

**ПОДГОТОВКА К ЕГЭ
ВЫСШИЙ УРОВЕНЬ КАЧЕСТВА**



ИНФОРМАТИКА

СДАЕМ БЕЗ ПРОБЛЕМ!

- ✓ Задания частей А, В и С
- ✓ Методические комментарии к каждому разделу
- ✓ Ответы и решения

**ПОДГОТОВКА К ЕГЭ
ВЫСШИЙ УРОВЕНЬ КАЧЕСТВА**

ЕГЭ **2012**

Е.М. Островская, Н.Н. Самылкина

ИНФОРМАТИКА
СДАЕМ БЕЗ ПРОБЛЕМ!

Москва  **ЭКСМО** 2011

УДК 373.167.1:004
ББК 32.81я7
О-77

Островская Е. М.

О-77 ЕГЭ 2012. Информатика. Сдаем без проблем! /
Е. М. Островская, Н. Н. Самылкина. — М. : Эксмо, 2011. —
160 с. — (ЕГЭ. Сдаем без проблем).

ISBN 978-5-699-51291-1

Пособие адресовано абитуриентам, поступающим в высшие учебные заведения, а также учащимся старших классов средних школ, гимназий, лицеев для подготовки к ЕГЭ по информатике.

Данное издание включает:

- задания частей А, В и С;
- методические комментарии к каждому разделу;
- примеры заданий с решениями;
- ответы ко всем заданиям.

Книга окажет помощь *учителям* при организации систематической подготовки учащихся к сдаче ЕГЭ по информатике.

УДК 373.167.1:004
ББК 32.81я7

ISBN 978-5-699-51291-1

© Островская Е.М., Самылкина Н.Н., 2011
© Оформление. ООО «Издательство «Эксмо»,
2011

ВВЕДЕНИЕ

По сравнению с другими предметами предмет «Информатика и ИКТ» имеет свои особенности, которые необходимо учитывать при подготовке к единому государственному экзамену. Большую часть учебного материала по предмету составляют практические умения в области ИКТ и программирования, которые затруднительно проверить в стандартном формате ЕГЭ. В то же время эти умения находятся в тесной связке с основным теоретическим содержанием предмета.

Поэтому перед разработчиками ЕГЭ по информатике и ИКТ стоит совсем непростая задача: в существующих условиях постараться проверить наиболее значимые элементы знаний и умений, необходимые для будущей профессиональной подготовки, при этом постараться не выйти за рамки стандарта общего образования по предмету. Все это говорит о том, что к экзамену по информатике и ИКТ необходимо специально уделить некоторое время на подготовку. Прежде всего просмотреть, какие темы проверяются и какими типами заданий это делается. Подготовка к ЕГЭ, как и подготовка к экзамену в традиционной форме, требует обобщающего повторения и систематизации имеющихся знаний. Наиболее успешный способ подготовки — подготовка с использованием тематических контрольных материалов и вариантов, аналогичных используемым на экзамене.

КАК РАБОТАТЬ С ИЗДАНИЕМ

Успешность сдачи единого государственного экзамена по информатике и ИКТ определяется прежде всего подготовкой выпускника по предмету и его уверенностью в успехе.

Предлагаем ученикам проверить свои знания на материале пособия «ЕГЭ. Информатика. Сдаем без проблем!», которое содержит тренировочные задания, сгруппированные в четыре основных раздела курса информатики и ИКТ. По темам, входящим в каждый раздел, разобраны типичные для вариантов ЕГЭ задания и предлагаются тематические тесты для проверки знаний.

Тематические тесты составлены таким образом, что помимо использования выпускниками для самостоятельного тренинга они также могут использоваться учителями в классе в конце изучения темы для контроля знаний или простого повторения. Каждый тематический тест охватывает либо одну тему (например «Логика»), либо все темы раздела и рассчитан на выполнение в течение урока или двух уроков.

Раздел «Представление информации» представлен темами: «Кодирование информации», «Измерение информации», «Системы счисления», «Пропускная способность канала связи». Раздел «Основы логики» раскрывает последовательно именно эту тему. Раздел «Информационные и коммуникационные технологии» объединяет темы: «Программные средства информационных и коммуникационных технологий», «Технология обработки графической и звуковой информации», «Технология обработки информации в электронных таблицах», «Технология хранения, поиска и сортировки информации в базах дан-

ных», «Телекоммуникационные технологии» и «Моделирование и компьютерный эксперимент». Раздел «Алгоритмизация и программирование» содержит только эту тему, но она входит во все части экзаменационной работы заданиями разного уровня сложности. Поэтому этот раздел объединяет как самые простые задания на примере исполнителей, действующих в обстановке, так и самые сложные, находящиеся в части С.

После выполнения каждого теста можно сравнить свои результаты с эталоном и самостоятельно определить, какие из тем вызвали затруднения и требуют дополнительного повторения. Для повторения и тренировки по вариантам, аналогичным используемым на ЕГЭ, можно использовать издания: «ЕГЭ. Информатика. Тренировочные задания» и «ЕГЭ. Информатика. Тематические тренировочные задания» авторов Н.Н. Самылкиной, Е.М. Островской.

Желаем удачи на экзамене!

1. ИНФОРМАЦИЯ И ЕЕ КОДИРОВАНИЕ

По данному разделу проверяются знания и умения выпускников: выполнять подсчет информационного объема сообщения и осуществлять перевод в разные единицы измерения; выполнять арифметические действия в двоичной, восьмеричной и шестнадцатеричной системах счисления; подсчитывать пропускную способность канала связи; решать задачи на кодирование различных видов информации (текстовой, графической, числовой). Задания в основном на проверку умений применять полученные знания в стандартной ситуации, т.е. базового (5 заданий) и повышенного (3 задания) уровня сложности, входят в первую и вторую части экзаменационной работы. Предполагается, что экзаменуемые в процессе изучения школьного курса информатики приобрели достаточный опыт в решении подобных задач. Указанные темы изучаются в основной школе и получают свое развитие в старшей школе в разной степени подробности в зависимости от выбранного профиля обучения.

Предлагаем разобрать решения типовых задач и выполнить вариант тематического теста, чтобы проверить степень подготовленности по данному разделу.

ПРИМЕРЫ ЗАДАНИЙ С РЕШЕНИЯМИ

1. Состояние некоторого объекта контролируется двадцатью семью датчиками. Наименьшее количество двоичных разрядов, необходимое для идентификации этих датчиков, равно
 - 1) 6
 - 2) 2
 - 3) 5
 - 4) 4

Решение

Задание на понимание формулы $N=2^i$. Для того чтобы закодировать 27 сигналов двоичным кодом, необходимо 5 разрядов (бит): $2^5=32$.

Ответ: 3.

2. Цветной сканер имеет разрешение 600×1200 точек/дюйм. Объем памяти, занимаемой просканированным изображением размером 2.5×4 дюйма, составляет около 21,6 Мбайт. Определите число возможных цветовых комбинаций сканера.

- 1) 256 2) 2^{16} 3) 16 777 216 4) 65 536

Решение

Для расчета каких-либо характеристик при кодировании графической информации меняется интерпретация используемых обозначений в формуле $N=2^i$, где i — *разрядность кода цвета или битовая глубина*; N — *количество цветов (палитра)*.

Объем графической информации можно найти путем умножения битовой глубины цвета на размер раstra:

$$I = i \times n_1 \text{ (точек по диагонали)} \times n_2 \text{ (точек по вертикали)}$$

Вначале определим общий объем изображения в точках: $600 \times 1200 \times 2,5 \times 4 = 7,2 \times 10^6$. Тогда информации на точку: $21,6 \times 10^6 / (7,2 \times 10^6) = 3$ байта/точку. 3 байта = 24 бит. Число возможных комбинаций 1 и 0: $2^{24} = 16777216$.

Ответ: 3.

3. Монитор работает в режиме с разрешением 1024×768 при глубине представления цвета 16 бит и частоте кадровой (вертикальной) развертки 100 Гц. Какую минимальную пропускную способность должен поддерживать видеоадаптер, работающий с монитором?

- 1) 1600 бит/с 3) 100 байт/с
2) около 75 Мбайт/с 4) 150 Мбайт/с

Решение

Объем одного изображения:

$$1024 * 768 * 26 \text{ байта} = 1572864 \text{ байт.}$$

Скорость определяется как произведение частоты и объема изображения: $1572864 * 100 = 150 \text{ Мбайт/с.}$

Ответ: 4.

4. Латинские буквы А, В, С, D закодированы двоичными числами (000, 001, 010, 011 соответственно). Какая последовательность букв закодирована в следующей двоичной записи: 000001000011?

1) ACAD 2) ABAD 3) AABD 4) ACAB

Решение

Проще всего заполнить таблицу, разбив с конца двоичную запись на триады:

| | | | |
|-----|-----|-----|-----|
| 000 | 001 | 000 | 011 |
| A | B | A | D |

Ответ: 2.

5. Учитель информатики сообщил алгоритм получения пароля для входа в операционную систему. Из исходных символов «KLASS11A1512» удалить гласные и все числа заменить на нули.

Какая из предложенных записей является паролем?

- 1) KLSS000000
2) KL0SS0000
3) KL00SS00A00
4) KLSS0000

Ответ: 1.

6. Среди приведенных чисел выберите наибольшее.

- 1) 1111101_2 3) 1231_4
2) 245_8 4) $A3_{16}$

Решение

Чтобы сравнить числа в разных системах счисления, необходимо их представить в системах счисления с одинаковым основанием, например с основанием 2.

$$245_8 = 10100101_2$$

$$1231_4 = 1101101_2$$

$$A3_{16} = 10100011_2$$

Наибольшим из четырех чисел является первое.
Ответ: 1.

7. Электронный почтовый ящик имеет объем 10 Мбайт. Информация на его адрес по открытому на прием каналу связи передается со скоростью 2,5 Кбайт/с. Через какое время у поставщика услуг электронной почты появится повод прислать извещение о переполнении почтового ящика? Укажите время в минутах.

Решение

Переведем в килобайты общий объем ящика 10240 Кбайт.

$$10 \cdot 1024 / 2,5 = 4096 / 60 = 68 \text{ мин.}$$

8. Символьная запись представляет собой набор из двоичных чисел и знаков «*». Каждая строка составлена по правилу: вначале строки знак «*», далее число в двоичном виде и знак «*».

Например, первые три строки выглядят так:

$$(1) \quad *1*$$

$$(2) \quad *10*$$

$$(3) \quad *11*$$

Сколько нулей в 12-й строке перед знаком звездочки?

Решение

12 в двоичной системе 1100, отсюда ответ 2.

Ответ: 2.

9. Даны 9 строк, которые нумеруются от 0 до 8. Нулевая строка состоит из одного символа — 0 (ноль). Каждая из последующих строк создается такими действиями: в очередную строку дважды подряд приписывается предыдущая строка и в конце дважды приписывается номер строки.

Вот первые 4 строки, созданные по этому правилу:

(0) 0

(1) 0011

(2) 0011001122

(3) 00110011220011001122 33

Запишите через запятую символ, стоящий в последней строке на **764-м** месте (считая слева направо) и количество нулей. [10]

Решение:

Нарисуем таблицу, в первом столбце укажем номера строк от 0 до 8, во втором — количество цифр в строке, в третьем — формулу для количества нулей.

| Номер строки | Кол-во цифр | Кол-во нулей |
|--------------|-------------|---|
| 0 | 1 | 1 |
| 1 | $1*2+2=4$ | 2 |
| 2 | $4*2+2=10$ | $4=2^2$ |
| 3 | 22 | $8=2^3$, т.е. степень двойки соответствует номеру строки, так как если количество символов предыдущей строки все время удваивается, то удваивается и кол-во нулей. |
| 4 | 46 | |
| 5 | 94 | |

| Номер строки | Кол-во цифр | Кол-во нулей |
|--------------|--|--------------|
| 6 | 190 | |
| 7 | 382 | |
| 8 | 766 Строка заканчивается двумя цифрами «8», а предыдущая строка заканчивалась двумя цифрами «7». Т.е. на 764-м месте стоит цифра «7». | $2^8=256$ |

Ответ: 7, 256.

10. Строки (цепочки арабских цифр) создаются по следующему правилу.

Первая строка состоит из одного символа — цифры «1». Каждая из последующих цепочек создается следующими действиями: в очередную строку записывается предыдущая цепочка цифр, затем приписывается еще число — номер строки по порядку, и далее дважды предыдущая цепочка цифр.

Вот первые 4 строки, созданные по этому правилу:

- (1) 1
 (2) 1211
 (3) 1211 3 1211 1211
 (4) 1211 3 1211 1211 4 1211 3 1211 1211 1211 3 1211 1211

Сколько раз встречается в шестой строке цифра 1? [10]

Ответ: 324

11. Строки (цепочки символов букв греческого алфавита) создаются по следующему правилу.

Первая строка состоит из одного символа — греческой буквы «М». Каждая из последующих строк создается следующим образом: в очередную строку

сначала записывается дважды предыдущая строка, а в конец приписывается еще один символ, чей порядковый номер в греческом алфавите соответствует номеру строки минус 1.

Вот первые 5 строк, созданные по этому правилу:

- (1) М
- (2) ММ А
- (3) ММА ММА В
- (4) ММА ММАВ ММА ММАВ Г
- (5) ММА ММАВ ММА ММАВ Г ММА ММАВ ММА ММАВ Г Δ

Греческий алфавит (для справки):

Α Β Γ Δ Ε Ζ Η Θ Ι Κ Λ Μ Ν Ξ Ο Π Ρ Σ Τ Υ Φ Χ Ψ Ω

Какой символ стоит в девятой строке на 505 месте (считая слева направо)?

Ответ: Α.

12. Даны 8 строк, которые нумеруются от 0 до 7. Нулевая строка состоит из одного символа — 0 (ноль). Каждая из последующих строк создается такими действиями: в очередную строку дважды подряд приписывается предыдущая строка и в начале дважды приписывается номер строки.

Вот первые 4 строки, созданные по этому правилу:

- (0) 0
- (1) 11 00
- (2) 22 11001100
- (3) 3322 1100110022 11001100 1

Запишите через запятую четыре символа, стоящих в последней строке на 1, 2, 3 и 4 местах (считая слева направо) и количество двоек. [10]

Ответ: 7, 7, 6, 6, 64.

Часть 1

4) 4

4. Состояние охраняемого объекта контролируется пятью датчиками. Сколько потребуется лампочек на контрольном пульте для идентификации этих датчиков и получения информации об объекте?

| | |
|------|------|
| 1) 8 | 3) 3 |
| 2) 2 | 4) 4 |
5. Выберите из перечисленных целое положительное число, которое можно записать в однобайтовую ячейку памяти.

| | |
|--------|---------|
| 1) 134 | 3) 675 |
| 2) 268 | 4) 1069 |
6. Сколько байт потребуется для записи адресов 65535 ячеек оперативной памяти?

| | |
|------|-------|
| 1) 6 | 3) 32 |
| 2) 2 | 4) 64 |
7. Из предложенных чисел выберите целое отрицательное число, которое можно записать в четырехбайтовую ячейку памяти.

| | |
|------------------|--------------|
| 1) $-2^{44} - 1$ | 3) -2^{32} |
| 2) $-2^{33} - 1$ | 4) -2^{31} |
8. Окружность на плоскости задана положением центра и радиусом. Сколько ячеек памяти компьютера потребуется для представления в ней данных о такой окружности, если одно число занимает одну ячейку памяти?

| | |
|------|------|
| 1) 6 | 3) 3 |
| 2) 2 | 4) 9 |
9. Куб в пространстве задан координатами вершин. Сколько ячеек памяти компьютера потребуется для представления в ней данных о кубе, если одно число занимает одну ячейку памяти?

| | |
|-------|------|
| 1) 6 | 3) 8 |
| 2) 16 | 4) 4 |

14. Монитор работает в режиме с разрешением 640×480 при глубине представления цвета 24 бита и частоте кадровой (вертикальной) развертки 150 Гц. Минимальная пропускная способность видеоадаптера, работающего с монитором, должна быть
- 1) 3600 бит/с
 - 2) около 45 Мбайт/с
 - 3) около 132 Мбайт/с
 - 4) 150 байт/с
15. Видимая область экрана 14-дюймового монитора составляет 265×198 мм, размер зерна (расстояние между точками люминофора) — 0,30 мм. Какое максимальное разрешение может поддерживать этот монитор (выберите ответ, ближайший к стандартам разрешений экрана)?
- 1) 1280×1024
 - 2) 1024×768
 - 3) 800×600
 - 4) 640×480
16. Цветной сканер имеет разрешение 300×300 точек/дюйм. Объем памяти, занимаемой просканированным изображением размером 2.5×4 дюйма, составляет около 2.7 Мбайт. Определите число возможных цветовых комбинаций сканера.
- 1) 3
 - 2) 2^{24}
 - 3) 256
 - 4) 2^3
17. Среди приведенных чисел выберите наибольшее.
- 1) B_{16}
 - 2) 1110_2
 - 3) 31_4
 - 4) 21_8

18. Даны числа в разных системах счисления: $A_8=10$, $B_{16}=C8$. Выберите из предложенных двоичных чисел C , которое отвечает условию $A < C < B$.
- 1) 100
 - 2) 111
 - 3) 1001
 - 4) 11111000
19. Латинские буквы X, Y, Z, F закодированы двоичными числами (000, 001, 010, 011 соответственно). Какая последовательность букв закодирована в следующей двоичной записи: 011001010000?
- 1) FXZY
 - 2) FYZX
 - 3) YZXF
 - 4) FZYX
20. Учитель информатики сообщил алгоритм получения пароля для входа в операционную систему. Из исходных символов «INFO11A1936» удалить числа и все гласные заменить на нули.
- Какая из предложенных записей является паролем?
- 1) INF00
 - 2) NF000
 - 3) ONF
 - 4) ONF00

Часть 2

Во второй части теста необходимо самостоятельно получить ответ в виде числа и записать его в специально отведенное для него место. Каждое число следует писать в отдельной клетке, начиная с первой.

