  $a[i,j]$  – элемент массива размерности  $N \times M$ . Определите значение массива  $P$  для исходных данных задачи 9.51:

```
for i:=1 to n do
begin p[i]:=a[i,1];
  for j:=2 to m do
    if a[i,j]>p[i]
    then p[i]:=a[i,j]
  end.
end.
```

9.53. Сформулируйте задачу, решаемую в предложенном фрагменте программы, где  $a[i,j]$  – элемент массива размерности  $N \times N$ . Определите значение массива  $A$  для исходных данных задачи 9.51:

```
for i:=1 to n do
  for j:=1 to trunc(n/2) do
    begin c:=a[i,j];
      a[i,j]:=a[i,n-j+1];
      a[i,n-j+1]:=c
    end.
```

9.54. Сформулируйте задачу, решаемую в предложенном фрагменте программы, где  $a[i,j]$  – элемент массива размерности  $N \times N$ . Определите значение массива  $A$  для исходных данных задачи 9.48:

```
q:=1;
for i:=1 to n do
  for j:=1 to n do q:=q+a[i,j];
  q:=exp(exp(2*ln(1/n))*ln(q)).
```

9.55. Сформулируйте задачу, решаемую в предложенном фрагменте программы, где  $a[i,j]$  – элемент массива размерности  $N \times N$ . Определите

$$A = \begin{pmatrix} 0 & 5 & 2 & 9 \\ 3 & 4 & 5 & -5 \end{pmatrix}$$

значение переменной  $P$  при  $N=4$ , :

```
p:=1;k:=0;
for i:=1 to n do
  for j:=n-i+1 to n do
    if a[i,j]>0
      then begin p:=p*a[i,j];
              k:=k+1
            end;
  if k>0
    then p:=p/k.
```

9.56. Сформулируйте задачу, решаемую в предложенном фрагменте программы, где  $a[i,j]$  – элемент массива размерности  $N \times N$ . Определите

$$A = \begin{pmatrix} 2 & 4 & 6 & 8 & 0 \\ 3 & 1 & 1 & 3 & 5 \\ 4 & 6 & 7 & 6 & 9 \end{pmatrix}$$

значение массива  $A$  при  $N=5$ ,  $K=3$ , :

```
for i:=k to n-1 do
  for j:=1 to n do a[i,j]:=a[i+1,j].
```

9.57. Сформулируйте задачу, решаемую в предложенном фрагменте программы, где  $a[i,j]$  – элемент массива размерности  $N \times N$ . Определите

значение массива  $A$  при  $N=4$ ,  $K=2$ ,  $A = \begin{pmatrix} 5 & 7 & 7 & 0 \\ 4 & 4 & 6 & 3 \\ 2 & 2 & 5 & 0 \\ 0 & 5 & 2 & 3 \end{pmatrix}$  :

```
for i:=1 to n do
begin a[i,k]:=0;
      a[k,i]:=0
end.
```



9.58. Сформулируйте задачу, решаемую в предложенном фрагменте программы, где  $a[i]$  – элемент массива размерности  $N$ . Определите значение переменной  $D$ , если  $N=5$ ,  $A=(-2, 3, 4, 6, -5)$ :

```
d:=1;
for i:=1 to n do
  if a[i]>0
    then if a[i]/3=trunc(a[i]/3)
      then d:=d*a[i].
```

9.59. Сформулируйте задачу, решаемую в предложенном фрагменте программы, где  $a[i]$  – элемент массива размерности  $N$ . Определите значение массива  $B$ , если  $N=4$ ,  $A=(1, 3, 5, -1)$ :

```
s:=0;
for i:=1 to n do
  begin s:=s+a[i];
        b[i]:=s/i
  end.
```

9.60. Сформулируйте задачу, решаемую в предложенном фрагменте программы, где  $a[i]$  – элемент массива размерности  $N$ . Определите значение переменных  $Q$  и  $K$ , если  $N=5$ ,  $A=(4, -1, 5, 7, 2)$ :

```
q:=a[q];k:=1;
for i:=2 to n do
  if a[i]>q
    then begin q:=a[i];
              k:=i
    end.
```

9.61. Сформулируйте задачу, решаемую в предложенном фрагменте программы, где  $a[i]$  – элемент массива размерности  $N$ . Определите значения переменных  $C$  и  $G$ , если  $N=5$ ,  $A=(-2, 5, 0, 8, -3)$ :

```
c:=0;g:=0;
for i:=1 to n do
  if a[i]>0
    then begin c:=c+a[i]; g:=g+1
    end.
```

9.62. Сформулируйте задачу, решаемую в предложенном фрагменте программы, где  $a[i]$  – элемент массива размерности  $N$ . Определите значение переменной  $M$ , если  $N=7$ ,  $A=(-2, 3, 4, -4, 6, 5, 0)$ :

```
m:=0;
for i:=1 to n do
  if a[i]>0
    then if a[i]/2=trunc(a[i]/2) then m:=m+1.
```

9.63. Сформулируйте задачу, решаемую в предложенном фрагменте программы, где  $a[i]$  – элемент массива размерности  $N$ . Определите значение массива  $B$ , если  $N=5$ ,  $A=(4, 5, 6, 7, 8)$ :

```
for i:=1 to n do b[i]:=a[n-i+1].
```

9.64. Найдите все ошибки в программе. Ответ объясните:

```

uses crt;
type rur=(a1,a2,a3,a4,a5,a6,a7,a8);
  mas=array[rur]of integer;
  mas1=array[(b1,b2,b3,b4,b5,b6,b7,b8)]of integer;
var a:mas;
  t:rur;
  b:mas1;
begin clrscr;
  for t:=a1 to a8 do begin a[t]:=random(10);write(a[t],' ') end;
  writeln;
  for t:=a8 downto a1 do write(a[t],' ');writeln;
  b[b1]:=1;
  write(a[a1]>b[b1]);writeln;
  t:=a4;
  while t<=a7 do
  begin write(ord(t),' ');
    t:=succ(t)
  end;
  readln;
end.

```

9.65. Задан массив  $A = \begin{pmatrix} 2 & 1 & 3 \\ 3 & 3 & 1 \\ 1 & 2 & 1 \end{pmatrix}$ . Установите, каким будет значение  $A$

после выполнения операторов:

```

for i:=1 to 3 do
  for j:=1 to 3 do
    A[i,j] := A[A[i,j],A[j,i]].

```

Существен ли порядок, в котором выбираются значения индексов  $i$  и  $j$ ?

9.66. Задан массив  $A = \begin{pmatrix} 2 & 1 & 3 \\ 3 & 3 & 1 \\ 1 & 2 & 1 \end{pmatrix}$ . Установите каким будет значение  $A$  после

выполнения операторов:

```

for i:=3 downto 1 do
  for j:=3 downto 1 do
    A[i,j] := A[A[i,j],A[j,i]].

```

Существен ли порядок, в котором выбираются значения индексов  $i$  и  $j$ ?

9.67. Задан одномерный массив  $A=(7, 5, 4, 6, 3, 2, 1)$ . Какое значение будет выведено в результате выполнения программы:

```

var a:array[1..7] of integer;
  i,j,k,c:integer;
begin write('Введите семь элементов массива ');
  for k:=1 to 7 do read(a[k]);
  i:=3; c:=0;

```

```

    for k:=1 to 100 do
    begin j:=a[i]; a[i]:=i; i:=j; c:=c+i end;
    write(c)

```

end.

9.68. Какое значение будет выведено в результате выполнения программы:

```

var i,k,k1,c : longint;
begin k:=3; k1:=1; c:=80;
    for i:=64 to 174 do
    begin k1:=-k1; k:=k+k1; c:=c+k*i end;
    write(c)

```

end.

9.69. Какое значение будет выведено в результате выполнения программы:

```

var x,y,d : integer;
begin x:=5; y:=2;
    if (y>x+2) or (y>0)
    then d:=x+2
    else d:=y-x;
    write(d);
    d:=5

```

end.

9.70. Какое значение будет выведено в результате выполнения программы:

```

var x1,y1,d1 : real;
begin x1:=1.5; y1:=1.5; d1:=0;
    if y1<0.3
    then d1:=x1-y1
    else d1:=d1-x1;
    write(d1)

```

end.

9.71. Какое значение нужно ввести при исполнении программы, чтобы в результате было выведено значение 128?

```

var i,n,p : longint;
begin write('Введите n ');
    readln(n);
    p:=1
    for i:=0 to n do p:=p*2;
    write(p)

```

end.

9.72. В какой последовательности нужно расположить перечисленные операторы, чтобы вычислить квадрат натурального числа  $n$  без использования операции умножения, причём общее число действий должно быть порядка  $n$ ?

- 1)  $kp:=1$ ,
- 2) while not ( $k=n$ ) do,
- 3) readln( $n$ ),
- 4)  $k:=k+1$ ,
- 5)  $ks:=ks+k+kp$ ,

- 6)  $kp:=k+1$ ,
- 7)  $k:=0$ ,
- 8)  $ks:=0$ ,
- 9)  $write(k, ', ', ks)$ .

9.73. Дан одномерный массив  $A=(1, 0, 1, 0, 0, 1, 1, 0)$ . Какими должны быть пропущенные операторы, чтобы в результате выполнения программы массив был преобразован к виду  $A=(0, 0, 0, 0, 1, 1, 1, 1)$ ?

```
var a : array[1..8] of integer;
    i,j,n,m,p : integer;
begin i:=1; m:=8; j:=m;
    while i<j do
        begin n:=0;
            if a[i]=0
            then begin n:=1; i:=? end;
            if a[j]=1
            then begin n:=1; j:=? end;
            if n=0
            then begin p:=a[i]; a[i]:=a[j]; a[j]:=p;
                    i:=i+1; j:=j-1
                end
            end
        end
    end.
```

## 10. Записи и другие структуры данных

10.1. Какими типами будете пользоваться при обработке данных следующего вида:

- 1) население Перми: район и количество жителей;
- 2) средний вес картины Васнецова: название и вес;
- 3) наиболее часто встречающаяся буква в каждом предложении заданного текста;
- 4) сколько раз указанная буква встречается в данном тексте;
- 5) время суток в часах, минутах и секундах;
- 6) дата рождения: число, месяц и день;
- 7) цены в магазине в рублях и копейках;
- 8) показания термометра;
- 9) количество учеников в школе: класс и количество учеников;
- 10) адрес: город, улица, дом, квартира?

10.2. Верно ли, что все поля записи должны быть разных типов?