1. Состояние охраняемого объекта контролируется тридцатью тремя датчиками. Определите наименьшее количество сигнальных лампочек, необходимых для идентификации этих датчиков и передачи с них информации.
2. Рассчитайте максимальное разрешение, которое сможет поддерживать монитор, если видимая область экрана составляет $283,5 \times 212,4$ мм, размер зерна (расстояние между точками люминофора) — 0,26 мм. Ответ представьте в виде, ближайшем к стандартам разрешений экрана, например 800×600 .
3. Определите число возможных цветовых комбинаций сканера, если он имеет разрешение 200×400 точек/дюйм, объем памяти, занимаемой просканированным изображением размером 2.5×4 дюйма составляет около 2.4 Мбайт.
4. Найдите сумму чисел F9 и B5, записанных в шестнадцатеричной системе счисления. Ответ запишите в двоичной системе счисления.
5. Десятичное число кратно 2. Определите минимальное количество нулей, которое должно быть в конце этого числа при переводе его в двоичную систему счисления.
6. Символьная запись представляет собой набор из двоичных чисел и знаков «*». Каждая строка составлена по правилу: число в двоичном виде, соответствующее номеру строки и знак «*».

Например, первые три строки выглядят так:

(4) 1*

(5) 10*

(6) 11*

Сколько единиц в записи 24 строки перед знаком звездочки?

7. Электронный почтовый ящик имеет объем 2 Мбайт. Информация на его адрес по открытому на прием каналу связи передается со скоростью 2 Кбайт/с. Через какое время у поставщика услуг электронной почты появится повод прислать извещение о переполнении почтового ящика? Укажите время в секундах.

2. ОСНОВЫ ЛОГИКИ

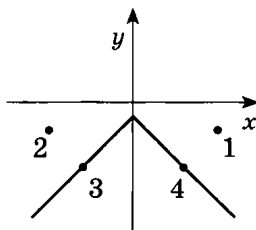
По данному разделу проверяются знания и умения выпускников: создавать и преобразовывать логические выражения в соответствии с законами логики, читать таблицу истинности, а также решать текстовые логические задачи любым удобным для этого способом (алгебраическим или составлением логического выражения). Всего в реальной экзаменационной работе содержится пять заданий: три с выбором ответа и два с кратким ответом. Два задания базового, два повышенного и одно — высокого уровня сложности, за которые можно получить по 1 баллу. Указанные темы в большинстве школ изучаются в старшей школе, хотя тема такова, что идеальным был бы вариант непрерывного ее изучения в курсе информатики с постепенным развитием тем раздела.

В части С каждого варианта КИМ ЕГЭ есть задание (С3), которое в соответствии со спецификацией проверяет умение построить дерево игры по заданному алгоритму и обосновать выигрышную стратегию. За это задание можно получить 3 балла. Такие задания лучше всего встраиваются в раздел «Основы логики», чем в разделе «Алгоритмизация и программирование».

Предлагаем разобрать решения типовых задач и выполнить вариант тематического теста, чтобы проверить степень подготовленности по данному разделу.

ПРИМЕРЫ ЗАДАНИЙ С РЕШЕНИЯМИ И ОТВЕТАМИ

1. На плоскости задан график $y = -|x|$.



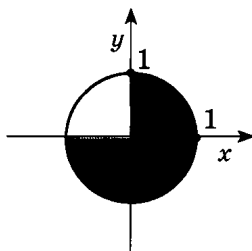
Укажите номер точки, координаты которой удовлетворяют условию:

$$(y < |x|) \wedge (y < 0) \wedge (x > 0)$$

- 1) 1 2) 2 3) 3 4) 4

Ответ: 4.

2. Укажите логическое выражение, значение которого истинно, если точка с координатами x и y попадает в заштрихованную область — часть круга с центром в начале координат и радиусом, равным 1 — ...



- 1) $((x * x + y * y) < 1) \wedge (x > 0) \wedge (y > 0)$
 2) $((x * x + y * y) < 1) \wedge \neg(y > 0) \wedge (x < 0)$
 3) $((x < 1) \wedge (x > -1)) \vee ((y < 1) \wedge (y > 0))$
 4) $(x < 1) \wedge (x > -1) \wedge (y < 1) \wedge (y > 0)$

Ответ: 2.

3. Дана таблица истинности некоторой логической функции F от трех аргументов: X , Y , Z .

| X | Y | Z | F |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |

Из перечисленных выражений выберите F.

1) $(X \vee Y) \wedge \neg Z$

3) $X \vee Y \vee Z$

2) $X \wedge Y \wedge Z$

4) $\neg X \vee \neg Y \vee \neg Z$

Решение

Составим таблицу истинности для всех выражений:

| x | y | z | $X \vee Y$ | $\neg Z$ | $(X \vee Y) \wedge \neg Z$ | $X \wedge Y \wedge Z$ | $X \vee Y \vee Z$ | $\neg X$ | $\neg Y$ | $\neg X \vee \neg Y \vee \neg Z$ |
|---|---|---|------------|----------|----------------------------|-----------------------|-------------------|----------|----------|----------------------------------|
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 |
| 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |

Значения шестого столбца и F полностью совпадают, значит это и есть правильный ответ под номером 1.

Ответ: 1.

4. Для каких чисел X и Y ложно высказывание

$$((X - Y) > 0) \wedge (X * Y \leq 0) \wedge (X + Y < 0)?$$

1) (0; 0)

2) (1; -4)

3) (1; 4)

4) (-1; -2)

Решение

Подставив предлагаемые варианты значений в выражение, получим 0, только для варианта 2.

5. Дано тождественно-истинное логическое выражение:

$$(A \wedge B) \vee (\neg B \wedge \neg C \wedge D) = 1$$

где A, B, C, D — логические переменные. Определите, сколько решений оно имеет.

В ответе не перечисляйте все различные наборы значений A , B , C и D , при которых выполняется данное равенство. В качестве ответа нужно указать только количество таких наборов.

Решение

Выражение будет тождественно-истинным в случае, если хотя бы одно из выражений $(A \wedge B)$ или $(\neg B \wedge \neg C \wedge D)$ истинно. Первое выражение истинно только при $A=1$ и $B=1$, значение D в этом случае может быть как 0, так и 1. Получаем два набора. Второе выражение истинно при $B=0$, $C=0$ и $D=1$, значение A в этом случае может быть и 0, и 1. Получили еще два набора.

Ответ: 4.

6. В зоомагазине продают больших и маленьких аквариумных рыбок. Большая рыбка стоит в два раза дороже маленькой. Саша купил 5 больших рыбок и 3 маленьких. Если бы он купил 3 больших рыбки и 2 маленькие, то потратил бы на 20 рублей меньше. Сколько стоит каждая рыбка?

Решение

Задача-разминка решается как алгебраически так и с помощью рассуждений. Будем рассуждать следующим образом.

Поскольку цена одной большой рыбки равна цене двух маленьких, то 5 больших рыбок будут стоить столько же, сколько 10 маленькие. Значит, 5 больших рыбок плюс 3 маленькие будут стоить столько же, сколько 13 маленьких. С другой стороны, цена 3 больших и 5 маленьких рыбок равняется цене 11 маленьких рыбок. Таким образом, разница между ценой 5 больших и 3 маленьких рыбок оказывается равной разнице между ценой 13 и 11 маленьких рыбок, т.е. равна цене 2 маленьких рыбок. Поскольку две маленькие рыбки стоят 20 рублей, то цена одной соответственно — 10 рублей.

Проверим. Цена маленькой рыбки 10 рублей, цена большой рыбки — 20 рублей. Сумма покупки 5 больших и 3 маленьких рыбок составит 130 рублей. Сумма покупки 3 больших и 5 маленьких рыбок составит 110 рублей, на 20 рублей меньше.

7. В парламенте некоторого государства заседают 100 парламентариев. Каждый из них либо честен, либо продажен. О парламентариях известны следующие два факта:

1. По крайней мере один из парламентариев является честным.

2. Из каждой произвольно выбранной пары парламентариев по крайней мере один продажен.

Опираясь на указанные факты, определите количество честных парламентариев.

Решение

В соответствии с первым фактом мы имеем одного честного парламентария. Опираясь на второй факт, получаем, что из оставшихся 99 парламентариев честных быть не может никто, иначе в паре с первым будут два честных парламентария, что противоречит второму факту. Как это ни грустно, получим, что в парламенте всего один честный парламентарий.

Ответ: 1.

8. Однажды в лесу расчистили дорожки и устроили соревнование по бегу на роликовых коньках. В соревнованиях участвовали: проказница Мартышка (П), Осел (О), Козел (К) и косопалый Мишка (М). По окончании болельщики затеяли спор о том, кто за кем пришел к финишу. Звери утверждали:

1) Проказница Мартышка (П) пришла второй, а Осел (О) был третьим.

2) Проказница Мартышка (П) была первой, а косопалый Мишка (М) пришел вторым.

3) Козел (К) был вторым, а Осел (О) пришел четвертым.

На самом деле каждый из зверей ошибся один раз.

В каком порядке участники соревнования пришли к финишу? В ответе укажите первые буквы имен.

Решение

Задача решается путем перебора предположений об истинности или ложности простых логических выражений. Предположим, что верным является часть утверждения первого болельщика (*проказница Мартышка (П) пришла второй*), тогда ложными будут второе утверждение второго болельщика (*косолапый Мишка (М) пришел вторым*) и первое утверждение третьего болельщика (*Козел (К) был вторым*). Из оставшихся истинных утверждений одно (*проказница Мартышка (П) была первой*) вступает в противоречие с утверждением первого болельщика. Получается, что проказница Мартышка заняла и первое, и второе места одновременно.

Рассмотрим вариант, что истинно второе утверждение первого болельщика (*Осел (О) был третьим*). Тогда ложными будут первое утверждение первого болельщика и второе утверждение третьего болельщика, а также второе утверждение второго болельщика. Но мы получим истинное первое утверждение третьего болельщика (*Козел (К) был вторым*) и истинное первое утверждение второго болельщика (*проказница Мартышка (П) была первой*). Ну а последним пришел косолапый Мишка (М).

Ответ: ПКОМ.

9. Двое играют в следующую игру. Из имеющейся кучки из 14 камней по очереди берут камни в количестве 1, 2 или 3 по своему усмотрению. Выигрывает тот, кто возьмет последний камень. Опишите выигрышную стратегию игры.

Решение

Ответ необязательно оформлять таблицей или иным способом. Можно просто описать словами, как будут ходить игроки. Эксперты проверят любое решение, на конечном результате это не скажется.

Все камни из кучи удобно разложить (хотя бы мысленно) на 4 малые кучки, в первой оставив 2 камня и в трех остальных — по 4 камня. Первый игрок берет два камня, далее, независимо от того, сколько камней взял второй игрок (1, 2 или 3), первый игрок забирает ставшиеся камни в малой кучке. Поэтому последний камень возьмет первый игрок, и для него это и есть выигрышная стратегия.

Для второго игрока стратегия выигрышная в случае, если первый игрок допустит на каком-либо ходу ошибку. Если вначале первый игрок взял один камень (или три камня), второму следует забрать второй камень из малой кучки (три камня из двух малых кучек). Таким образом, второй игрок становится в позицию начинающего и выигрывает. [4]

10. Два друга придумали новую игру. На любое поле шахматной доски по очереди ставят шашки любого цвета. Выигрывает тот, кто сделает последний ход, т.е. после него все клетки оказываются занятыми шашками.

Кто выигрывает при безошибочной игре обоих игроков — игрок, делающий первый ход, или игрок, делающий второй ход? Каким должен быть первый ход выигрывающего игрока? Ответ обоснуйте.

Решение

Выигрывает второй игрок в том случае, если будет ставить свои шашки симметрично шашкам первого игрока относительно центра доски. Можно привести рисунок шахматной доски, хотя обоснование ответа очевидно.

11. Задание из демоверсии 2010 года. Два игрока играют в следующую игру. На координатной плоскости стоит фишка. Игроки ходят по очереди. В начале игры фишка находится в точке с координатами $(-3, 2)$. Ход состоит в том, что игрок перемещает фишку из точки с координатами (x, y) в одну из трех точек: или в точку с координатами $(x+5, y)$, или в точку с координатами $(x, y+4)$, или в точку с координатами $(x+3, y+3)$. Выигрывает игрок, после хода которого расстояние по прямой от фишки до точки с координатами $(0, 0)$ больше 12 единиц. Кто выигрывает при безошибочной игре обоих игроков — игрок, делающий первый ход, или игрок, делающий второй ход? Каким должен быть первый ход выигрывающего игрока? Ответ обоснуйте. [10]

Решение

Выигрывает второй игрок.

Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке записаны координаты фишки на каждом этапе игры.

| | 1 ход | 2 ход | 3 ход | 4 ход |
|-------------------|--------------------------------|---|--|---|
| Стартовая позиция | I игрок (все варианты хода) | II игрок (выигрышный ход) | I игрок (все варианты хода) | II игрок (выигрышный ход, один из вариантов) |
| -3,2 | 0,5 | 5,5 | 8,8 | 13,8 |
| | | | 5,9 | 10,9 |
| | | | 10,5 | 15,5 |
| | -3,6 | 2,6 | 5,9 | 10,9 |
| | | | 2,10 | 7,10 |
| | | | 7,6 | 12,6 |
| | 2,2 | 5,5 или 2,6 (экзаменуемо- му достаточно привести один из вариантов) | Те же варианты третьего-четвертого ходов | |
| | | | | |
| | | | | |

ТЕМАТИЧЕСКИЙ ТЕСТ

Тест состоит из трех частей и содержит 16 заданий. Время выполнения теста — 80 минут. За каждое правильно выполненное задание части 1 и 2 можно набрать 1 балл, за выполнение задания части 3 можно набрать 3 балла. Если задание вызывает затруднение, перейдите к следующему. К невыполненным заданиям можно будет вернуться, если останется время.

Часть 1

При выполнении заданий этой части необходимо поставить «х» в клетке с номером ответа, который вы считаете правильным.

1. Формулой логического высказывания

«Мы поедem в Санкт-Петербург, и если встретимся там с друзьями, то интересно проведем время» является...

- 1) $A \wedge (B \rightarrow C)$
- 2) $(A \wedge B) \rightarrow C \vee D$
- 3) $(A \wedge B) \leftrightarrow (C \wedge D)$
- 4) $A \wedge B \rightarrow C$

**2. Какое логическое выражение равносильно данному:
 $A \wedge B \vee A \wedge \neg B$?**

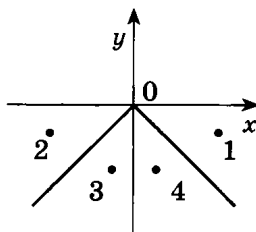
- 1) $A \wedge B$
- 2) $A \vee B$
- 3) B
- 4) A

3. Дано логическое выражение: $B \wedge \neg A \wedge (B \vee A)$

Определите его значение.

- 1) 1
- 2) 0
- 3) A
- 4) B

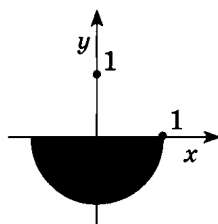
4. На плоскости задан график $y = -x$.