10.3. Почему при описании записи ее поля могут перечисляться в любом порядке?

10.4. Верно ли, что названия полей записи могут совпадать с именами переменных, констант и других объектов программы, но не могут совпадать с названиями полей других записей?

10.5. Почему в уточненном имени (конструкция  $r.f$ ) имя поля ( $f$ ) должно указываться явно и не может быть задано в виде выражения?

10.6. Установите, допустимы ли описания:

```
type point1=array[(x,y)] of real;  
      point2=record x,y : real end;  
var   p1:point1; p2:point2; d : real;  
begin d:=sqrt(sqr(p1[x]-p2.x)+sqr(p1[y]-p2.y)) end.
```

1) Какое значение  $d$  получится?

2) Допустимы ли конструкции  $p1[\text{succ}(x)]$  и  $p2.\text{succ}(x)$ ?

3) Допустимо ли описание выпуклого многоугольника, заданного координатами вершин в порядке их обхода по часовой стрелке в виде

```
var   a = array[1..n] of point1;  
      b = array[1..n] of point2;
```

4) Какой фрагмент программы позволит найти периметр выпуклого многоугольника?

```
{фрагмент 1}  
s:=0;  
for i:=1 to n-1 do s:=s+sqrt(sqr(a[i][x]-a[i+1][x])+sqr(a[i][y]-a[i+1][y]));  
s:=s+sqrt(sqr(a[1][x]-a[n][x])+sqr(a[1][y]-a[n][y]));  
{фрагмент 2}  
s:=0;  
for i:=1 to n-1 do s:=s+sqrt(sqr(a[i,x]-a[i+1,x])+sqr(a[i,y]-a[i+1,y]));  
s:=s+sqrt(sqr(a[1,x]-a[n,x])+sqr(a[1,y]-a[n,y]));  
{фрагмент 3}  
s:=0;  
for i:=1 to n-1 do s:=s+sqrt(sqr(b[i].x-b[i+1].x)+sqr(b[i].y-b[i+1].y));  
s:=s+sqrt(sqr(b[1].x-b[n].x)+sqr(b[1].y-b[n].y));  
{фрагмент 4}  
var c:array[1..2,1..n] of real;  
...  
s:=0;  
for i:=1 to n-1 do s:=s+sqrt(sqr(c[1,i]-c[1,i+1])+sqr(c[2,i]-c[2,i+1]));  
s:=s+sqrt(sqr(c[1,1]-c[1,n])+sqr(c[2,1]-c[2,n]));
```

10.7. Для описания учеников, составляющих класс, использовалась следующая структура данных:

```
type schuler = record fam:string[15]; {фамилия ученика}  
                    b1,b2,b3,b4,b5: 2..5; {баллы за дисциплину}  
                    sb:real; {средний балл}  
end;  
var klass:array[1..25] of schuler.
```

Какой фрагмент программы правильно вводит массив klass?

```
{фрагмент 1}  
for i:=1 to 25 do  
begin readln(klass[i].fam);  
      readln(klass[i].b1);  
      readln(klass[i].b2);  
      readln(klass[i].b3);
```

```

        readln(klass[i].b4);
        readln(klass[i].b5);
    end;
    {фрагмент 2}
    for i:=1 to 25 do
        with klass[i] do
            begin readln(fam);
                readln(b1,b2,b3,b4,b5)
            end;
        {фрагмент 3}
        for i:=1 to 25 do
            begin readln(klass[i].fam);
                with klass[i] do
                    begin b1:=2+random(4);
                        b2:=2+random(4);
                        b3:=2+random(4);
                        b4:=2+random(4);
                        b5:=2+random(4);
                    end
                end;
            end;
        {фрагмент 4}
        s:='абвгдеёжзийклмнопрстуфхцшщъыьэюя';
        for i:=1 to n do
            with klass[i] do
                begin fam:="";
                    d:=1+random(25); {длина фамилии}
                    for j:=1 to d do
                        begin m:=1+random(33);
                            fam:=fam+s[m]
                        end;
                    readln(b1,b2,b3);
                    b4:=2+random(4);
                    readln(b5)
                end.
            end.
        end.
    end.

```

Дополните правильные фрагменты так, чтобы вычислялся и заносился в структуру средний балл каждого ученика.

10.8. Дано время, описанное следующим образом:

```
type time = record h:0..23; m,s:0..59 end;
```

Напишите программу:

1) проверяющую, предшествует ли момент времени  $t_1$  моменту времени  $t_2$  (в рамках суток);

2) увеличивающую момент времени  $t_1$  на одну секунду (учтите смену суток).

10.9. Дана дата, описанная следующим образом:

```
type year = 1500..2500;
```

```

month = 1..12;
day = 1..31;
data = record y:year; m:month; d:day end;

```

Напишите программу:

- 1) проверяющую, предшествует ли дата  $d1$  дате  $d2$ ;
- 2) увеличивающую дату на один день (учтите смену месяца и года);
- 3) получающую дату, которая наступит через  $m$  дней;
- 4) получающую дату, которая была за  $m$  дней до сегодняшнего дня;
- 5) вычисляющую количество суток между датами  $d1$  и  $d2$ ;
- 6) день недели, выпадающий на дату  $d1$ , если известно, что в первый день нашей эры был понедельник.

10.10. Установите, правильно ли записан фрагмент программы? Объясните почему. Если фрагмент правильный, то какую работу он выполняет? Если – нет, то как его исправить?

```

type katja = record s:real;
                    n:integer;
                    i:integer;
                    a:array[1..100] of real
                end;
var a:katja;
begin write('Сколько элементов массива примут участие в вычислениях
(<100)');
    readln(a.n);
    for a.i:=1 to a.n do
        with a do readln(a[i]);
        a.s:=0;
        for a.i:=1 to a.n do a.s:=a.s+a.a[a.i];
        write(a.s);
    end.

```

10.11. Исправьте ошибки в программе:

```

type katja = record s:real;
                    n:integer;
                    i:integer;
                    a:array[1..100] of real
                end;
var a:katja;
    i:integer;
begin write('Сколько элементов массива примут участие в вычислениях
(<100)');
    readln(a.n);
    for i:=1 to a.n do
        with a do readln(a[i]);
        a.s:=0;
        for i:=1 to a.n do a.s:=a.s+a.a[i];
        write(a.s:7:3);
    end.

```

end.

10.12. Факультет университета описывается следующими структурами:

```
type year = 1500..2500;
    month = 1..12;
    day = 1..31;
    data = record y:year; m:month; d:day end; {дата}
    nom = record n:char; nn:1000..9999 end; {номер зачетной книжки}
    spec = (asu, at, evt, kres, ktei);
    student = record    fam:string[18];
                        dr:data; {дата рождения}
                        nz:nom;
                        stip:boolean; {получает ли стипендию}
                        rost:integer; {рост в сантиметрах}
                        oc:array[1..5] of 2..5; {оценки за последнюю
сессию}
                        sr:real; {средний балл за весь период обучения}
    end;
    группа = record    imja : record a:spec;
                        god:00..99;
                        np:1..5
                        end; {название группы}
                        spis:array[1..30] of student; {список
студентов}
                        fs:student; {фамилия старосты}
    end;
    fac = array [spec,1..5] of группа;
```

Напишите программу, которая устанавливает названия тех групп факультета, в которых больше половины студентов с ростом выше 175 сантиметров получают стипендию и имеют средний балл не менее 4.5, при условии, что за последнюю сессию они не имеют четверок.

10.13. Установите, что напечатает программа:

```
type aa = record
    b1:record c1,c2:integer end;
    b2:array[1..3]of char;
end;
var a:aa;
    i:integer;
begin a.b1.c2:=1;
    a.b1.c1:=2;
    a.b2[1]:='3';
    a.b2[2]:='5';
    a.b2[3]:='4';
    with a do
    begin with b1 do write(c1,' ',c2,'z');
        for i:=1 to 3 do write(b2[i],' ');
```



```

        end;
    end.
10.14. Пусть описаны данные:
    type  date = record day:1..31;
           month:1..12;
           year:1..9999
        end;
    reminder = record message:array[1..5] of string;
                 event:array[1..100] of date;
        end;
    var    today:date;
           memos:array[1..100] of reminder;
           calendar:array[1..365] of date;

```

Определите тип, если он определен, у следующих имен:

- 1) today.year;
- 2) memos[2];
- 3) memos[4].month;
- 4) calendar[200];
- 5) memos[16].message[2];
- 6) memos[17].message[3][2];
- 7) calendar[1].date;
- 8) memos[10].event.

10.15. Обязательно ли количество ключевых слов begin, встречающихся в некоторой программе, совпадает с количеством ключевых слов end? Сформулируйте общее правило.

10.16. Изучите модуль работы со стеком. Напишите программу, которая с его помощью переворачивает каждую строку двумерного массива.

```

{Модуль работы со стеком}
unit stackops;
interface
procedure puch(elem:integer); {поместить в стек}
function pop:integer;
function empty:boolean;
function full:boolean;
implementation
const max=100;
var stack:array[1..max]of integer;
    top:integer;
procedure push;
begin if top+1>max then exit;
      stack[top+1]:=elem;
      inc(top)
end;
function pop;
begin if top=0 then exit;

```

```

        pop:=stack[top];
        dec(top)
    end;
    function empty;
    begin empty:=top=0 end;
    function full;
    begin full:=top=max end;
    begin top:=0
    end.

```

10.17. Вычислите значения выражений:

- 1)  $[2..3,4,6] = [2..4,6]$ ;
- 2)  $[1,2] \supset [2,1]$ ;
- 3)  $['b'..'a'] = ['c'..'b']$ ;
- 4)  $[1..15,13..18] \supset [1..18]$ ;
- 5)  $[1] \supset [1,1,1]$ ;
- 6)  $[1] \leq [2]$ ;
- 7)  $[5..7] \geq [8,6,7,5]$ ;
- 8)  $[5..7] \leq [8,6,7,5]$ ;
- 9)  $[1,3] * [2,4]$ ;
- 10)  $['a','b'] + ['c']$ ;
- 11)  $[3] - [1]$ ;
- 12)  $[5,4] * [] + [4,3,2] - [3..5]$ ;
- 13)  $[] * [5,4] + [4,3,2] - [3..5]$ ;
- 14)  $[5,4] * [4,3,2] + [] - [3..5]$ ;
- 15)  $[5,4] * [4,3,2] + [3..5] - []$ ;
- 16)  $[5,4] * [3..5] + [] - [4,3,2]$ ;
- 17)  $[3..5] * [4,5] + [4,3,2] - [3,5]$ ;
- 18)  $[1] \text{ in } [[1],[2],[3]]$ .

10.18. Установите, какую работу выполняет программа. Заполните пустые комментарии нужными пояснениями:

```

type CharSet = set of char;
const Digits : CharSet = ['0'..'9'];
      Letters : CharSet = ['a'..'z','A'..'Z'];
var countDig,           {счетчик цифр}
    countLet : word;    {счетчик латинских букв}
    s : string;         {}
    i:integer;          {}
begin countLet:=0;      {}
    countDig:=countLet; {}
    s:='Введите строку символов ';
    write(s);readln(s); {}
    for i:=1 to length(s) do {}
        if s[i] in Digits           {1}
        then countDig:=countDig+1
        else if s[i] in Letters

```

```

        then countLet:=countLet+1;
writeln('Количество цифр = ', countDig : 5);
writeln('Количество букв = ', countLet : 5)
end.

```

10.19. Будет ли программа из пункта 10.18 выполнять ту же работу, если оператор, отмеченный комментарием {1}, заменить следующими двумя операторами?

```

if s[i] in Digits then inc(countDig);
if s[i] in Letters then inc(countLet).

```

Какой вариант программы, на Ваш взгляд, предпочтительнее?

10.20. Установите, что будет напечатано следующей программой:

```

var s : set of 'A'..'Z';
    ch : char;
begin s:=[];
      ch:='A';
      repeat include(s,ch);
            inc(ch);inc(ch)
      until ch>='Z';
      for ch:='Z' downto 'A' do
        if ch in s then write(ch) end.

```

10.21. Почему в результате работы программы выдается ответ: «Составить нельзя»? Можно ли подобрать такое слово, чтобы ответ программы стал утвердительным?

```

var   s:string;
      a:set of char;
      i:integer;
      f:boolean;
begin s:='хорошо';
      a:=[];
      for i:=1 to length(s) do include(a,s[i]);
      s:='шорох';
      i:=1;f:=false;
      while (i<=length(s)) and not f do
        if s[i] in a then begin exclude(a,s[i]);i:=i+1 end
        else f:=true;
        if not f
          then write('Заданное слово можно составить из букв слова
"хорошо")
          else write('Составить нельзя');
end.

```

10.22. Упростите запись программы, подсчитывающей в строке количество строчных и заглавных латинских и строчных гласных русских букв:

```

var s:string;
    klat : integer;
    kglas : integer;

```

```

    i : integer;
begin write('Введите строку ');
    readln(s);
    klat:=0;
    kglas:=0;
    for i:=1 to length(s) do
    begin if ('a'<=copy(s,i,1)) and (copy(s,i,1)<='z') or
        ('A'<=copy(s,i,1)) and (copy(s,i,1)<='Z')
        then klat:=klat+1;
        if (copy(s,i,1)='y') or
            (copy(s,i,1)='e') or
            (copy(s,i,1)='ё') or
            (copy(s,i,1)='ы') or
            (copy(s,i,1)='а') or
            (copy(s,i,1)='о') or
            (copy(s,i,1)='э') or
            (copy(s,i,1)='я') or
            (copy(s,i,1)='и') or
            (copy(s,i,1)='ю')
        then kglas:=kglas+1
    end;
    write('латинских ',klat,' гласных ',kglas)
end.

```

## 11. Процедуры и функции

11.1. Что такое функция, определяемая пользователем?

11.2. Чем отличаются функции, определяемые пользователем, от встроенных функций, например, синус, косинус и т.п.?

11.3. Что такое процедура?

11.4. Чем отличаются процедуры без параметров от процедур с параметрами?

11.5. Что такое глобальные и локальные данные? Чем они отличаются?

11.6. Что такое параметр? Какие бывают параметры?

11.7. Что такое описание процедуры или функции и чем оно отличается от оператора процедуры или указателя функции?

11.8. Как устанавливается связь между аргументами и параметрами?

11.9. Могут ли быть процедуры и функции вложенными? Приведите примеры.

11.10. Напишите процедуру без параметров, процедуру с параметрами и функцию, которые находят номер максимального элемента последовательности, заданной формулой общего члена:  $a_i = F(i, x)$ . Вычисление очередного элемента последовательности оформите в виде функции.

11.11. Напишите программу для построения на одном экране графиков функций:  $y = a * x + b$ ;  $y = a * |x| + b$ ;  $y = |a * x + b|$ .

11.12. Как передать массив в процедуру с помощью метода аргумент-параметр?

11.13. Объясните разницу в передаче массива в качестве параметра:

```
const nn=100;  
type mas=array[1..nn]of real;  
var a:mas;
```

```
procedure p1(const b:mas);  
begin {тело процедуры} end;
```

```
procedure p2(b:mas);  
begin {тело процедуры} end;
```

```
procedure p3(var b:mas);  
begin {тело процедуры} end;
```

```
procedure p4(b:array of real);  
begin {тело процедуры} end;
```

```
begin {ввод массива a}  
  p1(a); {1}  
  p2(a); {2}  
  p3(a); {3}  
  p4(a); {4}  
end.
```

Укажите, в какой точке - первой, второй или третьей - массив может изменить своё значение.

11.14. Установите, что будет напечатано в результате работы первого и второго варианта программы. Сравните результаты, сделайте выводы:

```
program FirstVariant;  
var x:real;  
procedure writeX;  
var x:real;  
begin write(x) end;  
begin x:=pi;  
  writeX;  
end.
```

```
program SecondVariant;  
var x:real;  
procedure writeX;  
  
begin write(x) end;  
begin x:=pi;  
  writeX;  
end.
```

11.15. Рассмотрите следующее описание функции:

```
function f(x,y:real):real;  
begin if x>=y  
  then f:=(x+y)/2  
  else f:=f(f(x+2,y-1),f(x+1,y-2))  
end.
```

Определите  $f(1,10)$ . Каким более простым способом можно представить и вычислить  $f(a,b)$ ?

11.16. Выполните следующие три простые программы и определите, что будет напечатано:

```
1) var a,b,c:integer;
   procedure P(x,y:integer;var z:integer);
   begin z:=x+y+z; write(x,y,z) end;
   begin a:=5; b:=8; c:=3;
         P(a,b,c);
         P(7,a+b+c,a);
         P(a*b,a div b,c)
   end.
```

```
2) var i,j,k:integer;
   procedure P(var i:integer);
   begin i:=i+1; write(i,j,k) end; {P}
```

```
   procedure Q(h:integer; var j:integer);
   var i:integer;
   procedure R;
   begin i:=i+1 end; {R}
   begin i:=j;
         if h=0
         then P(j)
         else if h=1
              then P(i)
              else R;
         write(i,j,k)
   end; {Q}
   begin i:=0;
         j:=1;
         k:=2;
         Q(1,i);
         Q(1,i);
         Q(k,j)
   end.
```

```
3) var r:real;
   procedure a(x:real);
   var y:real;
   procedure b(var x:real);
   begin x:=y+x end;
   begin y:=1; b(y); write(x+y) end;
   begin r:=2; a(r); write('r=',r) end.
```

11.17. Определите, что будет напечатано в результате выполнения программы. Всегда ли функция правильно вычисляет следующее число месяца?

```
type day = 1..31;
month = (january, february, march, april, may, june,
        july, august, september, october, november, december);
```

```

    year = integer;
    date = record day : day; month : month; year : year end;
function NextDay(CData : date) : day;
begin if CData.month in [january, march, may, july, august, december]
    then CData.day := CData.day mod 31
    else if CData.month in [april, june, september, october, november]
        then CData.day := CData.day mod 30
        else CData.day := CData.day mod 28;
    NextDay := succ(CData.day)
end;
function NextMonth(CData : date) : month;
begin if (CData.month in [january, march, may, july, august, december])
    and (CData.day=31)
    then if ord(CData.month)=11
        then NextMonth:=january
        else NextMonth:=succ(CData.month)
    else if (CData.month in [april, june, september, november])
        and (CData.day=30)
        then NextMonth:=succ(CData.month)
        else if (CData.month=february) and (CData.day=28)
            then NextMonth:=march
            else NextMonth:=CData.month
    end;
function NextYear(CData : date) : year;
begin if (CData.day=31)and(CData.month=december)
    then NextYear:=CData.year+1
    else NextYear:=CData.year
end;
function PrintMonth(mo:month):string;
begin case mo of
    january   :PrintMonth:='january';
    february  :PrintMonth:='february';
    march     :PrintMonth:='march';
    april     :PrintMonth:='april';
    may       :PrintMonth:='may';
    june      :PrintMonth:='june';
    july      :PrintMonth:='july';
    august    :PrintMonth:='august';
    september :PrintMonth:='september';
    november  :PrintMonth:='november';
    december  :PrintMonth:='december';
end
end;
var AData : date; ADay : day;
begin with AData do

```

```

begin day := 31;
      month := december;
      year := 2000
end;
Aday := NextDay(AData);
writeln(ADay);
writeln(PrintMonth(NextMonth(AData)));
writeln(NextYear(AData));
end.

```

11.18. Что будет напечатано после выполнения программы:

```

var a:integer;
function f(var x:integer):integer;
begin f:=x; x:=0 end;
begin a:=1;
      writeln(f(a)-f(a)) end.

```

11.19. Укажите правильные описания функций для вычисления факториала:

- 1) function f(n:integer):integer;  
begin f:=n\*f(f(n-1)) end;
- 2) function f(n:integer):integer;  
begin if n=0 then f:=1 else f:=f(n+1)/(n+1) end;
- 3) function f(var n:integer):integer;  
begin if n=0 then f:=1 else f:=f(n-1)\*n end;
- 4) function f(n:integer):integer;  
begin if n=0 then f:=1 else f:=n\*(n-1)\*f(n-2) end;
- 5) function f(n:integer):integer;  
begin if n<=1 then f:=1 else f:=n\*(n-1)\*f(n-2) end;
- 6) function f(n:integer):integer;  
begin if n=0 then f:=1 else f:=n\*f(n-1) end.