Укажите номер точки, координаты которой удовлетворяют условию:

$$(y > |x|) \wedge (y < 0) \wedge (x > 0)$$

- 1) 1 2) 2 3) 3 4) 4
5. Укажите логическое выражение, значение которого истинно, если точка с координатами x и y попадает в заштрихованную область — часть круга с центром в начале координат и радиусом, равным 1 — ...



- 1) $((x < 1) \wedge (x > -1)) \vee ((y < 1) \wedge (y > 0))$
 2) $((x^2 + y^2) < 1) \wedge (y < 0)$
 3) $(x < 1) \wedge (x > -1) \wedge (y < 1) \wedge (y > 0)$
 4) $((x^2 + y^2) < 1) \wedge (\neg (y > 0) \wedge (x > 0))$
6. Дана таблица истинности некоторой логической функции F от трех аргументов: X, Y, Z .

| X | Y | Z | F |
|---|---|---|---|
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |

Из перечисленных выражений выберите F.

- 1) $\neg X \wedge (\neg Y \vee \neg Z)$
 - 2) $X \wedge Y \wedge Z$
 - 3) $X \vee Y \vee Z$
 - 4) $(X \vee Y) \wedge \neg Z$
7. Выберите из перечисленных логическое выражение, значение которого будет истинно при $K=1$, $M=0$, $L=0$, $N=1$.
- 1) $\neg N \vee (M \vee (\neg N \wedge K L))$
 - 2) $(M \wedge \neg N) \wedge (N \wedge (K \vee L))$
 - 3) $L \vee M \wedge (\neg(K \vee \neg N))$
 - 4) $K \wedge \neg L \vee \neg N \wedge M$
8. Для каких чисел A и B истинно высказывание:
 $\neg ((B > 0) \vee (A - B \geq 2)) \wedge (A \leq 0)$?
- 1) $A = 1$; $B = -3$
 - 2) $A = -3$; $B = -1$
 - 3) $A = -3$; $B = 1$
 - 4) $A = 1$; $B = 3$
9. Выберите из перечисленных тождественно-ложное логическое выражение. (*Примечание.* Тождественно ложным называют выражение, ложное при любых значениях его аргументов.)
- 1) $(Y \rightarrow Z) \rightarrow (Z \vee \neg (X \wedge Y))$
 - 2) $\neg (X \wedge Y \wedge Z) \rightarrow (X \vee Y)$
 - 3) $(Y \wedge Z) \rightarrow (X \vee Y \wedge Z)$
 - 4) $(X \wedge Y) \rightarrow (X \vee Y \wedge Z)$

Часть 2

Во второй части теста необходимо самостоятельно получить ответ в виде числа или последовательности букв и записать его в специально отведенное для него место. Каждый символ следует писать в отдельной клетке, начиная с первой.

1. Определите наибольшее целое число X , при котором истинно логическое высказывание

$$(X * X - 25 > 0) \wedge (100 > (X + 15) * (X - 5)).$$

2. Сколько различных решений имеет уравнение

$$(A \wedge B \vee \neg C) = 1?$$

A, B, C — логические переменные. В ответе не перечисляйте все наборы значений, при которых выполняется равенство, а укажите количество таких наборов.

3. На очередном Международном конкурсе музыкальных исполнителей им. П.И. Чайковского зрители поспорили, участник какой страны будет победителем. Были высказаны следующие предположения:

— первым будет пианист из России, а вторым — пианист из Израиля;

— исполнитель из России будет вторым, а из Польши — третьим;

— вторым будет пианист из Германии, а польский музыкант будет четвертым.

Оказалось, в каждом из высказанных предположений одно — истинно, другое — ложно. Определите победителей, написав строку из 4 символов — первых букв названий стран, получивших места с первого по четвертое (в порядке занятых мест). (Буквы заглавные.)

Например, ответ мог бы выглядеть так: ИРГП.

4. Андрей Иванович(1), Федор Петрович(2), Валерий Сергеевич(3) и Григорий Алексеевич(4) сопровождали своих дочерей на бал. Заключительный танец каждая из девушек танцевала не со своим отцом. Образовались следующие пары: Лена с Андреем Ивановичем, Анна с отцом Кати, Тома с отцом Анны, Федор Петрович с дочерью Валерия Сергеевича, а Валерий Сергеевич с дочерью Андрея Ива-

новича. Определите: кто кому приходится дочерью? Выпишите пары: отец — дочь, проставив соответствующий номер и первую букву имени девочки. Например, ответ мог бы выглядеть так: 1К, 2А и т.д.

5. На детской площадке подруги решили прокатиться на карусели и разместились по кругу: Алла, Тома, Даша и Надя. Определите, какого цвета было платье на каждой девочке, если известно, что:

1) девочка в зеленом платье — не Алла и между Надей и девочкой в голубом платье сидит не Тома;

2) девочка в белом платье сидит между девочкой в розовом платье и Томой.

Ответ запишите парами первых букв имени и цвета. Например: АР, ТБ, НГ, ДЗ.

Часть 3

В третьей части теста также необходимо самостоятельно получить ответ, и записать полное решение и полученный ответ в бланк ответов № 2.

Ответы записывайте четко и разборчиво.

1. Два третьеклассника играют в «крестики-нолики» по своим правилам. Каждый игрок по своему усмотрению может поставить как «крестик», так и «нолик» в любой клетке поля 3×3 . Выигрывает тот, кто получит после своего хода три подряд идущих одинаковых символа (по вертикали, по диагонали или по горизонтали). Кто выигрывает при безошибочной игре обоих игроков — игрок, делающий первый ход, или игрок, делающий второй ход? Каким должен быть первый ход выигрывающего игрока? Ответ обоснуйте.

2. Два игрока играют в следующую игру. На координатной плоскости стоит фишка. Игроки ходят по очереди. В начале игры фишка находится в точке с координатами $(3, 2)$. Ход состоит в том, что игрок перемещает фишку из точки с координатами (x, y) в одну из трех точек: или в точку с координатами $(x+3, y)$, или в точку с координатами $(x, y+2)$, или в точку с координатами $(x, y+4)$. Выигрывает игрок, после хода которого расстояние от фишки до точки с координатами $(0, 0)$ больше 12 единиц. Кто выигрывает при безошибочной игре обоих игроков — игрок, делающий первый ход, или игрок, делающий второй ход? Каким должен быть первый ход выигрывающего игрока? Ответ обоснуйте. [1]

3. «ИНФОРМАЦИОННЫЕ И КОММУНИКАЦИОННЫЕ ТЕХНОЛОГИИ» И «МОДЕЛИРОВАНИЕ И КОМПЬЮТЕРНЫЙ ЭКСПЕРИМЕНТ»

Раздел «Информационные и коммуникационные технологии» состоит из пяти тем («Программные средства информационных и коммуникационных технологий», «Технология обработки графической и звуковой информации», «Технология обработки информации в электронных таблицах», «Технология хранения, поиска и сортировки информации в базах данных», «Телекоммуникационные технологии») и в экзамене представлен семью заданиями в первой и второй частях работы (три задания базового уровня и четыре задания повышенного уровня сложности).

Специфика предмета «Информатика и ИКТ» заключается в большой практической составляющей предмета (информационные и коммуникационные технологии занимают не менее 50% учебного времени), и эту часть нельзя проверить теоретическими заданиями. Более того, значительная часть практических навыков прямо определяется используемыми техническими средствами (компьютерами и программным обеспечением), которые, во-первых, различаются между собой, во-вторых, быстро изменяются.

Проверяемые теоретические элементы содержания по перечисленным темам в вариантах КИМ ЕГЭ по информатике относятся к базовому и повышенному уровню и изучались учащимися в основной школе. В старшей школе, где информатика изучалась на базовом уровне, материал должен быть повторен и систематизирован в несколько расширенном виде на задачах иного уровня. В школах, где предмет изучался на профильном уровне, предлагаемые в вариантах ЕГЭ задания будут достаточно простыми.


Кодификатор элементов содержания по информатике содержит такой раздел, как «Моделирование и компьютерный эксперимент», состоящий из четырех тем. В настоящее время в ЕГЭ проверяется только одна тема раздела — «Представление и считывание данных в разных типах информационных моделей (схемы, карты, таблицы, графики, формулы)». Вариант ЕГЭ содержит одно задание базового уровня.

Предлагаем разобрать решения типовых задач и выполнить вариант тематического теста, чтобы проверить степень подготовленности по данному разделу.

ПРИМЕРЫ ЗАДАНИЙ С РЕШЕНИЯМИ И ОТВЕТАМИ


1. В ячейку A1 электронной таблицы EXCEL введено слово «понедельник».

| | A | B | C | D |
|---|---------------|---|---|---|
| 1 | понедельник → | | | |
| 2 | | | | |
| 3 | | | | |
| 4 | | | | |

Ячейка A1 выделяется, и маркер заполнения  «протягивается» вправо. Что появится в ячейке B1?

- 1) ячейка B1 останется пустой
- 2) «вторник»
- 3) «понедельник»
- 4) «воскресенье»

Решение

В электронной таблице EXCEL есть функция автозаполнения, которая работает при протягивании  и в следующей ячейке появится слово «вторник».

Ответ: 2.

2. Задан фрагмент таблицы базы данных.

| Address ID | First Name | Last Name | Address | Home's Phone |
|------------|------------|-----------|--------------------------|--------------|
| 1 | Аркадьев | Андрей | ул. Маршала Жукова, 45,5 | 255-50-55 |
| 2 | Андреева | Анна | ул. Андреева, 25,5 | 659-55-55 |
| 3 | Кошелева | Ольга | ул. Кирова, 10,2 | 303-32-33 |
| 4 | Товарина | Тамара | ул. Транспортная, 1,14 | 727-79-77 |

Сколько записей будет найдено, если выбраны следующие условия поиска:

- поле поиска — **First Name**;
- образец поиска — сочетание символов «ва»;
- совпадение с образцом — в любой части поля.

1) 1 2) 2 3) 3 4) 0

Решение

Следует последовательно выполнить два шага:

1. Выбрать поле **First Name**;
2. В этом поле найти сочетание символов «ва», причем в любой части слов, т.е. это может быть не только окончание фамилии, но и слог в середине. Этим условиям соответствуют записи под номерами 2, 3, 4.

Ответ: 3.

3. Доступ к файлу film.doc, находящемуся на сервере www.net, осуществляется по протоколу ftp. В таблице фрагменты адреса файла закодированы буквами от А до Ж. Запишите последовательность этих букв, кодирующую адрес указанного файла. [10]

| | |
|---|------|
| А | ::// |
| Б | film |
| В | ftp |
| Г | .net |
| Д | .doc |
| Е | / |
| Ж | www |

Решение

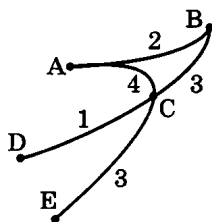
Адрес искомого файла ftp://www.net/film.doc

Ответ: ВАЖГЕБД.

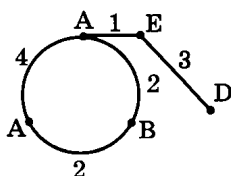
4. В таблице приведен тариф на грузовые перевозки между соседними населенными пунктами. Выберите схему, соответствующую таблице. [7]

| | А | В | С | Д | Е |
|---|---|---|---|---|---|
| А | х | 2 | 4 | | 1 |
| В | 2 | х | 2 | | |
| С | 4 | 2 | х | | |
| Д | | | | х | 3 |
| Е | 1 | | | 3 | х |

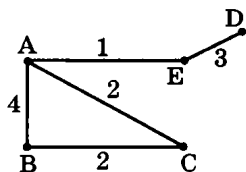
1)



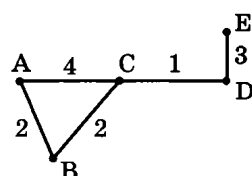
3)



2)



4)



Ответ: 3.

ЗАДАНИЯ

Тест состоит из двух частей и содержит 22 задания. Время выполнения теста — 30 минут. За каждое правильно выполненное задание можно набрать 1 балл. Если задание вызывает затруднение, перейдите к следующему. К невыполненным заданиям можно будет вернуться, если останется время.

Часть 1

При выполнении заданий этой части необходимо поставить «х» в клетке с номером ответа, который вы считаете правильным.

1. Дан фрагмент электронной таблицы:

| | А | В | С | Д |
|---|---|-----------------|---|---|
| 1 | | | | |
| 2 | | | | |
| 3 | | Список учащихся | | |
| 4 | | | | |

Укажите адрес ячейки, в которой содержится текст: «Список учащихся».

- 1) С3
 - 2) В2
 - 3) D3
 - 4) В3
2. Сколько ячеек в блоке В5 : D8 электронной таблицы?
- 1) 13
 - 2) 12
 - 3) 9
 - 4) 6


3. В электронной таблице в блок A1:A3 введены последовательно числа — 2, 3, 4, а в блок B1:B3 — 10, 12, 14. Значение ячейки C1 вычисляется по формуле: $=A1+B1-A2$.

| | A | B | C | D |
|---|---|----|---|---|
| 1 | 2 | 10 | 9 | |
| 2 | 3 | 12 | ↓ | |
| 3 | 4 | 14 | | |
| 4 | | | | |

Какое число появится в ячейке C3, если выделить ячейку C1 и маркер заполнения  «протянуть» вниз?

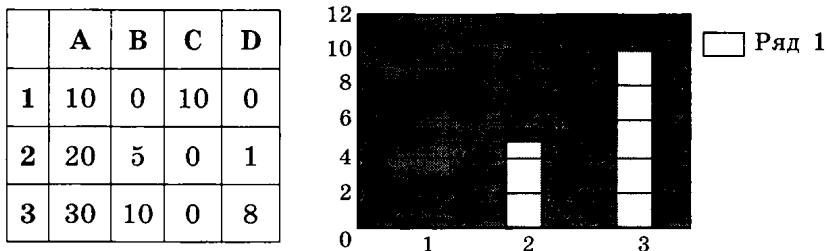
- 1) 11
 - 2) 18
 - 3) 19
 - 4) 21
4. В электронную таблицу введены данные, причем в ячейках A1:B2 данные являются первичными, а в ячейках C1:C2 — производными.

| | A | B | C | D |
|---|----|----|---|---|
| 1 | 2 | 13 | 4 | |
| 2 | 7 | 16 | | |
| 3 | 12 | 19 | | |
| 4 | | | | |

Какое число появится в ячейке C2, если выделить ячейку C1 и маркер заполнения  «протянуть» вниз?

- 1) 6
- 2) 7
- 3) -3
- 4) -10

5. По данным электронной таблицы построена диаграмма.



Укажите имя столбца таблицы, данные которого отражены на диаграмме.

- 1) A
- 2) B
- 3) C
- 4) D

6. Задана таблица базы данных.

| Address ID | First Name | Last Name | Address | Home's Phone |
|------------|------------|-----------|-------------------------|--------------|
| 1 | Васильев | Андрей | ул. Жданова, 15, 5 | 555-11-55 |
| 2 | Петров | Иван | ул. Марьина Роща, 31, 3 | 333-22-33 |
| 3 | Пырьев | Георгий | б-р Жулебинский 7,79 | 777-55-77 |
| 4 | Уманишвили | Арчил | ул. Космонавтов 17,1 | 195-23-45 |

Сколько записей базы данных будет найдено, если выбраны следующие условия поиска:

- поле поиска — «**First Name**»;
- образец поиска — сочетание символов «ев»;
- совпадение с образцом — в любой части поля.

- 1) 1
- 2) 2
- 3) 3
- 4) 0

7. Дан фрагмент таблицы базы данных.

| Ad- dress ID | First Name | Last Name | Address | Home's Phone |
|--------------------|-----------------|----------------|------------------------------|-----------------|
| 1 | Панков | Валентин | пр. Мира, 8, 20 | 184-20-83 |
| 2 | Ковалев | Евгений | ул. Зорге, 6, 48 | 521-12-40 |
| 3 | План- кова | Евгения | ул. Киев- ская, 5, 25 | 243-12-17 |
| 4 | Панков- ский | Виктор | Кутузовский проспект 2, 8 | 243-20-27 |
| 5 | Мурова | Вален- тина | Кутузовский проезд, 4, 6 | 243-67-07 |

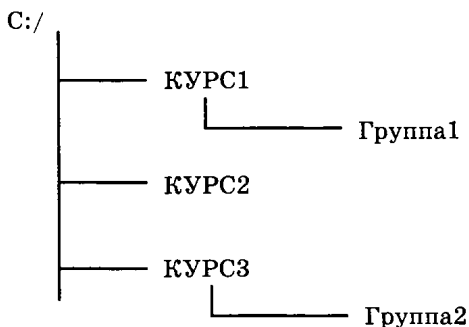
По данным таблицы требуется восстановить номер телефона абонента, о котором известно, что его фамилия либо Панков, либо Панковский, что проживает он на Кутузовском проспекте и что номер его телефона оканчивается на цифру «7». Чтобы найти номер телефона этого абонента, необходимо применить следующий фильтр:

- 1) (First Name= "П*") И (Address= "К*") И (Home's Phone=#7)
- 2) (First Name= "Панко*") И (Address= "Кутузовский*") И (Home's Phone= ### - ## -#7)
- 3) (First Name= "Панко*") И (Address= "Кутузовский проспект*") И (Home's Phone=### - ## - #7)
- 4) (First Name= "П*") И (Address= "Кутузовский *") И (Home's Phone=#7)

8. Файл turbo.exe записан на диске С, в каталоге TP7, подкаталоге BIN. Выберите правильный путь доступа к файлу.