11.20. Установите, вычисляют ли приведенные ниже процедуры и функция одно и то же значение? Объясните своё решение. Посмотрите, как изменяются описания переменных  $p$  и  $q$ .

```

1) var n,m:integer; {исходные целые числа}
    c:integer; {число сочетаний из n по m}
    fn,fm,fnm:integer; {переменные для хранения n!, m! и (n-m)!}
    p:integer; {хранит факториал}
    q:integer; {исходное число для вычисления факториала}
procedure fact1;
var i:integer;
begin p:=1; for i:=1 to q do p:=p*i end;
begin write('Введите n и m ');
      readln(n,m);
      q:=n; fact1; fn:=p;
      q:=m; fact1; fm:=q;
      q:=n-m; fact1; fnm:=p;
      write('число сочетаний = ',fn div (fm*fnm))

```



```

end.
2) var n,m:integer; {исходные целые числа}
    c:integer; {число сочетаний из n по m}
    fn,fm,fnm:integer; {переменные для хранения n!, m! и (n-m)!}
procedure fact2(q:integer;var p:integer);
var i:integer;
begin p:=1; for i:=1 to q do p:=p*i end;
begin write('Введите n и m ');
    readln(n,m);
    fact2(n,fn);
    fact2(m,fm);
    fact2(n-m,fnm);
    write('число сочетаний = ',fn div (fm*fnm))
end.

```

```

3) var n,m:integer; {исходные целые числа}
    c:integer; {число сочетаний из n по m}
function fact3(q:integer):integer;
var i:integer; p:integer;
begin p:=1; for i:=1 to q do p:=p*i; fact3:=p end;
begin write('Введите n и m ');
    readln(n,m);
    write('число сочетаний = ',fact3(n) div (fact3(m)*fact3(n-m)))
end.

```

11.21. Исправьте процедуру так, чтобы она переводила число из десятичной системы счисления не только в восьмеричную, но в любую от 2 до 36.

```

procedure convert(x:integer);
begin if x>1 then convert(x div 8);
    write(x mod 8:1)
end.

```

11.22. Найдите все ошибки в следующем фрагменте программы:

```

type M = set of char;
function f(a,b : M; x:char):M;
begin if a*b<0
    then a:=[x]
    else if a<b
        then a:=b+x
        else if ord(x) in a-b
            then a:=a-[x..'<='];
    f:=a+b
end.

```

11.23. Определите результат выполнения программы:

```

procedure XX(n : word);
begin write(n:4);
    if n<150 then XX(2*n);
    write(n:4)
end.

```

```

end;
begin XX(1);
  writeln;
  write('Нажмите <Enter>');
  readln;
end.

```

11.24. Ниже приводится процедура перевода записи произвольного арабского числа в римскую систему счисления. Правильно ли она это делает? Исправьте возможные ошибки.

```

procedure convert(n : word);
const no=12;
      index:array[0..no] of word =(1, 4, 5, 9, 10, 40, 50, 90, 100, 400, 500,
900, 1000);
      strings:array[0..no] of string[2] =
('I', 'IV', 'V', 'IX', 'X', 'XL', 'L', 'XC', 'C', 'CD', 'D', 'CM', 'M');
var i : word;
begin if n=0 then exit;
      if keypressed then halt; {Зачем нужен этот оператор?}
      if n>=index[no]
      then begin write(strings[no]); n:=n-index[no] end
      else begin i:=no;
                  repeat
                    if keypressed then halt; {Зачем нужен этот оператор?}
                    dec(i);
                    until (n>=index[i]) and (n<index[i+1]);
                    write(strings[i]);
                    n:=n-index[i]
                  end;
                  convert(n);
end.

```

## 12. Файлы

12.1. Чем отличается файл от массива?

12.2. Верно ли, что элементы файла могут быть только одного типа?

12.3. Можно ли сравнивать файлы или присваивать один другому?

12.4. Файл t1.dat был создан программой

```

var f:file of integer;
    i:integer;
begin assign(f,'t1.dat');
      rewrite(f);
      read(i);
      while i<>0 do
      begin write(f,i);
            read(i)
      end;

```

```

        close(f);
    end.

```

Можно ли прочитать пятую запись этого файла, минуя четыре предыдущие? Можно ли изменить пятую запись этого файла? А вставить за пятой записью новую, не переписывая файла? Можно ли изменить описание переменной *i* на *real*? А на *string*?

12.5. Файл t2.dat был создан программой

```

var    f:text;
        i:integer;
begin assign(f,'t2.dat');
        rewrite(f);
        read(i);
        while i<>0 do
        begin writeln(f,i);
                read(i)
        end;
        close(f);
    end.

```

Можно ли прочитать пятую запись этого файла, минуя четыре предыдущие? Можно ли изменить пятую запись этого файла? А вставить за пятой записью новую, не переписывая файла?

12.6. Файл t3.dat был создан программой

```

var    f:file of char;
        i:char;
begin assign(f,'t3.dat');
        rewrite(f);
        readln(i);
        while i<>0 do
        begin write(f,i);
                readln(i)
        end;
        close(f);
    end.

```

Можно ли прочитать пятую запись этого файла, минуя четыре предыдущие? Можно ли изменить пятую запись этого файла? А вставить за пятой записью новую, не переписывая файла? Можно ли изменить описание переменной *i* на *byte*?

12.7. Студент, написавший следующую программу, утверждает, что она сравнивает два файла на равенство. Прав ли он?

```

var    f1,f2:file of real;
        a,b:real;
        f:boolean;
begin assign(f1,'file1.dat');reset(f1);
        assign(f2,'file2.dat');reset(f2);
        f:=true;

```

```

while not eof(f1) and not eof(f2) and f do
begin read(f1,a);
      read(f2,b);
      f:=a=b
end;
if f then write('файлы равны') else write('файлы различаются')
end.

```

12.8. Студент, написавший следующую программу, утверждает, что она удаляет из файла записи с номерами с  $n$  по  $m$  включительно. Прав ли он?

```

var   f1:file of string;           {файл}
      s:string;                   {для хранения записи файла}
      n,m,                        {интервал удаляемых записей}
      i,j:integer;               {номера обрабатываемых записей}
begin assign(f1,'f1.dat');
      reset(f1);
      write('задайте n,m? ');
      readln(n,m);
      if n>filesize(f1)
      then write('нет таких записей в файле')
      else begin   if m>filesize(f1) then m:=filesize(f1);
                    i:=n-1;j:=m;
                    while j<=filesize(f1)-1 do
                    begin seek(f1,j);
                          read(f1,s);
                          seek(f1,i);
                          write(f1,s);
                          i:=i+1;
                          j:=j+1;
                    end;
                    seek(f1,i);
                    truncate(f1);
                    close(f1);
                end;
      end.

```

Можно ли в этой программе избавиться от переменной  $j$  или  $i$ ? А от обеих сразу?

12.9. Установите, возможны ли приведенные ниже объявления файлов:

- 1) type WordFile=file of word;  
 var MyFile : WordFile;
- 2) var MyFile : file of Word.

12.10. Установите, какая задача решается программой

```

var   FileName : string; {имя файла}
      Fvar : file of char; {переменная файлового типа}
      Letter : char; {читаемый из файла символ}
begin write('Введите имя файла ');

```

```

readln(FileName);
assign(Fvar,FileName);
{$I-} {отключен автоматический контроль ввода/вывода}
reset(FVar);
{$I+} {включен автоматический контроль ввода/вывода}
if ioresult<>0
then begin writeln('Не открыт файл ',FileName); Halt end;
while not eof(Fvar) do
begin read(Fvar,Letter);
      Letter:=upcase(Letter);
      seek(Fvar,filepos(Fvar)-1);
      write(Fvar,Letter)
end;
close(Fvar)
end.

```

12.11. Установите, какую работу выполняет программа

```

{Форматирование файла}
const d=40;
var   book,copy:text;
      c:char;
      k:integer;
begin assign(book,'book.txt');
      assign(copy,'temp.txt');
      reset(book);
      rewrite(copy);
      while not eof(book) do
        if eoln(book)
        then readln(book)
        else begin read(book,c);write(copy,c) end;
      reset(copy);
      rewrite(book);
      k:=0;
      while not eof(copy) do
        begin read(copy,c);
              write(book,c);
              k:=k+1;
              if (c='.') or (k=d)
              then begin writeln(book);k:=0 end
        end;
      close(book);close(copy);
end.

```

12.12. Определите результат работы программы

```

type str80 = string[80];
var   ff : file of str80;
      gg : text;

```

```

ss : str80;
s : string;
k : word;
begin assign(ff,'strings.str');
rewrite(ff);
assign(gg,'strings.txt');
rewrite(gg);
ss:='@#&';
for k:=1 to 99 do
begin str(k :2,s);
if k<10
then ss:=copy(ss,1,3)+'0'+s[2];
else ss:=copy(ss,1,3)+s;
write(ff,ss);
writeln(gg,ss)
end;
close(ff);close(gg);
end.

```

Напишите программы для просмотра полученных файлов.

12.13. Какую работу выполняет функция? Напишите программу для создания файла, который может быть обработан представленной функцией, заполнив его подходящими случайными числами.

```

function max(var t:text):real;
var m,x:real;
begin reset(t);
read(t,m);
while not eof(t) do
begin read(t,x);
if x>m then m:=x
end;
max:=m
end.

```

12.14. Установите, есть ли ошибки в программе

```

var f:file of integer;
i:integer;
procedure pp(var fail:file of integer; var f:boolean);
var i:integer;
begin reset(fail);
i:=0;
while not eof(fail) do
begin read(fail,i);
inc(i)
end;
k:=filesize(fail);
if i=k then pp:=true else pp:=false

```

```

end;
begin write('Введите имя файла ');
      readln(i);
      assign(f,i);
      pp(fail,f);
      close(fail);close(f);
end.

```

12.15. Исправьте ошибку в программе

```

var f1,f2 : file of real;
    r : real;
    s1,s2 : string;
begin write('Введите имя исходного файла ');
      readln(s1);
      write('Введите имя создаваемого файла ');
      readln(s2);
      assign(f1,s1);reset(f1);
      assign(f2,s2);rewrite(f2);
      while not eof(f1) do
      begin read(f1,r);
              write(f2,r);
      end;
end.

```

### 13. Разные задачи

13.1. Какие операторы алгоритмического языка изменяют содержимое оперативной памяти?

13.2. Какие операторы языка Паскаль используются для записи линейных алгоритмов?

13.3. Какие операторы языка Паскаль используются для записи ветвящихся алгоритмов?

13.4. Какие операторы языка Паскаль используются для записи циклических алгоритмов?

13.5. Возможна ли на Паскале запись? Почему?

```

var  a,b,c:integer;
      b,d,e:real;
      s,c:string;

```

13.6. Возможна ли на Паскале запись? Почему?

```

const dwa=2;
      two=2;
      zwei=2;

```

13.7. Запишите самый короткий оператор Паскаля.