- | | |
|-------------------------|--------------------------|
| 1) turbo.exe | 3) C:\BIN\ turbo.exe |
| 2) C:\TP7\BIN\turbo.exe | 4) C:\TP7\ BIN\turbo.exe |

9. Дано дерево каталогов:



Определите полное имя файла Группа2.

- 1) C:/КУРС1/Группа2
 - 2) C:/КУРС2/КУРС3/Группа2
 - 3) C:/КУРС3/Группа1/Группа2
 - 4) C:/КУРС3/Группа2
10. Файл, полное имя которого было K:\DOC\KLASS\spisok1, сохранили в подкаталоге ARHIV корневого каталога диска L. Определите полное имя сохраненного файла.
- 1) L:\DOC\ARHIV\spisok1
 - 2) L:\KLASS\ARHIV\spisok1
 - 3) L:\ARHIV\spisok1
 - 4) L:\spisok1
11. Для выполнения операций с несколькими файлами одновременно используются **маски имен файлов**. Маска может содержать последовательность букв, цифр и прочих допустимых в именах файлов символов. Чаще всего используют следующие символы:

Символ «?» (вопросительный знак) означает один произвольный символ.

Символ «*» (звездочка) означает любую последовательность символов произвольной длины, в том числе «*» может обозначать и пустую последовательность.

Выберите имя файла, удовлетворяющее маске: **v?d*.doc**.

- 1) vedomosti.doc
- 2) vesti.doc
- 3) vird.doc
- 4) vse_documenti.doc

12. Выберите из перечисленных IP-адрес конкретного узла

- 1) 61.259.16.41
- 2) 226.17.81.22
- 3) 202.96.455.14
- 4) 221.12.22.31

13. Компьютер находится в сети 192.168.22.0. Укажите IP-адрес, к которому он без посредников не может обратиться

- 1) 192.168.22.10
- 2) 192.168.16.2
- 3) 192.168.22.254
- 4) 192.168.22.1

14. Укажите протокол, не позволяющий передать графическое изображение

- 1) HTTP
- 2) POP3
- 3) DNS
- 4) SMTP

15. В турнирной таблице приведены итоги первенства по теннису:

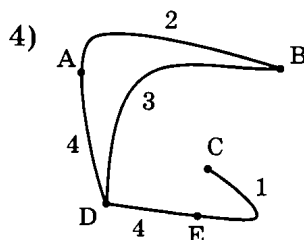
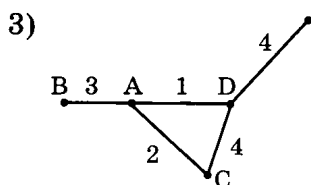
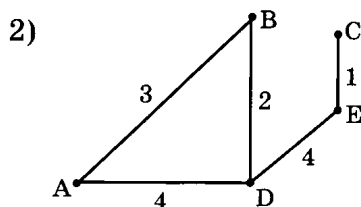
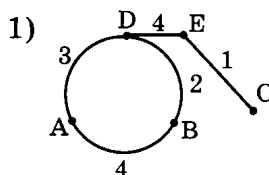
| Участники | А | Б | В | Г |
|-----------|---|---|---|---|
| А | | 3 | 0 | 6 |
| Б | 0 | | 3 | 0 |
| В | 3 | 0 | | 3 |
| Г | 0 | 3 | 3 | |

Для какой пары игроков результаты игры занесены неверно?

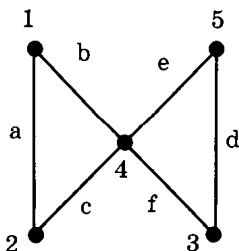
- 1) А и Б
- 2) А и Г
- 3) Б и В
- 4) В и Г

16. В таблице приведен тариф на грузовые перевозки между соседними населенными пунктами. Выберите схему, соответствующую таблице. [7]

| X | A | B | C | D | E |
|---|---|---|---|---|---|
| a | X | 3 | | 4 | |
| B | 3 | X | | 2 | |
| C | | | X | | 1 |
| D | 4 | 2 | | X | 4 |
| E | | | 1 | 4 | X |



17. Дан граф:



Выберите его табличное представление.

1)

| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | | a | | b | |
| 2 | a | | | c | |
| 3 | | | | f | d |
| 4 | b | c | f | | e |
| 5 | | | d | e | |

2)

| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | | a | | b | |
| 2 | | a | | c | |
| 3 | | | | f | d |
| 4 | b | c | f | | e |
| 5 | | | d | e | |

3)

| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | a | | b | | |
| 2 | a | | | c | |
| 3 | | | f | | d |
| 4 | b | c | f | | e |
| 5 | | | d | e | |

4)

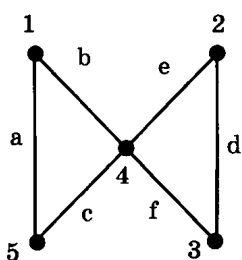
| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | a | a | | b | |
| 2 | | | | c | |
| 3 | | | | f | d |
| 4 | b | c | f | d | e |
| 5 | | | | e | |

18. Дано следующее описание графа:

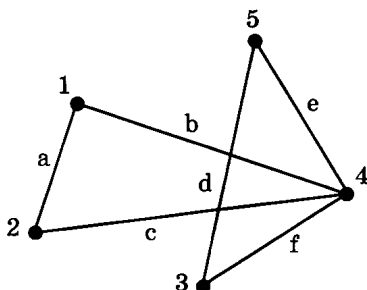
a(1, 2) b(1, 4) c(2, 4) d(3, 5) e(4, 5) f(3, 4)

Укажите его графическое представление.

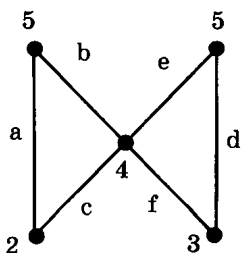
1)



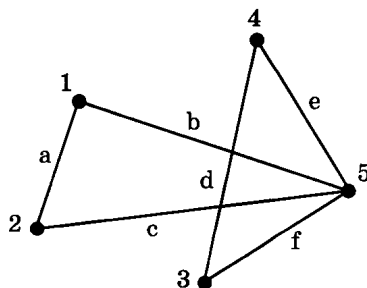
3)



2)



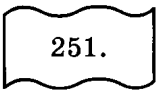
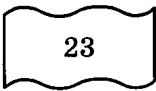
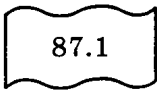
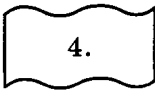
4)



Часть 2

Во второй части теста необходимо самостоятельно получить ответ в виде числа или последовательности букв и записать его в специально отведенное для него место. Каждый символ следует писать в отдельной клетке, начиная с первой.

1. Расставьте фрагменты IP-адреса в верной последовательности.

| | | | |
|---|---|---|---|
|  251. |  23 |  87.1 |  4. |
| А | Б | В | Г |

2. В языке запросов поисковой системы знак «!» перед словом означает, что слово должно быть использовано именно в таком виде. В таблице приведены поисковые запросы, одновременно выданные к одной поисковой системе. Расположите их в порядке возрастания количества страниц, соответствующих запросу

| | |
|---|---|
| 1 | Аэроплан, дирижабль или дилижанс до !Москвы |
| 2 | Аэроплан, дирижабль или дилижанс до Москвы |
| 3 | (Аэроплан дирижабль дилижанс)&(Москва) |
| 4 | Аэроплан&дирижабль&дилижанс&Москва |

3. На основе характеристик строящихся жилых домов (из таблицы) составьте рейтинг фирм (укажите их номера), которые строят наиболее комфортное для проживания семьи жилье, принимая все характеристики равнозначными

| Характеристики/ фирмы | 1) Жилстрой | 2) Метр² | 3) Элит | 4) Мосстрой |
|---|------------------------|--------------------------------|--------------------|------------------------|
| Экологически чистый район — парковая зона | + | — | + | — |
| Развитая транс- портная сеть | — | + | — | + |
| Охранная систе- ма с видеонаблю- дением | + | — | + | — |
| Спутниковое телевидение | + | — | + | — |
| Выделенный ка- нал Интернета | — | + | + | + |
| Домофон | + | + | + | — |
| Дополнительная очистка воды | + | — | + | — |
| Тройное остекле- ние окон | — | — | + | — |
| Терморегулируе- мые радиаторы | + | + | + | + |
| Счетчики рас- хода воды | + | — | + | + |
| Подземный пар- кинг | + | — | — | + |
| Фитнес-центр | — | + | + | — |

4. Абонент скачивает из файлообменной сети фильм объемом 700 Мбайт. Фильм скачивается из пяти разных источников, каждый из которых отдает в среднем 30 Кбайт в секунду. Сколько времени займет скачивание фильма? В ответе укажите только число в минутах, слово «минут» или «мин» добавлять не нужно.

4. АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ

Самый большой раздел, «Алгоритмизация и программирование», включает собственно темы «Алгоритмизация и основы программирования» и «Технология программирования» — всего 10 заданий всех трех уровней сложности. Раздел проверяет умение читать блок-схемы, составлять алгоритмы и писать программы.

Далее представлены типовые тренировочные задания по этому разделу и вариант тематического теста.

Первая задача из типовых по теме дается с подробным разбором решения, для остальных задач приведены ответы с комментариями.

При выполнении заданий каждой темы прежде всего надо внимательно прочитать задание, выделить исходные данные и требуемые на выходе алгоритма или программы. Затем составить алгоритм решения задачи и записать его по пунктам или нарисовать блок-схему.

Полезно составить таблицу, содержащую шаги выполнения программы и значения переменных на каждом из шагов, а также значения проверяемых условий.

При решении задач с массивами полезно рассмотреть выполнение алгоритма или программы на массиве небольшой размерности (4–5 элементов).

Обязательно надо проверять работу алгоритма в граничных и критических точках (максимальное, минимальное значение, равенство нулю, положительные, отрицательные значения и т.п.).

Проверять все ветвления в программе.

Использовать комментарии в программах, чтобы после написания программы выполнить ее проверку, облегчить чтение программы проверяющим экспертам.

Все записи выполнять четким разборчивым почерком, на бланках № 1 и 2 следовать приведенным образцам написания символов.

ПРИМЕРЫ ЗАДАНИЙ С РЕШЕНИЯМИ И ОТВЕТАМИ

1. Чему будет равно значение переменной *s* после выполнения фрагмента программы?

| БЕЙСИК | ПАСКАЛЬ | ПСЕВДОКОД |
|---|---|---|
| 10 <i>x</i> = 0 20 <i>y</i> = <i>x</i> 30 <i>h</i> 1 = 1 40 <i>h</i> 2 = 2 50 <i>s</i> = 2 60 While (<i>x</i> <2) or (<i>y</i> <=5) 70 <i>s</i> = <i>s</i> + 1 80 <i>x</i> = <i>x</i> + <i>h</i> 1 90 <i>y</i> = <i>y</i> + <i>h</i> 2 100 Wend | <i>x</i> := 0; <i>y</i> := <i>x</i> ; <i>h</i> 1 := 1; <i>h</i> 2 := 2; <i>s</i> := 2; While (<i>x</i> <2) or (<i>y</i> <=5) do begin <i>s</i> := <i>s</i> + 1; <i>x</i> := <i>x</i> + <i>h</i> 1; <i>y</i> := <i>y</i> + <i>h</i> 2; end; | <i>x</i> := 0 <i>y</i> := <i>x</i> <i>h</i> 1 := 1 <i>h</i> 2 := 2 <i>s</i> := 2 <u>пока</u> (<i>x</i> <2) <u>или</u> (<i>y</i> <=5) <i>s</i> := <i>s</i> + 1 <i>x</i> := <i>x</i> + <i>h</i> 1 <i>y</i> := <i>y</i> + <i>h</i> 2 <u>кц</u> |

- 1) 5
- 2) 2
- 3) 3
- 4) 4

Решение

Сначала надо записать в таблицу значения всех переменных в начале программы, а затем рассмотреть каждый шаг цикла Пока. Также следует следить за результатами проверки условий. Тело цикла выполняется, пока условие (*x*<2) or (*y*<=5) истинно. Как только условие становится ложным, происходит выход из цикла. Последнее значение переменной будет искомым ответом к задаче. Логическое условие представляет собой дизъюнкцию двух простых выражений (высказываний). Оно истинно (будем обозначать 1) тогда, когда истинно хотя бы одно из них.

| Истинность условия ($x < 2$) or ($y \leq 5$) | x | y | Н1 не изм. | Н2 не изм. | s | Шаг цикла |
|---|---|---|---------------|---------------|---|--------------|
| 1 | 0 | 0 | 1 | 2 | 2 | 0 |
| 1 | 1 | 2 | 1 | 2 | 3 | 1 |
| 1 | 2 | 4 | 1 | 2 | 4 | 2 |
| 0 — ложно, выход из цикла | 3 | 6 | 1 | 2 | 5 | 3 |

$S=5$ — искомое значение.

Ответ: 1.

2. Чему будет равно значение переменной z после выполнения фрагмента программы?

| БЕЙСИК | ПАСКАЛЬ | ПСЕВДОКОД |
|--|---|---|
| $x = -3$ $y = x + 5$ if ($y > 0$) and ($y < 5$) $z = x + y$ then $z = x * y$ else $z = x + y$ 60 $z = x * y$ $z = x - z$ | $x := -3;$ $y := x + 5;$ if ($y > 0$) and ($y < 5$) then $z := x * y$ else $z := x + y;$ $z := x - z;$ | $x := -3$ $y := x + 5$ <u>если</u> ($y > 0$) <u>и</u> ($y < 5$) <u>то</u> $z := x * y$ <u>иначе</u> $z := x + y$ <u>все</u> $z := x - z$ |

1) 1

2) 2

3) 3

4) 4

Ответ: 3.

3. Для определения наибольшего из трех значений переменных a , b и c необходимо написать фрагмент программы.

| | БЕЙСИК | ПАСКАЛЬ | ПСЕВДОКОД |
|----|--|---|--|
| 1) | <pre>10 d = a 20 if b > d then d = b 30 if c > d then d = c 40 print d</pre> | <pre>d := a; if b > d then d := b; if c > d then d := c; WriteLn (d);</pre> | <pre>d := a если b > d то d := b все если c > d то d := c все напечатать d</pre> |
| 2) | <pre>10 if a>b>c then d = a 20 print d</pre> | <pre>if (a>b>c) then d :=a; WriteLn (d);</pre> | <pre>если (a>b>c) то d := a все напечатать d</pre> |
| 3) | <pre>10 if b>a>c then d = c 20 print d</pre> | <pre>if (b>a>c) then d :=c; WriteLn (d);</pre> | <pre>если (b>a>c) то d := c все напечатать d</pre> |
| 4) | <pre>10 if c>a>b then d = b 20 print d</pre> | <pre>if (c>a>b) then d :=b; WriteLn (d);</pre> | <pre>если (c>a>b) то d := b все напечатать d</pre> |

Ответ: 1.

4. Чему будет равно значение переменной a после выполнения фрагмента программы?

| БЕЙСИК | ПАСКАЛЬ | ПСЕВДОКОД |
|--|---|--|
| <pre>10 x= 1 20 h1 = 1 30 h2 = 2 40 a = - 1 50 While (x<=3)</pre> | <pre>x := 1; h1 := 1; h2 := 2; a := -1; While (x <=3) do begin y := 2;</pre> | <pre>x := 1 h1 := 1 h2 := 2 a := -1 пока (x <= 3)</pre> |

Продолжение табл.

| БЕЙСИК | ПАСКАЛЬ | ПСЕВДОКОД |
|--|---|--|
| 60 Y = 2 70 WHILE (Y<=3) 80 A = A + 1 90 Y = Y + H2 100 WEND 110 X = X + H1 120 WEND | While (y <= 3) do begin a := a + 1; y := y + h2; end; x := x + h1; end; | y := 2 <u>пока</u> (y <= 3) a := a + 1 y := y + h2 <u>кц</u> x := x + h1 <u>кц</u> |

1) 1

2) 2

3) 3

4) 4

Ответ: 2.