13.8. Напишите самую короткую программу на Паскале.

13.9. Какой результат вычисляет программа? Исправьте ее для счета в прямом направлении (от 1 к n).

```

program Examl;
var   n:integer;
      R:real;
begin readln(n);
      R:=0;
      while n>0 do
      begin R:=R+1/n;
            n:=n-1
      end;
      writeln(R)
end.

```

13.10. Перепишите приведенную выше программу с использованием операторов repeat...until и for.

13.11. Опишите:

- 1) целую переменную  $a$  и логическую переменную  $b$ ;
- 2) две строковых и одну вещественную переменную, образовав их имена только из символов  $x$  и  $y$ ;
- 3) одну литерную переменную, три целых переменных и две логических переменных, образовав их имена только из символов  $s, p, q$ .

13.12. Перечислите все операции, которые не разрешены над целыми числами.

13.13. Перечислите, для каких целей используется знак плюс в программах на Паскале.

13.14. Перечислите, для каких целей используется точка в программах на Паскале.

13.15. Укажите все ошибки в программе:

```

program test3
{ здесь возможны ошибки };
type else = array[1..n, boolean] of integer;
var   a:else;
      n,i:integer
      s:real;
begin write(Введите элементы массива');
      for i=1 to n do
      begin write('Задайте,' i, ' элемент');
            read(a[i])
      end
      s:=random;
      for n=n downto i do
            if n div 2=1 then s:=s+a[i]
            else; write('ответ = s')
      end.

```

Все ли найденные Вами ошибки обнаружит транслятор?

13.16. Укажите все ошибки в программе:

```

programm task4;

```



```

begin procedure max;
  if a>b
  then if b>c
       then m:=b
    else if b>c
       then m:=b
       else m:=c
  end;
var a{исходные данные}, b, c:real;
    m:integer; {результат;}
begin write('Введите переменные ',a,b,c);
      read(a,b,c,a);
      write(max 'максимальное из трех')
end.

```

Нарисуйте блок-схему для процедуры.

13.17. Укажите все ошибки в программе:

```

program task5;
const if:=35;
      fi:=if-15;
type arr = array[fi...if] of integer;
var   a: {массив} fi;
      i,n,j: int{индексы и количество элементов}eger;
begin write('Введите ',n);
      read(n);
      for i:=1 to n-1 do
      begin read(a[i])
            for j:=i+1 downto n do
              if a(i)>a(j) and i>j
              then a(i)=a(j);
            i:=if;
            while i<=n do write(a[i],' ');
          end.

```

Все ли найденные Вами ошибки обнаружит транслятор?

13.18. Укажите все ошибки в программе:

```

program test6;
const tri = dwa+odin;
      dwa = odin+odin;
      odin = 1;
var   n:real;
      s,i:{integer};
begin read(n);
      write('введите n ');
      if 1<n and n<=10
      then begin s:=0;;
            for dwa*n to tri do

```

```

begin s:=s+1
end;
else s:=s mod odin+dwa+tri
write('результат ',s)
end.

```

Все ли найденные Вами ошибки обнаружит транслятор?

13.19. Восстановите пропущенные операторы и элементы выражений программы (они обозначены знаком ?) так, чтобы она решала задачу: «Задан двумерный массив, состоящий из трех строк и произвольного количества столбцов. Подсчитайте количество столбцов, элементы которых могут быть длинами сторон треугольника». Установите назначение используемых переменных. Определите, сколько строк будет напечатано по окончании работы программы.

```

?
type mas = array[1..?,1..?] of real; { ? }
var   w:mas;      { ? }
      n:integer;   { ? }
      i:integer;   { ? }
      s: ? ; { ? }
begin write('Введите ? ');
      readln(n);
      for i:=1 to n do
      begin w[1,i]:=3.5+random(15);
             ?
             w[3,i]:=1.5+random(35)
      end;
      ?
      for i:=1 to n do
      if w[1,i]<w[2,i]+w[3,i]
      then ?
      s:=s+1;writeln(w[1,i],',',w[2,i],',',w[3,i])
      write('Результат = ',s);
end.

```

13.20. Восстановите пропущенные операторы и элементы выражений программы (они обозначены знаком ?) так, чтобы она решала задачу: «Проверьте равны ли два двумерных массива». Опишите назначения всех переменных. Какой результат выдаст эта программа, если будут введены три следующие строки:

```

3
1 2 3 4 5 6 7 8 9 0
1 2 3 4 5 6 7 8 9 0 ?
?
type mas=array[?..?,?..?]of real; { ? }
var   w1,w2:mas; { ? } n:integer; { ? } i,j:integer; { ? } f: ?; { ? }
begin write('Введите ? ');

```

```

readln(n);
write('Введите ',n:1,' x ',n:1,' = ', ? , ' элементов первого массива ');
for i:=1 to n do
for j:=1 to n do read(w1[?,?]);
write('Введите ',n:1,' x ',n:1,' = ', ? , ' элементов второго массива ');
for i:=1 to n do
for j:=1 to n do read(w2[?,?]);
i:=1; f:=?;
while (i<=n) and not f do
begin j:=1;
while (j<=n) and not f do
if w1[i,j]=w2[i,j] then ? else ?;
if not f then i:=i+1
end;
if not f then write('массивы ?') else write('массивы ?');
end.

```

13.21. Восстановите пропущенные операторы и элементы выражений программы (они обозначены знаком ?) так, чтобы она решала задачу: «Нечетные строки матрицы замените вектором». Опишите назначения всех переменных. Сколько строк будет выведено на экран по окончании работы программы?

```

?
type mas1=array[?..? ] of integer;    { ? }
mas2=array[?..? ] of ?;                { ? }
var x:mas1;    { ? }
w:mas2;    { ? }
n:integer;    { ? }
i:integer;    { ? }
j:integer;    { ? }
begin write('Введите ? ');
readln(n);
writeln('Введите ? ');
for i:=1 to n do
begin write('Введите ',?,'-й элемент вектора ');
?
end;
{ Первоначальное заполнение матрицы единицами }
?
?
{ Решение задачи }
for i:=0 to n div 2 do w[?]:=x;
{ Вывод результатов }
for i:=1 to n do
begin ?;
for j:=1 to n do write(w[i,j], ' ')

```

end;

end.

13.22. Восстановите пропущенные операторы и элементы выражений программы (они обозначены знаком ?) так, чтобы она решала задачу: «Переставьте указанные столбцы двумерного массива, заполненного случайными символами. Распечатайте исходный массив и результат перестановки». Опишите назначения всех переменных. Укажите максимальную размерность массива, чтобы вся выводимая информация поместилась на одном экране.

?

```
type mas=array[1..?,1..?]of char; { ? }
var w:mas; { ? } n:integer; { ? } m:integer; { ? } i:integer; { ? } j:integer; { ? }
}
p,q:integer; { ? } ch:char; { ? }
begin write('Введите ? ');
  readln(n,m);
  { формирование массива }
  for i:=1 to n do
    for j:=1 to m do w[i,j]:=?;
    { печать сформированного массива по строкам }
    for i:=1 to n do
      begin writeln;
        for j:=1 to m do ?
      end;
      write('Введите ? ');
      readln(p,q);
      { преобразование массива }
      for i:=1 to n do
        begin ?; ?; ? end;
        { печать преобразованного массива по строкам }
        for i:=1 to n do
          begin writeln;
            for j:=1 to m do ?
          end;
        end;
      end.
end.
```

13.23. Укажите все ошибки в программе:

```
var a : array[1...n] of integer;
    i,j,k,l,n,m:integer;
begin n:=010;
  while i<=n do
    begin a[i]:=sin(i)+sqrt(2*cos(1-abs(i))/ln(i);
      i:=i+1
    end;
  for a[1]:=2 to n do s:=s+a[a[1]]
  for n:=1 to n do write[a(n)]
```

end.

13.24. Укажите все ошибки в программе:

```
type var=array[1..1,2..2] of boolean;
var   i:var;
      j,k,l,n,m:integer;
begin write('Введите целое ',m);
      readln(m,k);
      k:=0;
      for i[1,2]:=false to true do k:=k*m);
      if k=true
      then write(i[1,2]);
           write(k,m)
      else write(Результат не может быть получен)
end.
```

13.25. Задан фрагмент программы:

```
s:=0; i:=0;
while n>0 do
begin if n mod 10 mod 2 = 0
      then s:=s+n mod 10;
           n:=n div 10;
           i:=i+1
end;
write('S=',s,' I=',i);
```

Определите, какими могли быть значения  $n$ , если в результате работы программы было напечатано  $S=2$   $I=4$ . Дополните фрагмент до полной работающей программы недостающими операторами.

13.26. Задан фрагмент программы:

```
s:=0;
for i:=1 to n do
begin s:=s*10;
      if a[i] mod i = 0
      then s:=s+1
end;
write('S=',s,' N=',n);
```

Определите, какими могли быть значения массива  $a$ , если в результате работы программы было напечатано  $S=1010$   $N=4$ . Дополните фрагмент до полной работающей программы недостающими операторами.

13.27. Установите, какая задача решается в предложенном фрагменте программы:

```
m:=a[1]; i:=1;
while i<=n do
begin if m<=a[i]
      then m:=a[i];
           i:=i+2
end; write(m).
```

13.28. Установите, какая задача решается в предложенном фрагменте программы:

```
m:=a[n];  
i:=n;  
while i>0 do  
begin if m<=a[i]  
      then m:=a[i];  
      i:=i-2  
end;  
write(m).
```

13.29. Установите, какая задача решается в предложенном фрагменте программы:

```
s:=0;  
i:=2;  
while i<=n do  
begin if a[i]<m  
      then s:=s+a[i];  
      i:=i+2  
end;  
write(s).
```

13.30. Установите, какая задача решается в предложенном фрагменте программы:

```
s:=1;  
i:=n;  
while i>0 do  
begin if a[i]>=m  
      then s:=s*a[i];  
      i:=i-2  
end;  
write(s).
```

13.31. Является ли символ else знаком операции?

13.32. Является ли символ : знаком операции отношения?

13.33. Пусть в программе имеются описания переменных:

var c : char; i : integer; x, y : real; b : boolean;

1) Определите тип следующих выражений:

- a) ord(c);
- b)  $x*x+\text{sqr}(y)<4$ ;
- c)  $1.0+2$ ;
- d)  $2/4$ ;
- e) '1';
- f) succ( $x>1$ );
- g) ord( $x=y$ );
- h) ord(c) and (ord(succ(c))).