5. Чему будет равно значение переменной **b** после выполнения фрагмента программы?

| БЕЙСИК | ПАСКАЛЬ | ПСЕВДОКОД |
|--|--|--|
| 10 B = 0 20 A = -1 30 FOR I = 1 TO 3 40 X = A * I 50 A = -A 60 B = B + X 70 NEXT I | b := 0; a := -1; for i := 1 to 3 do begin x := a * i; a := -a; b := b + x; end; | b := 0 a := -1 <u>нц для i от 1 до 3</u> x := a * i a := -a b := b + x <u>кц</u> |

1) -1

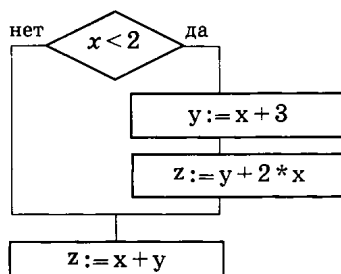
2) -2

3) -3

4) -4

Ответ: 2.

6. Задан фрагмент блок-схемы



Ему соответствует фрагмент программы

| | БЕЙСИК | ПАСКАЛЬ | ПСЕВДОКОД |
|----|---|---|--|
| 1) | 10 IF NOT(X < 2) THEN 40 20 Y = X + 3 30 Z = Y + 2 * X 40 Z = X + Y | if x < 2 then begin y := x + 3; z := y + 2 * x; end; z := x + y; | <u>если</u> x < 2 <u>то</u> y := x + 3 z := y + 2 * x <u>все</u> z := x + y |
| 2) | 10 IF (X < 2) THEN 40 20 Y = X + 3 30 Z = Y + 2 * X 40 Z = X + Y | if x >= 2 then begin y := x + 3; z := y + 2 * x; end z := x + y; | <u>если</u> x >= 2 <u>то</u> y := x + 3 z := y + 2 * x <u>все</u> z := x + y |
| 3) | 10 IF X < 2 THEN Y = X + 3 20 Z = Y + 2 * X 30 Z = X + Y | if x < 2 then y := x + 3; z := y + 2 * x; z := x + y; | <u>если</u> x < 2 <u>то</u> y := x + 3 <u>все</u> z := y + 2 * x z := x + y |
| 4) | 10 IF X >= 2 THEN Y = X + 3 20 Z = Y + 2 * X 30 Z = X + Y | if x >= 2 then y := x + 3; z := y + 2 * x; z := x + y; | <u>если</u> x >= 2 <u>то</u> y := x + 3 <u>все</u> z := y + 2 * x z := x + y |

1) 1

2) 2

3) 3

4) 4

Ответ: 1.

7. Задан фрагмент программы, обрабатывающий массив X размером N. После выполнения фрагмента программы

| БЕЙСИК | ПАСКАЛЬ | ПСЕВДОКОД |
|---|---|--|
| <pre> Imax:=1;imin:=1 For i:=2 to n If X(i)<x(imin) then imin:=I else If X(i)>x(imax) then imax=I End if End if Next i y:= X(imax) x(imax)=x(imin) x(imin)=y </pre> | <pre> Imax:=1; imin:=1; For i:=2 to n do Begin If X[i]<x[imin] then imin:=I else If X[i]>x[imax] then imax:=I; End; y:= X[1]; x[1]:=x[imax]; x[imax]:=y; </pre> | <pre> Imax:=1 imin:=1 Нц для I от 2 до n если X[i]<x[imin] то imin:=I иначе если X[i]>x[imax] то imax:=I все все кц y:= X[imax] x[imax]:=x[imin] x[imin]:=y </pre> |

| | |
|----|--|
| 1) | Массив остался без изменений |
| 2) | Максимальный и минимальный элементы поменялись местами |
| 3) | Максимальный элемент стал первым в массиве |
| 4) | Минимальный элемент стал первым в массиве |

Решение

Для представления одномерного массива нарисуйте две строки таблицы. 1-я строка — номер элемента, 2-я строка — значение элемента, например, для $n=5$.

| № элемента массива | 1 | 2 | 3 | 4 | 5 | Imax | Imin |
|---------------------------|---|----|----|---|---|------|------|
| Значение элемента массива | 4 | 12 | -5 | 3 | 2 | 1 | 1 |

В начале программы запоминается номер первого элемента дважды: в переменных i_{\max} и i_{\min} . Начальное значение параметра цикла равно 2. Далее нарисуйте таблицу, в которой будете прослеживать значения переменной цикла i и переменных программы i_{\max} , i_{\min} , y , $x[i]$. Во фрагменте программы элементы массива сравниваются так: сначала 2-й элемент с первым. Запоминаются номера максимального и минимального элементов. Эта процедура выполняется для всех остальных элементов массива. Результат проверки условия — истина (обозначим 1) или ложь (обозначим 0).

| i | $X[i] < x[i_{\min}]$ | $X[i] > x[i_{\max}]$ | i_{\min} | i_{\max} | $X[i]$ | $X[i_{\max}]$ | $X[1]$ | y |
|----------------|----------------------|----------------------|------------|------------|--------|---------------|--------|-----|
| – | – | – | 1 | 1 | – | – | – | – |
| 2 | 0 | 1 | 1 | 2 | 12 | – | – | – |
| 3 | 1 | 0 | 3 | 2 | –5 | – | – | – |
| 4 | 0 | 0 | 3 | 2 | 3 | – | – | – |
| 5 | 0 | 0 | 3 | 2 | 2 | – | – | – |
| Выход из цикла | | | | | | 12 | 4 | 4 |
| | | | | | | 12 | 12 | 4 |
| | | | | | | 4 | 12 | 4 |

Т.е. в результате работы цикла получим номера максимального и минимального элементов массива — соответственно 2 и 3. Далее, используя вспомогательную переменную y , ставим максимальный элемент на первое место, а первый элемент — на его место.

Ответ: 3.

8. Дан фрагмент программы

| БЕЙСИК | ПАСКАЛЬ | ПСЕВДОКОД |
|--|---|---|
| Const n=5 Const m=5 Defint I,j Dim A(n,m) as integer Begin For i=1 to n do For j=1 to m do If i=j then A(I,j)=1 else A(I,j)=0 End if Next j Next i ... | Const n=5; Const m=5; Var I,j:integer; A: array[1..n,1..m] of integer; Begin For i:=1 to n do For j:=1 to m do If i=j then A[I,j]:=1 else A[I,j]:=0; ... | Константа n=5 Константа m=5 Арг цел I,j Арг цел таб A[1:n,1:m] <u>начало</u> <u>Нц для I от 1 до n</u> <u>Нц для j от 1 до m</u> <u>если i=j</u> <u>то A[I,j]:=1</u> <u>иначе A[I,j]:=0</u> <u>Все</u> <u>Кц</u> <u>кц</u> ... |

Чему равна сумма значений элементов массива А?

- 1) 1 2) 0 3) 5 4) 25

Ответ: 3.

9. Имеется таблица, состоящая из трех строк и трех столбцов. Чему будет равно значение элемента таблицы A(1,3) после выполнения фрагмента программы?

| БЕЙСИК | ПАСКАЛЬ | ПСЕВДОКОД |
|---|---|--|
| 10 FOR I = 1 TO 3 20 FOR J = 1 TO 3 30 IF J > I THEN 60 40 A(I, J) = I - J 50 GOTO 70 60 A(I, J) = I + J 70 NEXT J 80 NEXT I | for i:= 1 to 3 do for j:=1 to 3 do if j > i then A [i, j]:= i + j else A [i, j]:= i - j; | <u>нц для i от 1 до 3</u> <u>нц для j от 1 до 3</u> <u>если j > i</u> <u>то A [i, j]:= i + j</u> <u>иначе</u> A[i, j] := i - j <u>все</u> <u>кц</u> <u>кц</u> |

- 1) 1 2) 2 3) 3 4) 4

Ответ: 4.

10. Задан массив X размером N. После выполнения фрагмента программы

| БЕЙСИК | ПАСКАЛЬ | ПСЕВДОКОД |
|--|---|--|
| $y=x(n)$ For i=n to 2 step -1 $x(i)=x(i-1)$ Next i $x(1)=y$ | $y:=x[n];$ For i:=n down to 2 do $x[i]:=x[i-1];$ $x[1]:=y;$ | $y:=x[n]$ <u>Нц для I</u> от n <u>до</u> 2 <u>шаг</u> -1 $x[i]:=x[i-1]$ <u>кц</u> $x[1]:=y$ |

| | |
|----|--|
| 1) | Массив остался без изменений |
| 2) | Все элементы получили значение X[1] |
| 3) | Первый элемент стал последним, а остальные переместились на позицию влево |
| 4) | Последний элемент стал первым, а остальные переместились на позицию вправо |

Ответ: 4.

11. Задан массив X размером N. После выполнения фрагмента программы

| БЕЙСИК | ПАСКАЛЬ | ПСЕВДОКОД |
|---|---|--|
| $y=x(n)$ For i=2 to n $x(i)=x(i-1)$ Next i $x(1)=y$ | $y:=x[n];$ For i:=2 to n do $x[i]:=x[i-1];$ $x[1]:=y;$ | $y:=x[n]$ <u>Нц для i от 2 до n</u> $x[i]:=x[i-1]$ <u>кц</u> $x[1]:=y$ |

| | |
|----|--|
| 1) | Массив остался без изменений |
| 2) | Все элементы, кроме первого, получили значение X[1] |
| 3) | Первый элемент стал последним, а остальные переместились на позицию влево |
| 4) | Последний элемент стал первым, а остальные переместились на позицию вправо |

Ответ: 2.

12. Задан фрагмент программы, обрабатывающий массив X размером N. После выполнения фрагмента программы

| БЕЙСИК | ПАСКАЛЬ | ПСЕВДОКОД |
|--|---|---|
| <pre> Imax:=1;imin:=1 For i:=2 to n If X(i)<x(imin) then imin:=I else If X(i)>x(imax) then imax:=I End if End if Next i y:= X(imax) x(imax)=x(imin) x(imin)=y </pre> | <pre> Imax:=1;imin:=1; For i:=2 to n do Begin If X[i]<x[imin] then imin:=I else If X[i]>x[imax] then imax:=I End; y:= X[imax]; x[imax]:=x[imin]; x[imin]:=y; </pre> | <pre> Imax:=1 imin:=1 <u>Нц для I от 2 до n</u> <u>если</u> X[i]<x[imin] <u>то</u> imin:=I <u>иначе</u> <u>если</u> X[i]>x[imax] <u>то</u> imax:=I <u>все</u> <u>все</u> <u>кц</u> y:= X[imax] x[imax]:=x[imin] x[imin]:=y </pre> |

| | |
|----|--|
| 1) | Массив остался без изменений |
| 2) | Максимальный и минимальный элементы поменялись местами |
| 3) | Максимальный элемент стал первым в массиве |
| 4) | Минимальный элемент стал первым в массиве |

Ответ: 2.

13. Цепочка из 5 бусин, каждая из которых помечена набором символов: A1, A2, B1, B2 и C1, образована по следующему правилу. На первом месте находится бусина, не содержащая цифры «2». На последнем месте находится бусина, не содержащая буквы С. В середине находится одна из бусин A2, B1, C1, которой нет на втором месте. Какая из перечисленных ниже цепочек создана по этому правилу?

- | | |
|---------------|---------------|
| 1) B2B1C1A1A2 | 3) B1A1A2B2C1 |
| 2) C1A1A2B1B2 | 4) A1C1B1C1B2 |

Решение

В 1-й цепочке первая бусина содержит цифру «2» — нарушение правила. Во 2-й цепочке в середине (3 место) находится такая же бусина, что и на 2-м месте — нарушение правила. В 3-й цепочке на последнем месте находится бусина с буквой С — нарушение правила. 4-я цепочка удовлетворяет всем правилам.

Ответ: 4.

14. Цепочка из 5 бусин, каждая из которых помечена набором символов: A1, B1, B2, C1 и C2, образована по следующему правилу. На первом месте находится бусина, не содержащая цифры «1». На последнем месте находится бусина, не содержащая буквы В. В середине находится одна из бусин A1, B2, C1, которой нет на втором месте. Какая из перечисленных ниже цепочек создана по этому правилу?

- 1) B2C1C1B2C2
- 2) C2C1C2B2C2
- 3) B2C2C1B2C2
- 4) A2C1C2B2C2

Ответ: 3.

15. В клетки полосы в один ряд записывают 4 символа по следующему правилу: на первом месте может быть один из символов А, В, С. На третьем месте — символы А,С,Д,Е, не стоящие на первом и последнем месте. На последнем месте — символы В,С,Е, не стоящие на втором месте.

- 1) BBCB
- 2) BCDE
- 3) ACDD
- 4) ADBA

Ответ: 2.

16. Даны 3 числа $A=10$, $B=3$, $C=5$. Были выполнены следующие действия: сравнили A и B , запомнили большее число. Затем сравнили полученное число и число C и запомнили большее из них. Какое число запомнили?

- 1) 10
- 2) 5
- 3) 3
- 4) 18

Ответ: 1.

17. Даны 4 числа $A_1=34$, $A_2=45$, $A_3=50$, $A_4=16$. Были выполнены следующие действия: сравнили A_1 и A_2 , запомнили большее число B_1 . Затем сравнили A_3 и A_4 , запомнили большее число B_2 . Сравнили B_1 и B_2 , запомнили большее из них. Какое число запомнили?

- 1) 16
- 2) 34
- 3) 45
- 4) 50

Ответ: 4.

18. Исполнитель РОБОТ может передвигаться на одну клетку **вверх**, **вниз**, **вправо** и **влево** прямоугольного клетчатого поля, на котором расположены горизонтальные и вертикальные стенки. Двигаться вперед он может только тогда, когда стенок перед ним нет.

Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится РОБОТ:

| | | | |
|--------------------|-------------------|-------------------|--------------------|
| сверху свободно | снизу свободно | слева свободно | справа свободно |
|--------------------|-------------------|-------------------|--------------------|

Цикл
ПОКА *< условие >* **делать**
команда 1
 ...
команда N
Конец

выполняется, пока условие истинно, иначе происходит переход на следующую строку.

| | | | | | |
|----|----|--|--|----|----|
| | | | | | •А |
| | | | | | |
| | | | | •С | |
| •В | | | | | |
| | | | | | |
| | •D | | | •X | |

Выберите точку, из которой РОБОТ начал движение, если он остановился клетке X, перемещаясь по следующему алгоритму:

Пока (слева свободно) **делать**
 Если снизу свободно **то**
 идти вниз
Иначе
 идти влево

конец

- 1) А
- 2) В
- 3) С
- 4) D

Решение

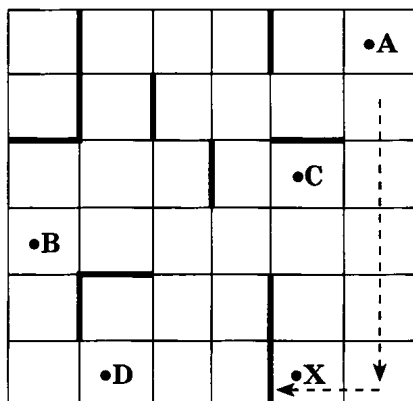
В фигурных скобках приведен комментарий к программе, а движение Робота показано штрих пунктирной линией.

Пока (слева свободно) делать**Если снизу свободно, то
идти вниз**

{истина}

Если {истина}{см. линию-стрелку
вниз от точки А}**Иначе****идти влево**{см. стрелку влево
от правой нижней
клетки к точке X}**конец**

Цикл **Пока** работал один раз, так как из точки X влево идти нельзя.

**Ответ: 1.**

19. Исполнитель РОБОТ может передвигаться на одну клетку **вверх, вниз, вправо и влево** прямоугольного клетчатого поля, на котором расположены горизонтальные и вертикальные стенки. Двигаться вперед он может только тогда, когда стенок перед ним нет.

Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится РОБОТ:

| | | | |
|--------------------|-------------------|-------------------|--------------------|
| сверху свободно | снизу свободно | слева свободно | справа свободно |
|--------------------|-------------------|-------------------|--------------------|

Цикл

ПОКА < *условие* > **делать**
команда 1

...

команда N

Конец

выполняется, пока условие истинно, иначе происходит переход на следующую строку.

| | | | | | |
|-----|-----|--|--|-----|-----|
| | • X | | | | • A |
| | | | | | |
| | | | | • C | |
| • B | | | | | |
| | | | | | |
| | • D | | | | |

Выберите точку, из которой РОБОТ начал движение, если он остановился в клетке X, перемещаясь по следующему алгоритму:

Пока (слева свободно) **делать**

Если сверху свободно, то
идти вверх

Иначе

идти влево

Если сверху свободно, то
идти вверх

конец

- | | |
|------|------|
| 1) A | 3) C |
| 2) B | 4) D |

Ответ: 3.