2) Определите, какие выражения правильны, а какие содержат ошибки:

- a) ord(succ(c));
- b) pred(ord(c));
- c) succ(inc(c));
- d) inc(c);
- e) inc(ord(c));
- f) dec(c);
- g) pred(low(c));
- h) succ(low(c)+48);
- i)  $1 < x < 2$ ;
- j)  $x = y = b$ ;
- k)  $x = y = i$ ;
- l)  $x = (y = 1.0)$ ;
- m)  $(x < 0.0) \text{ or } (x > 1.0)$ ;
- n)  $(x > 0.0) = (x < 1.0)$ .

3) Вычислите значения выражений:

- a) ord(succ(i)) при  $i=5$ ;
- b) succ(ord(i)) при  $i=5$ ;
- c) chr(pred(ord(c))) при  $c='1'$ ;
- d) succ('1');
- e) ord('1')+ord(2);
- f) ord('1')+ord('2')=ord('3').

13.34. Из приведенного набора выберите операторы для решения задачи:

Вычислите  $P = 1 * (2+3) * (4+5+6) * \dots * (56+57+\dots+66)$ .

- 1) for i:=1 to 11 do,
- 2) k:=k+1,
- 3) p:=1,
- 4) for j:=1 to i do,
- 5) s:=0,
- 6) s:=s+k,
- 7) k:=0,
- 8) p:=p\*s,
- 9) for i:=1 to n do,
- 10) readln(n),
- 11) write(p),
- 12) s:=s+j,
- 13) p:=p\*(i+j).

13.35. Установите, что будет напечатано программой

```
const n=3; m=5;
var a : array[1..n,1..m] of integer;
    i,j,k,s1,s2 : integer;
begin k:=0;
    for i:=n downto 1 do
    begin for j:=m downto 1 do
        begin k:=k+1; a[i,j]:=k; write(a[i,j], ' '); end;
    end;
```

```

    for j:=1 to m div 2 do
    begin s1:=0; s2:=0;
        for i:=1 to n do
        begin s1:=s1+a(i,j);
            s2:=s2+a[i,m-j+1]
        end;
        write(s1,' ',s2)
    end
end.

```

13.36. Укажите, в каких частях программы на Паскале используются выражения?

13.37. Найдите формулы, которые вычисляются в программах:

- a) s:=0;  
 for i:=1 to n do  
   if not odd(i)  
   then s:=sin(i-s)  
   else s:=cos(i+s);
- b) s:=0;  
 for i:=1 to n do  
   if i mod 3=0  
   then s:=i-sin(s)  
   else s:=i+cos(s);
- c) s:=1;  
 for i:=1 to n do  
   if i mod 5=0  
   then s:=sin(abs(s)-i)  
   else s:=cos(abs(s)+i);
- d) s:=1;  
 for i:=1 to n do  
   if odd(i)  
   then s:=sin(-i/s)  
   else s:=cos(+i/s);
- e) s:=1;  
 for i:=1 to n do  
   if i mod 2=0  
   then s:=s/sin(s)  
   else s:=s/cos(s);
- f) s:=2;  
 for i:=1 to n do  
   if i mod 3<>0  
   then s:=i/sin(s)  
   else s:=cos(s)/i.

## 14. Задачи для программирования

14.1. Приведите примеры (найдите в учебниках или придумайте сами):



- 1) цикла, для которого заранее известно количество повторений;
- 2) цикла, для которого не известно количество выполнений тела, но существует только одна причина его окончания;
- 3) цикла, который может завершиться более чем по одной причине.

14.2. Решите три приведенные ниже задачи. Укажите, в чем сходство и в чем различие в их решениях:

1) напишите программу, вводящую  $n$  чисел с клавиатуры и вычисляющую среднее арифметическое введенных чисел;

2) напишите программу, которая вводит числа с клавиатуры. Признаком окончания ввода является число ноль. Программа должна подсчитать среднее арифметическое введенных чисел;

3) напишите программу, которая вводит последовательность чисел с клавиатуры. Последним числом последовательности является число 100. Программа должна подсчитать среднее арифметическое введенных чисел.

14.3. Решите задачу без использования оператора безусловного перехода. Последовательность чисел вводится с клавиатуры до тех пор, пока не встретится два нуля подряд. Напишите программу, проверяющую, является ли последовательность возрастающей, т.е. каждый её элемент больше или равен предыдущему.

14.4. Решите указанные задачи. Сравните их решения. В чём сходство и различия в их решениях?

Последовательность элементов задана формулой общего члена  $a_i = \sin(i + i/n)$ , где  $i$  изменяется от 1 до  $n$ .

1) напишите программу, находящую максимальный элемент этой последовательности;

2) напишите программу, находящую номер максимального элемента этой последовательности;

3) напишите программу, находящую минимальный элемент этой последовательности;

4) напишите программу, находящую номер минимального элемента этой последовательности;

5) напишите программу, находящую количество элементов между максимальным и минимальным элементами этой последовательности.

14.5. Напишите программу, которая находит сумму четных элементов среди  $N$  первых элементов последовательности, заданной следующими правилами:  $u_1 = u_2 = 1$ ;  $u_i = u_{i-1} + u_{i-2}$ .

14.6. Напишите программу, которая выводит на экран все трехзначные числа, в записи которых имеются цифры 2 или 4.

14.7. Напишите программу, которая выводит на принтер символ, зависящий от заданного нечетного  $N$ .

$N=3$

```

*  *
*  *
* * *
```

$N=5$

```

*  *  *
*  *  *
*  *  *
*  *  *
* * * * *
```

$N=7$

```

*  *  *  *
*  *  *  *
*  *  *  *
*  *  *  *
*  *  *  *
*  *  *  *
* * * * *
```

14.8. Напишите программу, сортирующую двумерный массив по возрастанию элементов во второй строке.

14.9. Напишите программы для вычисления:

1)  $-1 + 2 + 3 - 4 + 5 + 6 - \dots - 97 + 98 + 99$ ;

2)  $1 + 2 - 3 + 4 + 5 - 6 + \dots + 98 - 99$ .

14.10. Напишите программы для решения неравенств:  $a \cdot x > 5$  и  $a \cdot x > b$ . Сравните написанные программы.

14.11. Почему при вычислении  $8!$  компьютер выдает отрицательный результат, не смотря на то, что находится произведение положительных чисел?

14.12. Напишите программу для определения количества секунд в неделе. Почему на калькуляторе эту задачу можно решить быстрее?

14.13. Какой результат вычисляет программа? Перепишите её для счёта в направлении от 1 к  $n$ .

```

var n : integer;
    r : real;
begin readln(n);
    r:=0;
    while n>0 do
    begin r:=r+1/n;
         n:=n-1
    end;
    writeln(r)
end.
```

14.14. Перепишите программу задания 14.13 с использованием операторов repeat ... until и for.

14.15. В двумерном массиве вещественных чисел размера  $n \times m$  найдите элементы, которые являются наименьшими в своей строке и одновременно наибольшими в своём столбце.

14.16. Напишите программу, которая заполняет двумерный массив размера  $n \times m$  вещественными числами в соответствии с условиями задачи 14.15. Сколько элементов, удовлетворяющих условию задачи 14.15, может одновременно находиться в массиве?

14.17. В одномерном массиве, содержащем четырехзначные целые числа, определите количество чисел, у которых совпадают вторая и третья цифры.

14.18. Имя файла строится в соответствии с шаблоном: сначала записываются символы «rgtu», затем задаётся номер, который может быть однозначным, двузначным или трехзначным, затем указываются символы «etf». Файл, номер которого кратен шести, является текстовым, остальные файлы

содержат программы на Паскале. Имя файла не должно содержать незначащие нули. Запишите выражение на алгоритмическом языке, генерирующее имя файла по описанному шаблону на основе заданного номера. Опишите процедуры и функции, которые будут использованы при решении задачи.

14.19. Даты вводятся в соответствии с шаблоном: одна или две цифры дня месяца, точка, три буквы названия месяца (янв, фев, мар, апр, май, июн, июл, авг, сен, окт, ноя, дек), точка, две цифры номера года двадцатого или двадцать первого века. Запишите выражение на Паскале, истинное, если введенная дата принадлежит летнему или осеннему месяцам. Опишите процедуры и функции, которые будут использованы при решении задачи.

14.20. Задаются названия дней недели: пн, вт, ср, чт, пт, сб, вс и число. Запишите выражение, возвращающее название дня недели, отстоящего от заданного на указанное количество дней. Опишите процедуры и функции, которые будут использованы при решении задачи.

14.21. Даты вводятся в соответствии с шаблоном: одна или две цифры дня месяца, три буквы названия месяца (янв, фев, мар, апр, май, июн, июл, авг, сен, окт, ноя, дек), две цифры номера года двадцатого или двадцать первого века. Например, 25дек00. Запишите выражение на Паскале, истинное, если первая из двух введенных дат была раньше второй. Опишите процедуры и функции, которые будут использованы при решении задачи.

14.22. Запишите в виде программы алгоритм двоичного поиска в одномерном упорядоченном массиве. Как и почему изменится программа, если поменять упорядоченность массива на обратную?

14.23. Задан русский текст, состоящий из слов, содержащих только буквы, и целых чисел, содержащий только цифры. Слова и числа разделяются одним или несколькими пробелами. Преобразуйте текст, выразив каждое число в рублях и копейках (число копеек всегда задаётся двумя цифрами). Например, число 317 заменяется на 3 руб. 17 коп.

14.24. Задан текст, состоящий из русских слов, разделённых одним или несколькими пробелами. Укажите слова, встречающиеся в тексте более одного раза. (Учитываются полные совпадения слов).

14.25. Задана окружность радиуса  $R$  с центром в начале координат и прямая, проходящая через точки  $A(x_1; y_1)$  и  $B(x_1; -y_1)$ . Напишите программу, проверяющую, пересекаются ли окружность и прямая или они касаются друг друга, или не пересекаются и не касаются. Приведите тесты для проверки работоспособности программы.

14.26. Треугольник задан координатами вершин:  $A(0;2)$ ,  $B(2;-2)$ ,  $C(-2;-2)$ . Напишите программу, проверяющую, принадлежит ли точка  $M(x;y)$  этому треугольнику. Приведите тесты для проверки работоспособности программы.

14.27. Прямоугольник задан координатами трёх вершин:  $A(x_1; y_1)$ ,  $B(x_2; y_1)$  и  $C(x_2; y_2)$ . Напишите программу, проверяющую, принадлежит ли точка  $M(x;y)$  этому прямоугольнику. Приведите тесты для проверки работоспособности программы.

14.28. Задан квадрат координатами левого верхнего угла  $A(x_1; y_1)$  и правого нижнего угла  $B(x_2; y_2)$ . Стороны квадрата параллельны осям координат. Имеется

окружность радиуса  $R$ . Напишите программу, проверяющую, поместится ли окружность в квадрат или квадрат в окружность, или они не поместятся друг в друга. Приведите тесты для проверки работоспособности программы.

14.29. Напишите программу для решения уравнения для случая  $n$  делений. Пример уравнения приведен для  $n=3$ :  $2001=1+8:(1+8:(1+8:(1+8*x)))$ .

14.30. Напишите программу для получения наибольшего и наименьшего четырёхзначных нечётных чисел, образованных цифрами 0, 1, 4, 5, 6, 8. Подсчитайте, сколько таких чисел можно образовать, используя заданные цифры.

14.31. Напишите программу, определяющую, сколько различных имён длиной не более пяти можно образовать в Паскале, используя только заданные символы. Исходные символы вводятся с клавиатуры и являются допустимыми для записи имен, например,  $a, b, 1, 5, \#$ .

14.32. Напишите программу, в которой по крайней мере по одному разу используется каждый из следующих операторов: а) присваивания; б) условный; в) цикл `while`; г) цикл `for`; д) ввода; е) вывода. Поясните, что делает эта программа.

14.33. Опишите семантику всех операторов Паскаля, которые используются для поиска элемента в несортированном одномерном массиве.

14.34. Напишите программу, которая выводит все трехзначные числа, которые делятся на каждую из цифр в их записи.

### **Ответы и советы (указания, решения)**

В конце каждого полноценного задачника приводятся ответы, а иногда и решения. В нашем задачнике это излишне, так как существует компьютер с системой программирования Turbo Pascal. С её помощью, что называется из первых уст, можно получить полный ответ на любой вопрос и задачу, из этой книги. Именно это автор настоятельно рекомендует делать после самостоятельного решения задач. Рекомендуется также разным образом модифицировать исходные данные вопросов и задач и сравнивать полученные решения.

Ниже все же приводятся подсказки и ответы к небольшому количеству вопросов, в основном к тем из них, в которых автор может быть неверно понят.

1.1. Информация - сведения об окружающем мире и протекающих в нем процессах, воспринимаемые человеком или специальным устройством и уменьшающие степень незнания или неопределенность выбора. Например, Гидрометеоцентр сообщает информацию о том, что завтра будет дождь, и, если Вы оказались в этот день без зонта, то остаётся пожалеть о своей неверчивости.

Вообще, при отсутствии нужной информации, человек испытывает отрицательные эмоции. Вспомните своё состояние перед контрольной, когда ещё неизвестна оценка, которую Вы получите. Наоборот, при получении информации человек испытывает положительные эмоции. Вспомните своё

состояние, когда сообщают об отличных результатах Вашей работы. Вы рады тому, что это умеете делать.

Приведите другие примеры своих состояний при наличии или отсутствии нужной информации.

1.2. Обычно определение даётся через род и видовое отличие, но для предельно широких понятий более широкого рода не существует. Поэтому приходится давать описательное или сопоставительное определения.

1.5. Алгоритм можно записать на естественном языке, или на алгоритмическом языке, или представить в виде блок-схемы.

1.26. Если в программе не используются стандартные процедуры или функции, например,  $\sin(x)$ ,  $\text{write}(a,b)$ ,  $\text{trunc}(d)$  и т.п., то обращения к библиотекам стандартных функций не потребуются.

2.56. Применим приём сведения данной задачи к решению предыдущей: чтобы найти вторую цифру её нужно сделать последней, а находить последнюю цифру мы умеем.  $123 \div 10 \bmod 10 = 2$ .

Проверим, получим ли мы вторую цифру в следующих случаях?

$$1998 \div 10 \bmod 10 = 9$$

$5 \div 10 \bmod 10 = 0$  (ответ правильный, т.к. ведущие нули при записи числа опускаются).

2.57. Так же, как в 2.56, находится вторая цифра числа. Одну задачу можно реализовать разными алгоритмами. Это решение (по сравнению с 2.56) неудачно тем, что его нельзя использовать в цикле для нахождения других цифр числа.

2.58. Два варианта решения:

1)  $123 \div 100 \bmod 10 = 1$  или  $123 \div 100 = 1$

2)  $123 \div 10 \div 10 \bmod 10 = 1$ .

2.59. `Var k:integer;`

`{ 1 } k mod 10`

`{ 2 } k div 10 mod 10`

`{ 3 } k div 100`

$$k \bmod 10 + k \div 10 \bmod 10 + k \div 100.$$

2.61. Выражение должно содержать обращение к функции, например, к `sumc`:

```
function sumc(n:integer):integer;
```

```
var s:integer;
```

```
begin s:=0;
```

```
  while n>0 do
```

```
    begin s:=s+n mod 10;
```

```
      n:=n div 10
```

```
    end;
```

```
  sumc:=s
```

```
end.
```

2.62.  $\text{trunc}(k*10) \bmod 10$ .

2.64.

- 1) (0, 1, 2, 3, 4),
- 2) (40, 41, 42, 43, 44),
- 3) ( 7),
- 4) (43, -43).

2.65.

- 1) (0, 6, 12, 18, 24),
- 2) (9, 19).

Решения:

1)  $x \bmod 5$  может дать в ответе 0, 1, 2, 3, 4.

Если  $x \bmod 5 = 0$ , то  $x \div 5 = 0$ , тогда  $x$  может быть 0, 1, 2, 3, 4, подходит 0.

Если  $x \bmod 5 = 1$ , то  $x \div 5 = 1$ , тогда  $x$  может быть 5, 6, 7, 8, 9, подходит 6.

Если  $x \bmod 5 = 2$ , то  $x \div 5 = 2$ , тогда  $x$  может быть 10, 11, 12, 13, 14, подходит 12.

Если  $x \bmod 5 = 3$ , то  $x \div 5 = 3$ , тогда  $x$  может быть 15, 16, 17, 18, 19, подходит 18.

Если  $x \bmod 5 = 4$ , то  $x \div 5 = 4$ , тогда  $x$  может быть 20, 21, 22, 23, 24, подходит 24.

2)  $20 \div x$  может иметь значения 0, 1, 2, 3, 4, 5, 6, 10.

Если 0, то  $x > 20$ , тогда остаток равен  $x$ , что не возможно.

Если 1, то  $x=20, 19, 18, 17, 16, 15, 14, 13, 12, 11$ . Подходит 19.

Если 2, то  $x=10, 9, 8, 7$ . Подходит 9.

Если 3, то  $x=6$ . Не подходит.

Если 4, то  $x=5$ . Не подходит.

Если 5, то  $x=4$ . Не подходит.

Если 6, то  $x=3$ . Не подходит.

Если 10, то  $x=2$ . Не подходит.

2.71. 6)  $\arcsin(x) = \arctan(x/\sqrt{1-\sin^2(x)})$ .

3.11. Сократить выражение на  $a$ .

3.15. Выражения 2 и 6 не имеют смысла, хотя с точки зрения компилятора записаны верно.

3.16.

1) Синтаксически неверно, если не описана функция  $\sin(x)$ .

2) Синтаксически неверно, если не описана переменная  $\sqrt{x}$ .

3) Выражение после отрывающей круглой скобки не может начинаться со знака операции.

4) Синтаксически верно, но при исполнении возникнет операция деления на ноль. Смысл выражения не верен.

5) Количество открывающих круглых скобок не соответствует количеству закрывающих.

3.19.

- 1) real;
- 2) ошибочно;
- 3) integer;
- 4) real;
- 5) boolean;

- 6) integer;
- 7) real;
- 8) ошибочно;
- 9) boolean.

3.21.  $x1 := X \text{ and } 31$ ; выделяются разряды с 1 по 5 ( $31 = 0000000000011111$ ).

$X := X \text{ div } 32$ ; происходит сдвиг  $X$  на 5 разрядов вправо.

$x2 := X \text{ and } 127$ ; выделяются младшие 7 разрядов, до сдвига это были разряды с 6 по 12.

$X := X \text{ div } 128$ ; производится сдвиг вправо ещё на 7 разрядов.

$x3 := X \text{ and } 7$ ; выделяются младшие 3 разряда, до сдвигов это были разряды с 13 по 15.

4.18.

- 1) Переменные  $x$ ,  $y$ , и  $z$  равны.
- 2) Целые переменные  $n$  и  $k$  имеют одинаковую четность.
- 3) Сумма кодов символьных переменных  $a$  и  $b$  равна 100.
- 4) Переменная  $x$  не меньше  $a$ , но меньше  $b$ .
- 5) Переменная  $x$  принадлежит интервалу  $(-2.5; 3.5)$ .
- 6) Только две переменных из трех равны.
- 7) Цифра единиц переменной  $k$  равна 5.
- 8) Переменная  $m$  кратна  $n$ .
- 9) Переменные  $x$  и  $y$  имеют разные знаки.

5.24.

- 1) 'конфета' < 'конфуз';
- 2) 'конура' < 'конфета'.

5.27.  $\text{copy}(a, 1, \text{pos}(',', a)) +$

$\text{copy}(a, \text{pos}(',', a) + \text{pos}(',', \text{copy}(a, \text{pos}(',', a) + 1, \text{length}(a) - \text{pos}(',', a))) + 1,$   
 $\text{length}(a) - \text{pos}(',', a) + \text{pos}(',', \text{copy}(a, \text{pos}(',', a) + 1, \text{length}(a) - \text{pos}(',', a)))$ ).

Часть выражения в первой строке выделяет подстроку с первого символа до первой запятой включительно. Части выражения во второй и третьей строках выделяют подстроку, начиная с символа после второй запятой до конца строки.

6.24. Обе записи допустимы в Паскале. Первая запись – оператор присваивания. Вторая – логическое выражение, не смотря на ложность.

7.9. 1) Нет; 2) Да.

7.12.

- 5)  $a := (a + b - \text{abs}(a - b)) \text{ div } 2$ ;
- 6)  $a := (a + b + \text{abs}(a - b)) \text{ div } 2$ .

7.22. 1) 1; 2) 1; 3) 2; 4) 1; 5) 1; 6) 0; 7) 1; 8) 1; 9) 1.