20. Исполнитель РОБОТ может передвигаться на одну клетку **вверх, вниз, вправо и влево** прямоугольного клетчатого поля, на котором расположены горизонтальные и вертикальные стенки. Двигаться вперед он может только тогда, когда стенок пред ним нет.

Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится РОБОТ:

| | | | |
|--------------------|-------------------|-------------------|--------------------|
| сверху свободно | снизу свободно | слева свободно | справа свободно |
|--------------------|-------------------|-------------------|--------------------|

Цикл

ПОКА $\langle \text{условие} \rangle$ делать
команда1

...

командаN

Конец

выполняется, пока условие истинно, иначе происходит переход на следующую строку.

Сколько клеток приведенного лабиринта соответствуют требованию, что, выполнив предложенную ниже программу, РОБОТ остановится в той же клетке, с которой он начал движение?

НАЧАЛО

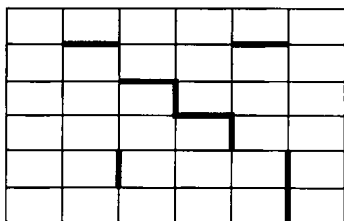
ПОКА $\langle \text{сверху свободно} \rangle$ вверх

ПОКА $\langle \text{справа свободно} \rangle$ вправо

ПОКА $\langle \text{снизу свободно} \rangle$ вниз

ПОКА $\langle \text{слева свободно} \rangle$ влево

КОНЕЦ



1) 1

2) 2

3) 3

4) 4

Ответ: 1.

21. Исполнитель РОБОТ может передвигаться на одну клетку **вверх, вниз, вправо и влево** прямоугольного клетчатого поля, на котором расположены горизонтальные и вертикальные стенки. Двигаться вперед он может только тогда, когда стенок перед ним нет, иначе РОБОТ разрушается.

Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится РОБОТ:

| | | | |
|--------------------|-------------------|-------------------|--------------------|
| сверху свободно | снизу свободно | слева свободно | справа свободно |
|--------------------|-------------------|-------------------|--------------------|

Цикл

ПОКА *< условие >* делать
команда1

...

командаN

Конец

выполняется, пока условие истинно, иначе происходит переход на следующую строку.

Сколько клеток приведенного лабиринта соответствуют требованию, что, выполнив предложенную ниже программу, РОБОТ остановится в той же клетке, с которой он начал движение?

НАЧАЛО

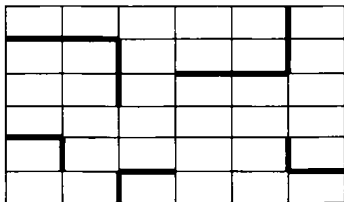
ПОКА **< снизу свободно >** **вниз**

ПОКА **< слева свободно >** **влево**

ПОКА **< сверху свободно >** **вверх**

ПОКА **< справа свободно >** **вправо**

КОНЕЦ



1) 1

2) 2

3) 3

4) 4

Ответ: 4.

22. Исполнитель РОБОТ может передвигаться на одну клетку вверх, вниз, вправо и влево прямоугольного клетчатого поля, на котором расположены горизонтальные и вертикальные стенки. Двигаться вперед он может только тогда, когда стенок перед ним нет.

Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится РОБОТ:

| | | | |
|--------------------|-------------------|-------------------|--------------------|
| сверху свободно | снизу свободно | слева свободно | справа свободно |
|--------------------|-------------------|-------------------|--------------------|

Цикл

ПОКА < условие > делать
команда1

...

командаN

Конец

выполняется, пока условие истинно, иначе происходит переход на следующую строку.

Сколько клеток приведенного лабиринта соответствуют требованию, что, выполнив предложенную ниже программу, РОБОТ остановится в той же клетке, с которой он начал движение?

Примечание: Если Робот выполнит движение в сторону стены, то он разрушится, а выполнение программы прервется.

НАЧАЛО

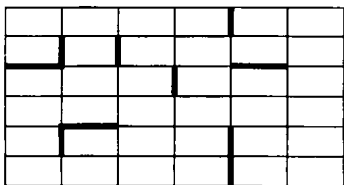
ПОКА < слева свободно > влево

ПОКА < снизу свободно > вниз

ПОКА < справа свободно > вправо

ПОКА < сверху свободно > вверх

КОНЕЦ



1) 1

2) 2

3) 3

4) 4

Ответ: 2.

23. Исполнитель РОБОТ может передвигаться на одну клетку **вверх, вниз, вправо и влево** прямоугольного клетчатого поля, на котором расположены горизонтальные и вертикальные стенки. Двигаться вперед он может только тогда, когда стенок перед ним нет. Команда Крась закрашивает клетку, в которой стоит Робот.

Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится РОБОТ:

| | | | |
|--------------------|-------------------|-------------------|--------------------|
| сверху свободно | снизу свободно | слева свободно | справа свободно |
|--------------------|-------------------|-------------------|--------------------|

Цикл

ПОКА < *условие* > **делать**
команда1

...

командаN

Конец

выполняется, пока условие истинно, иначе происходит переход на следующую строку.

Сколько клеток (K1) приведенного лабиринта соответствуют требованию, что, выполнив предложенную ниже программу, РОБОТ остановится в той же клетке, с которой он начал движение? Сколько клеток (K2) приведенного лабиринта будут при этом закрашены? В ответе укажите сумму таких клеток (K1+K2)?

Примечание: Если Робот выполнит движение в сторону стены, то он разрушится, а выполнение программы прервется.

НАЧАЛО

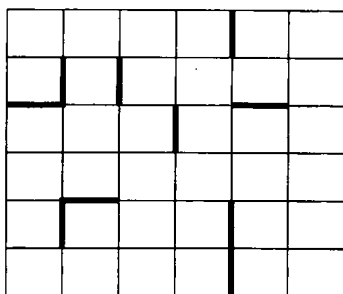
ПОКА < **слева свободно** > **влево крась**

ПОКА < **снизу свободно** > **вниз крась**

ПОКА < **справа свободно** > **вправо**

ПОКА < **сверху свободно** > **вверх**

КОНЕЦ



1) 4

2) 5

3) 6

4) 7

Ответ: 2.

24. Запишите значение переменной f после выполнения фрагмента алгоритма для переменной целого типа $x=789$.

| БЕЙСИК | ПАСКАЛЬ | ПСЕВДОКОД |
|------------------------------|---------------------------|---------------------------------------|
| $B=x \bmod 100 \setminus 10$ | $B:=x \bmod 100 \div 10;$ | $B:=\text{div}(\text{mod}(x,100),10)$ |
| $C=x \bmod 100 \setminus 10$ | $C:=x \bmod 10;$ | $C:=\text{mod}(x,10)$ |
| $D=x \setminus 10$ | $D:=x \div 100;$ | $D:=\text{div}(x,10)$ |
| $Y=100*b+10*c+d$ | $Y:=100*b+10*c+d;$ | $Y:=100*b+10*c+d$ |
| $f=x+y$ | $f:=x+y;$ | $f:=x+y$ |

Ответ: _____*Решение*

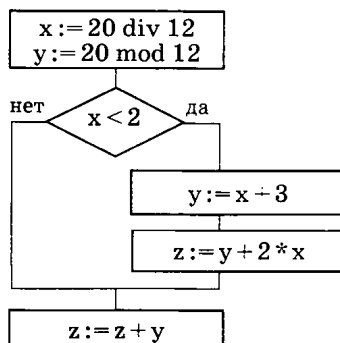
Вспомним, что операция \bmod определяет остаток от деления целых чисел, а \div — целую часть от деления целых чисел. Деление целых десятичных чисел на 10, 100, 1000 и т.д. с использованием этих функций дает цифры числа. А умножение цифр на 10, 100, 1000 и т.д. позволяет получать числа, где одни и те же цифры имеют разный вес в числе, т.е. являются единицами, десятками, сотнями, тысячами и т.д. Кроме того, следует знать приоритеты выполнения математических операций. Mod и Div — операции одного порядка, поэтому выполняются последовательно, как записаны в выражении (анало-

гично это: сложение и вычитание, умножение и деление). Следует нарисовать таблицу и указать в первом столбце номера команд (мы их для удобства также запишем), а во втором столбце записывать значения переменных после выполнения каждой команды (оператора).

| КОМАНДЫ | ЗНАЧЕНИЯ |
|--|--|
| $X=789$ | 789 |
| $B:=x \bmod 100 \operatorname{div} 10$ | $789 \bmod 100=89$ $89 \operatorname{div} 10=8$ $b=8$ — средняя цифра числа |
| $C:=x \bmod 10;$ | $789 \bmod 10=9$ $C=9$ — последняя цифра числа |
| $D:=x \operatorname{div} 100;$ | $789 \operatorname{div} 100=7$ $D=7$ — первая цифра числа Т.е. в исходном числе следующий порядок цифр DBC |
| $Y:=100*b+10*c+d;$ | $100*8+10*9+7=897$, $y=897$ В новом числе порядок цифр — BCD |
| $f:=x+y;$ | Сумма: $789+897=1686$ |

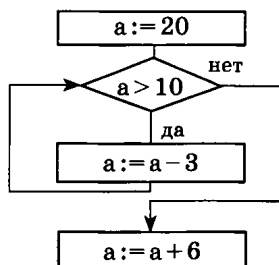
Ответ: 1686.

25. Запишите значение переменной z после выполнения фрагмента алгоритма



Ответ: 10.

26. Запишите значение переменной z после выполнения фрагмента алгоритма



Ответ: 14.

27. Исполнитель Кузнечик живет на числовой оси и имеет следующую систему команд:

Вперед a — Кузнечик прыгает вправо на a единиц;

Назад b — Кузнечик прыгает назад на b единиц.

Переменные a и b могут принимать любые целые положительные значения. Кузнечик из некоторой точки, выполняя только команды

1) Вперед 5

2) Назад 2,

оказался в точке 7. Из какой точки он начал движение, если выполнил 4 команды, а первой его командой была команда Вперед 5? Если таких точек несколько, то перечислите их через запятую в порядке возрастания.

Решение

Пусть n и m соответственно количество первой и второй команд, а x — искомая точка. Тогда

$$x + 5*n - 2*m = 7$$

Рассмотрим комбинации n и m ($n > 0$, так как первой его командой была команда Вперед 5):

| m | n | x |
|-----|-----|-----|
| 0 | 4 | -13 |
| 1 | 3 | -6 |
| 3 | 1 | 8 |
| 2 | 2 | 1 |

Ответ: -13, -6, 1, 8.

28. С помощью двух сосудов, А объемом 5 литров и В объемом 13 литров, требуется отмерить 7 литров, причем вначале оба сосуда пусты и допустимы только следующие действия:

- если сосуд А пустой, то наполнить его до краев;
- сосуд А перелить в сосуд В;
- если сосуд В наполнен до краев, то освободить его так, чтобы он остался пустым.

Сколько раз будет наполняться сосуд В до краев, прежде чем будет отмерен требуемый объем в литрах?

Ответ: 1.

29. Исполнитель Вычислитель имеет следующую систему команд:

- 1) к числу прибавить 2
- 2) число умножить на 5.

Запишите порядок команд в программе получения из числа 7 числа 247, содержащей не более 6 команд, указывая лишь номера команд.

(Например, программа **21211** — это программа для исполнителя с командами 1) к числу прибавить 3 и 2) число умножить на 4.

умножь на 4

прибавь 3

умножь на 4

прибавь 3

прибавь 3,

которая преобразует число 2 в число 50.)

Ответ: 121121.

30. Исполнитель Черепашка перемещается на экране компьютера, оставляя след в виде линии. В каждый конкретный момент известно положение исполнителя и направление его движения. У исполнителя существуют две команды:

Вперед n , где n — целое число, вызывает передвижение черепашки на n шагов в направлении движения.

Направо m , где m — целое число, вызывает изменение направления движения на m градусов по часовой стрелке (от 0° до 359°).

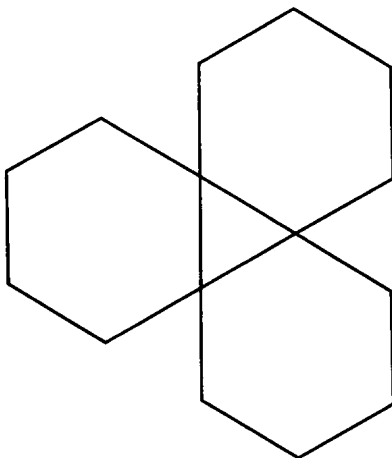
Запись **Повтори 5 [Команда1 Команда2]** означает, что последовательность команд в скобках повторится 5 раз.

Исполнитель интерпретирует эту запись как одну команду.

Черепашке был дан для исполнения следующий алгоритм:

Повтори 3 [Повтори 6[Вперед 100 Направо X] Направо Y вперед 100]

На экране появилась следующая фигура.



Определить и записать в ответе значения X и Y через запятую.

Ответ: 60, 120.

31. Исполнитель РОБОТ может передвигаться на одну клетку **вверх**, **вниз**, **вправо** и **влево** прямоугольного клетчатого поля, на котором расположены горизонтальные и вертикальные стенки. Двигаться вперед

он может только тогда, когда стенок перед ним нет, иначе он разрушается.

| | | | | | |
|----|----|--|--|----|----|
| | •X | | | | •A |
| | | | | | |
| | | | | •C | |
| •B | | | | | |
| | | | | | |
| | •D | | | | |

Сколько клеток пройдет Робот, перемещаясь по следующему алгоритму из точки В (считать исходную и конечную точки движения):

Пока (справа свободно) **делать**

Если сверху свободно, то
идти вверх

Иначе

идти вправо

Если сверху свободно, то
идти вверх

Иначе

идти вправо

конец

Ответ: 7.

32. Исполнитель Черепашка перемещается на экране компьютера, оставляя след в виде линии. В каждый конкретный момент известно положение исполнителя и направление его движения. У исполнителя существуют две команды:

Вперед n , где n — целое число, вызывает передвижение черепашки на n шагов в направлении движения.

Направо m , где m — целое число, вызывает изменение направления движения на m градусов по часовой стрелке (от 0° до 359°).

Запись Повтори 5 [Команда1 Команда2] означает, что последовательность команд в скобках повторится 5 раз.

Исполнитель интерпретирует эту запись как одну команду.

Черепашке был дан для исполнения следующий алгоритм:

Повтори 60 [Вперед 100 Направо 12]

Сколько внутренних углов будет иметь получившаяся фигура?

Определить и записать в ответе число.

Ответ: 30.

ПРИМЕРЫ ЗАДАНИЙ С РЕШЕНИЯМИ И ОТВЕТАМИ (ЧАСТЬ С)

- С1.** Прочесть фрагмент программы на языке программирования и исправить допущенные ошибки. *Применение знаний и умений в новой ситуации. Повышенный уровень.* Примерное время выполнения задания — 30 мин.

Ниже приведены 6 заданий. 1-е задание дается с разбором, а для остальных, при необходимости, пояснение можно найти в разделе «Ответы».

1. Требовалось написать программу вычисления НОД (наибольший общий делитель) трех целых чисел.

Программист написал программу с ошибкой.

| ПАСКАЛЬ | БЕЙСИК |
|--|--|
| <pre> Var a,b,c,nod1,m:integer; Procedure nod(x,y:integer; var z:integer); Var n:integer; Begin z:=x; n:=y; While z<>n do </pre> | <pre> Defint a,b,c,m,nod1 Sub nod(x as integer,y as integer,z as integer) z=x:n=y While z <> n If z>n then z=z-n else n=n-z End if </pre> |

| ПАСКАЛЬ | БЕЙСИК |
|---|--|
| <pre> begin If z>n then z:=z-n else n:=n-z; end; end; Begin Writeln('Введите три целых числа'); Readln(a); Readln(b); Readln(c); Nod(a,c,nod1); Nod(a,b,nod1); Nod(c,b,m); Writeln ('НОД=NOD=', m); readln; End.</pre> | <pre> Wend End sub cls Input "a=";a Input "b=";b Input "c=";c Call nod(a,c,nod1) Call nod(a,b,nod1) Call nod(c,b,m) Print "НОД=";m End</pre> |

- Привести пример таких исходных данных a, b и c , для которых программа дает неправильный результат.
- Исправить ошибку в программе так, чтобы она соответствовала поставленной задаче.
- Укажите, какая часть программы является лишней.

Разбор задания 1.

В программе используется процедура $\text{Nod}(x, y, z)$, где x и y — исходные два целых числа, z — результат, наибольший общий делитель. Для определения НОД трех чисел необходимо найти НОД первых двух, а потом НОД третьего числа и найденного НОД первых двух чисел. Например, найти НОД трех чисел: 10, 100, 15. $\text{НОД}(10, 100) = 10$, а $\text{НОД}(10, 15) = 5$. Искомый НОД трех чисел равен 5. В программе допущена ошибка при обращении к процедуре: **$\text{Nod}(c, b, m)$** , т.е. находится НОД третьего и второго числа без учета первого.

Ответ на 1-й вопрос. При следующих данных программа будет давать ошибку: $a=3$, $b=81$, $c=9$, $Nod=9$.

(Кстати, при $a=81$, $b=9$, $c=3$, $Nod=3$, т.е. результат будет правильным). Но в заданиях С1 следует указывать те исходные данные, при которых результат получается неправильным. Т.е. ответ на первый вопрос: $a=3$, $b=81$, $c=9$, $Nod=9$.

Отсюда ответ на 2-й вопрос. Следует заменить второе обращение к процедуре $Nod(c,b,m)$ на следующее: $Nod(c,nod1,m)$. Ниже приведен текст программы с исправлением.

Ответ на 3-й вопрос. Первое обращение к процедуре $Nod(a,c,nod1)$; является лишним.

| Правильная программа | |
|--|--|
| ПАСКАЛЬ | БЕЙСИК |
| <pre> Var a,b,c,nod1,m:integer; Procedure nod(x,y:integer; var z:integer); Var n:integer; Begin z:=x; n:=y; While z<>n do begin If z>n then z:=z-n else n:=n-z; end; end; Begin Writeln('Введите три целых числа'); Readln(a); Readln(b); Readln(c); Nod(a,b,nod1); Nod(c,nod1,m); {OK} Writeln ('НОД=NOD=', m); readln; End.</pre> | <pre> Defint a,b,c,m,nod1 Sub nod(x as integer,y as integer,z as integer) z=x:n=y While z <> n If z>n then z=z-n else n=n-z End if Wend End sub cls Input "a=";a Input "b=";b Input "c=";c Call nod(a,b,nod1) Call nod(c,nod1,m) Print "НОД=";m End</pre> |

2. Требовалось написать программу вычисления НОК (наименьшего общего кратного) двух целых чисел a и b , используя НОД (наибольший общий делитель)

$$\text{НОК} = (a*b)/\text{НОД}$$

Программист написал программу с ошибкой.

| ПАСКАЛЬ | БЕЙСИК |
|---|--|
| <pre> Var a,b,m,n,nok:integer; Begin Readln(a); Readln(b); Readln(m); M:=a;n:=b; While m=n do begin If m>n then m:=m-n else n:=n-m; end; Nok:=(a*b) div m; Writeln ('nok=', nok); End.</pre> | <pre> Defint a,b,m,n,n Input "a=";a Input "b=";b Input "m=";m M=a:n=b While m = n If m>n then m=m-n else n=n-m End if Wend N=(a*b) \ m Print "nok=";n End</pre> |

- Привести пример таких исходных данных a и b , для которых программа дает неправильный результат
 - Исправить ошибку в программе так, чтобы она соответствовала поставленной задаче.
 - Укажите, какая часть программы является лишней.
3. Требовалось написать программу определения симметричности *пятизначного целого числа* A , т.е. читается ли оно одинаково слева направо и справа налево. Программист написал программу с ошибкой.

| ПАСКАЛЬ | БЕЙСИК |
|---|---|
| <pre> Var A:integer; Result:boolean; Begin Writeln('Введите пятизначное целое число'); Readln(A); Result:=(a div 10000=A mod 10) and (A mod 100 div 10=A div 1000 mod 10); Writeln('Число симметрич- ное — ',result); Writeln('Число несимметрич- ное — ',not result); Readln; end.</pre> | <pre> Defint A Input "Введите пятизнач- ное число"; A R=(a \ 10000 = A mod 10) and (A mod 10000 \ 1000= A mod 1000 \ 10) Print "Число симметрич- ное — ";R Print "Число несимметрич- ное — "; not R end</pre> |

- Привести пример такого А, для которого програм-
ма дает неправильный результат.
- Исправить ошибку в программе так, чтобы она со-
ответствовала поставленной задаче.
- Укажите, какая часть программы является лишней.

4. Требовалось написать программу определения симме-
тричности *четырёхзначного целого числа А*, т.е. чита-
ется ли оно одинаково слева направо и справа налево.
Программист написал программу с ошибкой.

| ПАСКАЛЬ | БЕЙСИК |
|---|--|
| <pre> Var A:integer; Result:boolean; Begin Writeln('Введите четырехзнач- ное целое число'); Readln(A); Result:=(a div 1000=A mod 10);</pre> | <pre> Defint A Input "Введите четырех- значное число";A R=(a \ 10000=A mod 10) R=(a \ 10000=A mod 10) or (A mod 10000 \ 1000=A mod 1000 \ 10);</pre> |

Продолжение табл.

| ПАСКАЛЬ | БЕЙСИК |
|---|--|
| <pre>Result:=(a div 1000=A mod 10) or (A mod 1000 div 100=A mod 100 div 10); Writeln('Число симметрич- ное — ',result; Readln; end.</pre> | <pre>Print "Число симметрич- ное — ";R end</pre> |

- Привести пример такого A , для которого программа дает неправильный результат.
- Исправить ошибку в программе так, чтобы она соответствовала поставленной задаче.
- Укажите, какая часть программы является лишней.

5. Требовалось написать программу, чтобы определить, пройдет ли через квадратное отверстие со стороной A кирпич с ребрами x , y , z (ребра кирпича пойдут вдоль сторон отверстия). Программист написал программу с ошибкой.

| ПАСКАЛЬ | БЕЙСИК |
|---|--|
| <pre>Var a,x,y,z:real; S:real; Begin Writeln('введите размеры кир- пича x, y, z'); Readln(x,y,z); Write('Введите размер отвер- ствия A='); Readln(a); S:=x*y; If (x<=a) and (y<=a) or (y<=a) and (z<=a) then writeln('пройдет') else writeln ('не пройдет'); Readln; End.</pre> | <pre>INPUT "Введите размеры кирпича x, y, z: ";x,y,z Print "Введите размер от- верствия A=";a S=x*y If (x<=a) and (y<=a) or (y<=a) and (z<=a) then print "пройдет" else print " не пройдет" End if End</pre> |

- Привести пример таких A, x, y, z , для которых программа дает неправильный результат.
- Исправить ошибку в программе так, чтобы она соответствовала поставленной задаче.
- Укажите, какая часть программы является лишней.

6. Требовалось написать программу, чтобы поменять местами во введенной из файла *input.dat* строки символы, стоящие на четных и нечетных местах. Результат следовало поместить в файл *output.dat*. Программист написал программу с ошибкой.

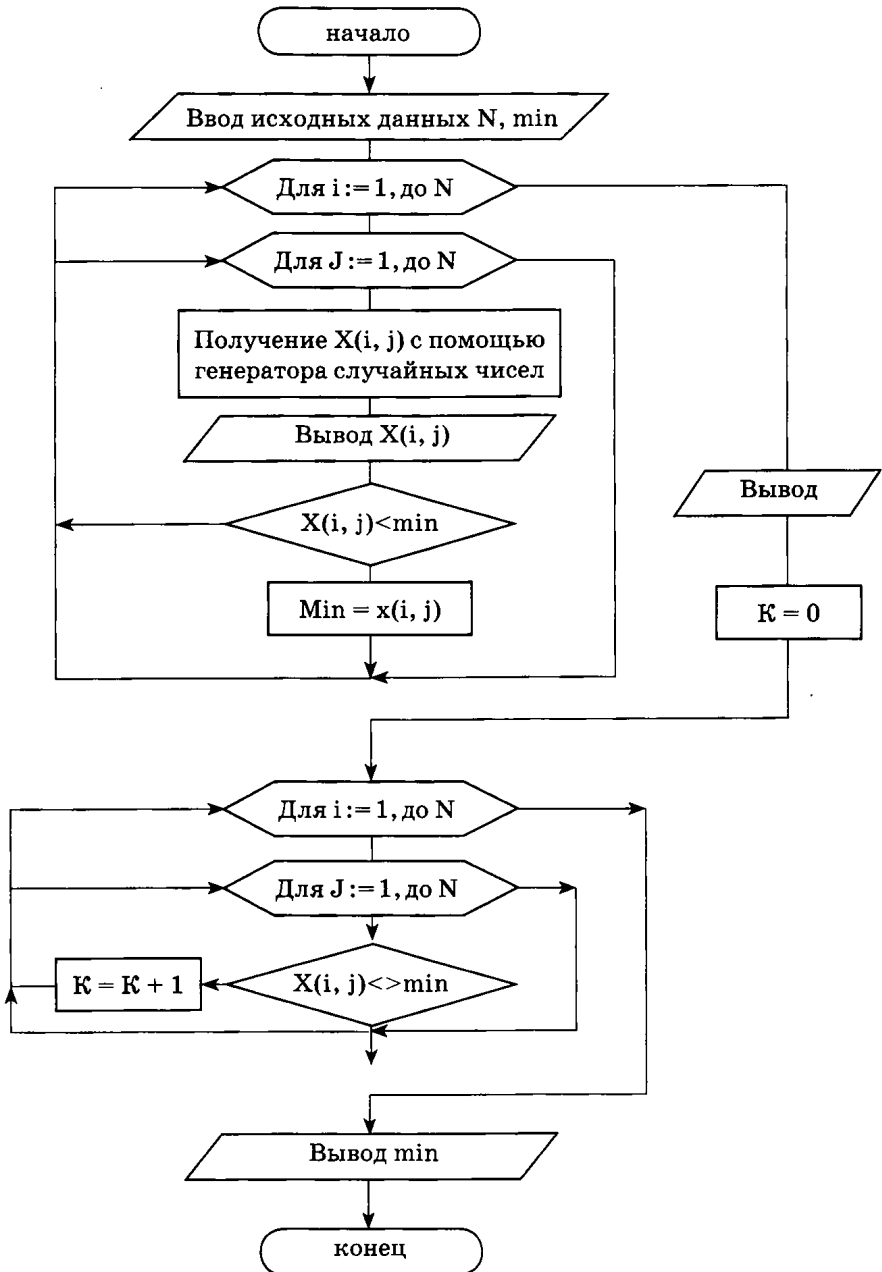
| ПАСКАЛЬ | БЕЙСИК |
|--|--|
| <pre> program file_my; uses crt; Var txt:text; Str:string; {для введенной строки} Ch:char; I,k,n:integer; Begin Assign(txt,'input.dat'); {связываем переменную txt с внешним файлом и открываем файл для чтения} Reset(txt); Readln(txt,str); {считываем из файла} Close(txt); {закрываем файл} N:=length(str); </pre> | <pre> DIM s\$(255) as string Defint I,k,n Defstr A,B,m,c 'или m\$ A:="input.dat" B:="output.dat" 'Открываем файл A для чтения как 1 OPEN a FOR INPUT AS #1 INPUT\$ #1,m 'm-набор символов. Читаем строку символов из файла CLOSE 'закрываем файл N=len(m) For i=1 to n S\$(i)= mid\$(m,I,1) Next i For i=1 to n </pre> |

| ПАСКАЛЬ | БЕЙСИК |
|--|--|
| <pre> For i:=1 to n do Begin Ch:=str[i*2]; str[i*2]:= str[i*2-1]; end; Assign(txt,'output.dat'); {связа- ваем переменную txt с внеш- ним файлом и открываем файл для записи} Rewrite(txt); Writeln(txt,str); {записываем в файл} Close(txt); {закрываем файл} Readln; End.</pre> | <pre> C=S\$(i*2) S\$(i*2)= S\$(i*2-1) Next i 'Открываем файл А для записи как 1 Open b for OUTPUT AS #1 Write #1,m CLOSE 'закрываем файл end</pre> |

- Привести пример такой строки, для которой программа дает неправильный результат.
- Исправьте программу так, чтобы в ней не выполнялось лишних операций.
- Исправить ошибку в программе так, чтобы она соответствовала поставленной задаче.

C2. Умение написать короткую (10–15 строк) простую программу обработки массива на языке программирования или записать алгоритм на естественном языке. *Применение знаний и умений в стандартной ситуации. Высокий уровень.* Примерное время выполнения задания — 30 мин.

1. Опишите на русском языке или на одном из языков программирования алгоритм определения количества элементов двумерного массива A размером $N \times N$ отличных от минимального.

Разбор решения задачи 1.**Создание блок-схемы алгоритма решения задачи**

Алгоритм решения задачи будет состоять из следующих блоков:

1. Описание двумерного массива любого типа.
2. Начало двух вложенных циклов по i и j .
3. Задание массива любым способом (ввод с клавиатуры с использованием случайной функции) с использованием вложенных циклов. Вывод элементов массива на экран. Так как одновременно с заданием массива можно выполнять поиск минимального элемента, то целесообразно это сделать. Поэтому до цикла следует задать начальное значение переменной для хранения минимального значения, например, $\text{min}=32767$ (максимальное целое число для типа Integer). Найти минимальный элемент.
4. Конец циклов.
5. Задать начальное значение счетчика количества элементов массива, отличных от минимального $K=0$.
6. Организовать сравнение всех элементов массива с найденным минимальным min , используя вложенные циклы. Если элемент не равен минимальному, то увеличить значение переменной-счетчика на 1.
7. Вывести значение количества элементов массива, отличных от минимального, на экран.

Пример правильной программы на языке Паскаль (Turbo Pascal 7.0)

```
Const n=15;    {задание константы n=15 — размер массива,
задается произвольно}
Type
  mass=array [1..n,1..n] of integer;
Var
  x:mass;      {описание двумерного массива x типа mass
размерности N×N}
  i,j,min,k:integer;
Begin
  Randomize;
  min:=32767;   {начальное значение переменной для
хранения минимума}
  For i:=1 to n do {цикл по I — по строкам}
```

```

begin
    For j:=1 to n do           {цикл по J — по столбцам}
    begin
        x[i,j]:=-10+random(40); {получение элементов
массива с помощью генератора случайных чисел}
        write(x[i,j]:5);       {вывод элементов массива в строку}
        if x[i,j] < min then min:=x[i,j]; {сравнение текущего
элемента с Min, если он меньше, то min получает его значение}
        end;
        writeln;
        end;
        writeln('min=',min);   {вывод значения минимального
                                элемента}
    k:=0; {обнуление счетчика элементов массива, отличных
от минимального}
    for i:=1 to n do           {вложенные циклы}
    for j:=1 to n do
        if x[i,j]<>min then k:=k+1; {если текущий элемент не ра-
вен min, то значение счетчика K увеличивается на 1}
        writeln('k=',k);       {вывод количества элементов, отличных
                                от минимального}
    readln;
end.

```

2. Опишите на русском языке или на одном из языков программирования алгоритм перестановки элементов целого одномерного массива A размером N в обратном порядке, не используя дополнительных массивов.
3. Опишите на русском языке или на одном из языков программирования алгоритм определения количества элементов двумерного массива A размером $N \times N$ отличных от максимального в каждой строке.
4. Массив содержит несколько предложений, разделенных точкой. Опишите на русском языке или одним из языков программирования алгоритм вывода на экран второго предложения.

5. Опишите на русском языке или на одном из языков программирования алгоритм определения номера строки массива A размером $N \times M$, содержащего одинаковые элементы. Известно, что хотя бы одна такая строка есть.
6. Опишите на русском языке или на одном из языков программирования алгоритм поиска элементов массива X размером N , кратных данному целому числу B . Определите их количество и сумму.
- С4.** Умение создавать собственные программы (30–50 строк) для решения задач средней сложности. *Уровень высокий. Применение знаний и умений в новой ситуации.* Примерное время выполнения задания – 60 мин.
1. На вход программе подаются сведения о студентах некоторого вуза. В первой строке сообщается количество студентов N , которое больше 20, но меньше или равно 100, каждая из следующих N строк имеет следующий формат: $\langle \text{Фамилия} \rangle \langle \text{Имя} \rangle \langle \text{Оценка1} \rangle \langle \text{Оценка2} \rangle \langle \text{Оценка3} \rangle \langle \text{Оценка4} \rangle \langle \text{Оценка5} \rangle$, где $\langle \text{Фамилия} \rangle$ — строка, состоящая не более чем из 20 символов, $\langle \text{Имя} \rangle$ — строка, состоящая не более чем из 15 символов, $\langle \text{Оценка1} \rangle \langle \text{Оценка2} \rangle \langle \text{Оценка3} \rangle \langle \text{Оценка4} \rangle \langle \text{Оценка5} \rangle$ — целое число от 1 до 5. $\langle \text{Фамилия} \rangle$ и $\langle \text{Имя} \rangle$, а также $\langle \text{Имя} \rangle$ и $\langle \text{Оценка1} \rangle \langle \text{Оценка2} \rangle \langle \text{Оценка3} \rangle \langle \text{Оценка4} \rangle \langle \text{Оценка5} \rangle$ разделены одним пробелом. Входные данные не упорядочены никаким способом.