8.8. Нельзя изменять переменную цикла в теле цикла. Не верите? Проверьте, набрав фрагмент на Паскале. Неожиданный результат будет для Вас лучшим ответом!

8.10. Первые два решения довольно очевидны: когда цикл не исполняется ни разу, т.е.  $(k+6)/2 > (2*k+16)/2$  или  $k < -10$ , а также когда  $x$  принимает одни и те же значения, но в обратном порядке, т.е. когда  $k$  – чётное число.

Казалось бы, что других вариантов быть не может, поскольку количество слагаемых одинаково, а при нечётном  $k$  в одной сумме каждое слагаемое на 0.5 меньше, чем соответствующее слагаемое в другой сумме.

Однако, существует ещё один вариант, при котором суммы могут быть одинаковы, поскольку слагаемые берутся по модулю. Ясно, что при этом в обеих суммах должны быть слагаемые разных знаков. Тем самым надо проверить нечётные  $k > -10$ , при которых  $(k+6)/2$  и  $(2*k+16)/2$  разных знаков, т.е.  $k = -7$ . Непосредственная проверка показывает, что данное значение  $k$  тоже подходит.

Итак, окончательный ответ имеет вид:  $k < -10$  или  $k$  - чётное или  $k = -7$ .

8.11.

3) В стандартном Паскале значение переменной цикла `for` после выхода из цикла не определено.

4) В стандартном Паскале запрещается изменять переменную цикла `for` в теле цикла.

8.28. Нет. Последний оператор должен иметь вид:  $s := s1 - 2*s2$ . Почему?

9.24. Если  $a[5]=1$ , то все элементы массива не изменятся. Случай  $a[5]=0$  не допустим. В остальных случаях не изменятся элементы с шестого до последнего.

9.64. Программа ошибок не содержит.

9.67. 305.

9.68. 33043.

11.16. 3)  $4.0000000000E+00$ ;  $2.0000000000E+00$ .

11.18. Обязательно проверьте свое решение, выполнив программу на Turbo Pascal.

12.11. Программа переформатирует текстовый файл `book.txt`, разбивая на строки так, чтобы каждая строка оканчивалась точкой либо содержала ровно  $d$  литер, если среди них нет точки.

12.13. Функция находит наибольшее вещественное число, из последовательности чисел, записанных в текстовый файл. Числа разделяются пробелами.

12.15. По окончании работы файлы нужно закрывать.

## Список литературы

1. Абрамов И.Г., Трифонов Н.П., Трифонова Г.Н. Введение в язык Паскаль. - М.:Наука,1988.

2. Абрамов С.А., Зима Е.В. Начала программирования на языке Паскаль. - М.:Наука,1987.

3. Аллок Д. Язык Паскаль в иллюстрациях. - М.:Мир,1991.

4. Бородич Ю.С., Вальвачев А.Н. Паскаль для персональных компьютеров. - Минск: Вышэйшая школа,1991.

5. Бутомо И.Д. Программирование на алгоритмическом языке Паскаль. - Л.:Изд-во ЛГУ,1985.

6. Вирт Н. Систематическое программирование. Введение. - М.:Мир,1977.

7. Вирт Н. Алгоритмы + структуры данных = программы. - М.: Мир,1985.



8. Грис Д. Наука программирования. - М.:Мир,1984.
9. Грогоно П. Программирование на языке Паскаль. - М.:Мир,1982.
10. Грэхем Р. Практический курс языка Паскаль для микроЭВМ. - М.: Мир,1986.
11. Дал У., Дейкстра Э., Хоор Л. Структурное программирование. - М.: Мир,1975.
12. Дейкстра Э. Дисциплина программирования. - М.:Мир,1978.
13. Джонс Ж., Харроу П. Решение задач в системе Турбо-Паскаль. - М. : Финансы и статистика,1991.
14. Епанешников А.М. Программирование в среде Turbo Pascal 7.0. - М.:Диалог-МИФИ,1993.
15. Зуев Е.А. Язык программирования Turbo Pascal 6.0, 7.0. - М.: Радио и связь,1993.
16. Йенсен К., Вирт Н. Паскаль. - М.: Финансы и статистика,1982.
17. Керниган Б. Элементы стиля программирования. - М.: Радио и связь,1984.
18. Керниган Б., Плоджер Ф. Инструментальные средства программирования на языке Паскаль. - М.: Радио и связь,1985.
19. Лаврищева Е.М. Сборочное программирование. - Киев: Наукова думка,1991.
20. Лавров С.С. Основные понятия и конструкции языков программирования. - М.: Финансы и статистика,1982.
21. Майерс Г. Искусство тестирования программ. - М.: Финансы и статистика,1982.
22. Майерс Г. Надежность программного обеспечения. - М.: Мир,1980.
23. Новичков В.С. Паскаль. - М.: Высшая школа,1994.
24. Перминов О.Н. Программирование на языке Паскаль. - М.: Радио и связь,1988.
25. Прайс Д. Программирование на языке Паскаль. - М.:Мир,1987.
26. Рюттен Т. Турбо-Паскаль 6.0. Основы языка программирования. - Киев: ВНУ,1992.
27. Турский В. Методология программирования. - М.: Мир,1981.
28. Уилсон И.Р., Эддиман А.М. Практическое введение в Паскаль. - М.: Радио и связь,1983.
29. Ульман Д. Базы данных на Паскале. - М.:Мир,1990.
30. Фаронов В.В. Программирование на персональных ЭВМ в среде Турбо-Паскаль. - М.: Изд-во МГТУ,1990.
31. Форсайт Р. Паскаль для всех. - М.: Машиностроение,1986.
32. Хьюз Дж., Мичтом Дж. Структурный подход к программированию. - М.: Мир,1980.
33. Миронова С.И. Основы программирования. Турбо-Паскаль 7.0 для школьников и абитуриентов. – М.: Аквариум; Издательство АСТ, 1999.
34. Пильщиков В.Н. Сборник упражнений по языку Паскаль. – М.: Наука, 1989.

35. Касьянов В.Н., Сабельфельд В.К. Сборник заданий по практикуму на ЭВМ. – М.: Наука, 1986.
36. Культин Н.Б. Программирование в Turbo Pascal 7.0 и Delphi. СПб.: БХВ – Санкт-Петербург, 1999.
37. Волчёнков Н.Г. Учимся программировать: Visual Basic 5. – М.: Диалог-МИФИ, 1998.
38. Григас Г. Начала программирования. – М.: Просвещение, 1987.
39. Марченко А.И., Марченко Л.А. Turbo Pascal. – М.: Бином Универсал; Киев: Юниор, 1997.
40. Фаронов В.В. Turbo Pascal 7.0. Начальный курс. – М.: Нолидж, 1997.
41. Грызлов В.И., Грызлова Т.П. Turbo Pascal 7.0. – М.: ДМК, 1998.
42. Немнюгин С.А. Turbo Pascal. – СПб.: Питер, 2000.
43. Крячков А.В., Сухина И.В., Томшин В.К. Программирование на С и С++. Практикум. – М.: Радио и связь, 1997.
44. Уэзерелл Ч. Этюды для программистов. – М.: Мир, 1982.
45. Светозарова Г.И., Мельников А.А., Козловский А.В. Практикум по программированию на языке Бейсик. – М.: Наука, 1988.
46. Бондарев В.М., Рублинецкий В.И., Качко Е.Г. Основы программирования. – Харьков: Фолио; Ростов н/Д: Феникс, 1997.
47. Гусева А.И. Учимся информатике: задачи и методы их решения. – М.: Диалог-МИФИ, 1998.
48. Гусева А.И. Учимся программировать PASCAL 7.0. – М.: Диалог-МИФИ, 1998.
49. Костюк Ю.Л. Основы алгоритмизации. – Томск: Изд-во Том. ун-та, 1996.
50. Асанов М.О. Дискретная оптимизация. – Екатеринбург: УралНАУКА, 1998.
51. Вьюкова Н.И., Галатенко В.А., Ходулев А.Б. Систематический подход к программированию. – М.: Наука, 1988.
52. Абрамов С.А., Антипов И.Н. Программирование на упрощенном Алголе. – М.: Наука, 1978.
53. Бабушкина И.А., Бушмелева Н.А. и др. Практикум по Турбо Паскалю. – М.: АБФ, 1998.
54. Культин Н. Turbo Pascal в задачах и примерах. – СПб.: БХВ – Санкт-Петербург, 2000.
55. Трифонов Н.П., Пасхин Е.Н. Практикум работы на ЭВМ. – М.: Наука, 1982.
56. Дагене В.А., Григас Г.К., Аугутис К.Ф. 100 задач по программированию. – М.: Просвещение, 1993.
57. Шень А. Программирование: теоремы и задачи. – М.: МЦНМО, 1995.
58. Гладков В.П. Задачи по информатике на вступительном экзамене в вуз и их решения. – Пермь, Перм. гос. техн. ун-т, 1997.
59. Бадин Н.М., Волченков С.Г. Ярославские олимпиады по информатике. – Ярославль, 1995.

# ОГЛАВЛЕНИЕ

Введение.....	3
1. Общие понятия.....	5
2. Типы данных (константы, переменные, операции, функции).....	7
3. Арифметические выражения.....	11
4. Логический тип данных. Логические выражения.....	15
5. Строковый тип данных.....	19
6. Программы на Паскале. Ввод, вывод, присваивание.....	23
7. Условный оператор, составной оператор, оператор выбора. Типы: перечислимый и диапазон.....	32
8. Циклы.....	42
9. Массивы.....	56
10. Записи и другие структуры данных.....	69
11. Процедуры и функции.....	78
12. Файлы.....	84
13. Разные задачи.....	89
14. Задачи для программирования.....	99
Ответы и советы (указания, решения).....	102
Список литературы.....	106

ГЛАДКОВ ВЛАДИМИР ПАВЛОВИЧ

Знаешь ли ты Паскаль?

Учебное пособие

Лит. редактор Н.В.Бабичева  
Техн. редактор Г.Я. Шилоносова  
Корректор Л.Г. Ксенофонтова

Лицензия ЛР № 020370 от 29.01.97

---

Подписано в печать 28.05.01	Формат 60х90/16
Печать офсетная.	Набор компьютерный. Усл. печ. л. 7
Уч.-изд.л. 6,4	Тираж 120
	Заказ № 57

---

Редакционно-издательский отдел  
Пермского государственного технического университета  
Адрес: 614600. Комсомольский проспект, 29а