Пример входных строк:

25

Федорова Ирина 5 4 5 5 4

Иванов Сергей 3 2 3 5 5

Светлов Андрей 5 5 5 5 5

Требуется написать программу, которая определит стипендию студента и будет выводить на экран фамилии

и имена студентов и их стипендию, которая начисляется следующим образом: имеющие все 5 — максимальная стипендия 6000 р. Студентам, получившим только 5 и 4 — 4000 р., получившим одну 3 — 2000 р. Остальные студенты стипендию не получают. Алфавитный порядок при выводе фамилий можно не соблюдать.

Пример выходных строк:

Федорова Ирина 4000 р.

Иванов Сергей 0 р.

Светлов Андрей 6000 р.

2. На вход программе подаются сведения об учениках 9-х и 11-х классов школы (не более 300 чел.) и о десяти предметах, сдаваемых по выбору. Названия предметов заданы в программе. В первой строке сообщается количество учащихся N, каждая из следующих N строк имеет формат: <Фамилия> <Имя> <Название класса> <Предмет>, <Балл>, где <Фамилия> — строка, состоящая не более чем из 20 символов, <Имя> — строка, состоящая не более чем из 12 символов, <Название класса> — 3 символа (например, 11а, 09б, 11б), <Предмет> — не более 12 символов, <Балл> — от 0 до 100. <Фамилия>, <Имя>, <Название класса>, <Предмет> и <Балл> разделены одним пробелом. Ввод строки заканчивается точкой.

Пример входных строк:

150

Кузнецов Сергей 09б история 45

Светлов Андрей 11а информатика 62

Кузнецов Сергей 09б химия 56

Требуется написать как можно более эффективную программу (укажите используемую версию языка программирования, например Borland Pascal 7.0), которая определит, какой предмет выбрали самое большое количество учащихся (при равенстве учащихся вывести все эти предметы), выведет на экран информацию о ко-

личестве учащихся, сдающих предметы из известного перечня, а также список учащихся с указанием фамилии, класса и балла. Вывести сообщение о наличии или отсутствии учащихся, набравших 100 баллов.

Пример выходных строк:

Есть учащиеся, получившие 100 баллов

Предмет Химия 12 учащихся

Иванов 11а 89

Петров 11в 56

...

Предмет Биология 12 учащихся

Степанов 11а 78

Антонов 11в 100

...

Предмет География 2 учащихся

Денисов 9в 56

Петров 9а 56

Самое большое количество учащихся — 12 — выбрали предметы:

Химия

Биология

3. На вход программе подаются сведения об осадках (миллиметры атмосферных осадков) в городах Московской области в течение трех месяцев. В первой строке сообщается количество записей (не более 99), каждая из следующих N строк имеет формат: <Город> <осадки1>, <осадки2>, <осадки3>, где <Город> — строка, состоящая не более чем из 20 символов, <осадки1>, <осадки2>, <осадки3> — целые числа. <Город> <осадки1> <осадки2> <осадки3> разделены одним пробелом.

Пример первых входных строк:

20

Подольск 50 55 68

Мытищи 45 50 55

Нарофоминск 55 60 65

Требуется написать как можно более эффективную программу (укажите используемую версию языка программирования, например Borland Pascal 7.0), которая будет выводить на экран информацию отдельно о городах с минимальным и максимальным уровнем осадков за трехмесячный период (если городов с одинаковыми минимальными (максимальными) значениями в году несколько, то вывести все эти города).

Примечание: в Московской области в месяце бывает в среднем 50—60 мм атмосферных осадков.

Пример выходных строк

Максимальные осадки в городе(дах)

Нарофоминск 180 мм

Минимальные осадки в городе(дах)

Мытищи 150 мм

Подольск 150 мм

4. На вход программе подаются сведения о пациентах поликлиники. В первой строке сообщается количество пациентов N , которых больше 10, но не более 300, во второй строке — текущий год; каждая из следующих N строк имеет следующий формат: <Фамилия> <Имя> <Отчество> <Номер истории болезни> <Пол> <Год рождения> <Код заболевания> <Наблюдение пациента>, где <Фамилия> — строка, состоящая не более чем из 20 символов, <Имя> — строка, состоящая не более чем из 15 символов, <Отчество> — строка, состоящая не более чем из 15 символов, <Номер истории болезни> — целое число, <Пол> — буква “м” или “ж”, <Год рождения> — целое число, <Код заболевания> — 4 символа, <Наблюдение пациента> — логическая переменная: 1 — пациент находится под наблюдением. 0 — нет. <Фамилия>, <Имя>, <Отчество>, <Номер истории болезни>, <Пол>, <Год рождения>, <Код заболевания>, <Наблюдение пациента> разделены одним пробелом.

Пример входной строки:

200

2009

Федорова Ирина Павловна 345 ж 1967 F450 0

Иванов Сергей Петрович 1029 м 1940 D500 1

Петров Сергей Иванович 4567 м 1993 S030 1

Волохова Зинаида Евгеньевна 345 ж 1992 H005 0

Требуется написать программу, которая будет выводить на экран списки пациентов по следующим категориям: девочки-подростки до 18 лет, юноши-подростки до 18 лет, пенсионеры-женщины (не моложе 55 лет), пенсионеры-мужчины (не моложе 60 лет). Вывести номер истории болезни, фамилию, инициалы, код заболевания и сведения о наблюдении. Известно, что в каждой из групп есть пациенты. Исходные данные вводятся корректно.

Результат работы программы может выглядеть так:**Девушки-подростки:**

| Номер ИБ | Фамилия И.О. | Код заболевания | Учет |
|----------|---------------|-----------------|------|
| 1234 | Волохова З.Е. | H005 | Нет |
| 235 | Сергеева А.А. | K900 | Да |

Юноши-подростки:

| | | | |
|-----|-------------|------|----|
| 123 | Петров С.И. | S030 | Да |
|-----|-------------|------|----|

Пенсионеры:

| | | | |
|------|---------------|------|-----|
| 2345 | Иванов С.П. | D500 | Да |
| 222 | Смирнова А.Т. | E006 | Нет |

5. На вход программе подаются сведения об учениках школы (не более 300 чел.) В первой строке сообщается количество учащихся N, каждая из следующих N строк имеет формат: <Фамилия> <Имя> <Название класса>, где <Фамилия> — строка, состоящая не бо-

лее чем из 20 символов, <Имя> — строка, состоящая не более чем из 12 символов, <Номер класса> — целое число (например, 11, 9, 10). <Фамилия>, <Имя>, <Номер класса> разделены одним пробелом. Примечание: фамилии и имена вводятся латинскими буквами.

Пример входных строк:

150

Кузнецов Сергей 9

Светлов Андрей 11

Кузнецов Сергей 9

Требуется написать как можно более эффективную программу (укажите используемую версию языка программирования, например Borland Pascal 7.0), которая определит, есть ли в школе однофамильцы. Вывести сообщение о наличии учащихся с одинаковыми фамилиями и их количество. Фамилии Иванов и Иванова, Репин и Репина, Одоевская и Одоевский считать различными.

Пример выходных строк:

Иванов 2 учащихся

Сидоров 3 учащихся

Петров 2 учащихся

Петрова 3 учащихся

6. На вход программе подаются сведения о спортсменах соревнования по бегу на длинные дистанции (не более 300 чел.) В первой строке сообщается количество спортсменов N, каждая из следующих N строк имеет формат: <Фамилия> <Имя> < Страна> <Время старта> <Время финиша>, где <Фамилия> — строка, состоящая не более чем из 20 символов, <Имя> — строка, состоящая не более чем из 12 символов, <Страна> — строка, состоящая не более чем из 20 символов, <Время старта> <Время финиша> имеют формат чч.мм, где чч — часы, мм — минуты, разделенные двоеточием (например, 14:12). <Фамилия>, <Имя>, <Класс>, <Время старта> и <Время финиша> разделены одним пробелом.

Пример входных строк:

150

Кузнецов Сергей USA 14:10 15:10

Светлов Андрей Russia 12:12 13:18

Петров Сергей Canada 10:04 10:55

Требуется написать как можно более эффективную программу (укажите используемую версию языка программирования, например Borland Pascal 7.0), которая выводит нумерованный список спортсменов в порядке увеличения времени, затраченного на дистанцию, и название страны победителя (если таких спортсменов несколько, также вывести названия их стран). Время вывести в минутах.

Пример выходных строк:

1. Петров Сергей 51 мин. Canada

2. Кузнецов Сергей 60 мин.

3. Светлов Андрей 66 мин.

Ответы на задания C1

2. Требовалось написать программу вычисления НОК (наименьшего общего кратного) двух целых чисел a и b , используя НОД (наибольший общий делитель)

$$\text{НОК} = (a \cdot b) / \text{НОД}$$

Программист написал программу с ошибкой.

| ПАСКАЛЬ | БЕЙСИК |
|---|--|
| <pre> Var a,b,m,n,nok:integer; Begin Readln(a); Readln(b); Readln(m); M:=a;n:=b; While m=n do begin If m>n then m:=m-n else n:=n-m; end; Nok:=(a*b) div m; Writeln ('nok=', nok); End.</pre> | <pre> Defint a,b,m,n,n Input "a=";a Input "b=";b Input "m=";m M=a;n=b While m = n If m>n then m=m-n else n=n-m End if Wend N=(a*b) \ m Print "nok=";n End</pre> |

Привести пример таких исходных данных a и b , для которых программа дает неправильный результат.

Ответ на первый вопрос: Для следующих данных результат будет ошибочен: при $A=81$, $b=48$, $\text{nok}=48$, а должно быть $\text{nok}=1296$. При $a=12$, $b=100$, $\text{nok}=100$, а должно быть $\text{nok}=300$.

Исправить ошибку в программе так, чтобы она соответствовала поставленной задаче.

Ответ на второй вопрос: **While m=n do** — ошибка, надо **While m<>n do**. В программе реализован алгоритм Евклида для нахождения НОД. Вычитание из большего числа меньшего числа надо производить до тех пор, пока они не равны, а при их равенстве надо закончить выполнение цикла **While**.

Укажите, какая часть программы является лишней.

Ответ: **Readln(m);** — лишняя часть программы, m — искомый НОД пары чисел, а не исходные данные.

| Правильная программа | |
|---|--|
| ПАСКАЛЬ | БЕЙСИК |
| <pre> Var a,b,m,n,nok:integer; Begin Write ('Введите целое число a ='); Readln(a); Write ('Введите целое число b ='); Readln(b); M:=a;n:=b; While m<>n do {OK} begin If m>n the m:=m-n else n:=n-m; end; Nok:=(a*b) div m; Writeln ('nok=', nok); End.</pre> | <pre> Defint a,b,m,n,k Input "целое a=";a Input "целое b=";b M=a;n=b While m <> n If m>n then m=m-n else n=n-m End if Wend N=(a*b) \ m Print "nok=";k End</pre> |

3. Требовалось написать программу определения симметричности пятизначного целого числа A , т.е. читается ли оно одинаково слева направо и справа налево. Программист написал программу с ошибкой.

| ПАСКАЛЬ | БЕЙСИК |
|---|--|
| <pre> Var A:integer; Result:boolean; Begin Writeln('Введите пятизначное целое число'); Readln(A); Result:=(a div 10000=A mod 10) and (A mod 100 div 10=A div 1000 mod 10); Writeln('Число симметрич- ное — ',result); Writeln('Число несимметрич- ное — ',not result); Readln; end.</pre> | <pre> Defint A Input "Введите пятизнач- ное число";A R=a \ 10000 = A mod 10) and (A mod 10000 \ 1000=A mod 1000 \ 10) Print "Число симметрич- ное — ";R Print "Число несимметрич- ное — "; not R end</pre> |

Привести пример такого A , для которого программа дает неправильный результат.

Ответ: $A = 53735$, т.к. $A:\text{integer}$, то $\max A = 32767$, $\min A = -32768$, т.е. для любого пятизначного числа больше максимального и меньше минимального будет неправильным результат.

Исправить ошибку в программе так, чтобы она соответствовала поставленной задаче.

Ответ: Переменную A надо описать как **Longint**.

Укажите, какая часть программы является лишней.

Ответ: `Writeln('Число несимметричное — ',not result);` — дублируется вывод результата.

| Правильная программа | |
|---|---|
| ПАСКАЛЬ | БЕЙСИК |
| <pre> Var A:Longint; Result:boolean; Begin Writeln('Введите пятизначное целое число'); Readln(A); Result:=(a div 10000=A mod 10) and (A mod 100 div 10=A div 1000 mod 10); Writeln('Число симметрич- ное — ',result; Readln; end.</pre> | <pre> DEFLNG A Input "Введите пятизнач- ное число";A R=(a \ 10000 = A mod 10) and (A mod 10000 \ 1000= A mod 1000 \ 10) Print "Число симметрич- ное — ";R end</pre> |

4. Требовалось написать программу определения симметричности **четырёхзначного** целого числа **A**, т.е. читается ли оно одинаково слева направо и справа налево. Программист написал программу с ошибкой.

| ПАСКАЛЬ | БЕЙСИК |
|--|--|
| <pre> Var A:integer; Result:boolean; Begin Writeln('Введите четырехзнач- ное целое число'); Readln(A); Result:=(a div 1000=A mod 10); Result:=(a div 1000=A mod 10) or (A mod 1000 div 100=A mod 100 div 10); Writeln('Число симметрич- ное — ',result; Readln; end.</pre> | <pre> Defint A Input "Введите четырех- значное число";A R=(a \ 10000=A mod 10) R=(a \ 10000=A mod 10) or (A mod 10000 \ 1000=A mod 1000 \ 10); Print "Число симметрич- ное — ";R end</pre> |

Привести пример такого A , для которого программа дает неправильный результат.

Ответ: $A = 1231$ — число симметричное, 3445 — тоже, т.е. когда хотя бы пара цифр были равны друг другу.

Исправить ошибку в программе так, чтобы она соответствовала поставленной задаче.

Ответ: Ошибка в том, что при сравнении пар цифр между выражениями отношения стоит логическая операция **OR**. Т.е. может быть равна только одна пара цифр и переменная **Result** принимает истинное значение. Нужно использовать логическую операцию **AND**, тогда логическая переменная **Result** будет принимать истинное значение только при равенстве обеих пар цифр.

Укажите, какая часть программы является лишней.

Ответ: $\text{Result} := (a \text{ div } 1000 = A \bmod 10)$; — лишняя часть, так как проверяется только одна пара цифр, в следующей команде проверяется сразу равенство двух пар цифр $\text{Result} := (a \text{ div } 1000 = A \bmod 10) \text{ and } (A \bmod 1000 \text{ div } 100 = A \bmod 100 \text{ div } 10)$;

| Правильная программа | |
|---|--|
| ПАСКАЛЬ | БЕЙСИК |
| <pre> Var A:integer; Result:boolean; Begin Writeln('Введите пятизначное целое число'); Readln(A); Result:=(a div 1000=A mod 10) and (A mod 1000 div 100=A mod 100 div 10); Writeln('Число симметрич- ное — ',result; Readln; end.</pre> | <pre> Defint A Input "Введите четырех- значное число";A R=(a \ 10000=A mod 10) and (A mod 10000 \ 1000=A mod 1000 \ 10); Print "Число симметрич- ное — ";R end</pre> |

5. Требовалось написать программу, чтобы определить, пройдет ли через квадратное отверстие со стороной A кирпич с ребрами x, y, z (ребра кирпича пойдут вдоль сторон отверстия). Программист написал программу с ошибкой.

| ПАСКАЛЬ | БЕЙСИК |
|---|--|
| <pre> Var a,x,y,z:real; S:real; Begin Writeln('введите размеры кирпича x, y, z'); Readln(x,y,z); Write('Введите размер отвер- ствия A='); Readln(a); S:=x*y; If (x<=a) and (y<=a) or (y<=a) and (z<=a) then writeln('пройдет') else writeln(' не пройдет'); Readln; End. </pre> | <pre> INPUT "Введите размеры кирпича x, y, z: ";x,y,z Print "Введите размер отвер- ствия A=";a S=x*y If (x<=a) and (y<=a) or (y<=a) and (z<=a) then print "прой- дет" else print " не пройдет" End if End </pre> |

Привести пример таких A, x, y, z , для которых программа дает неправильный результат.

Ответ: $A=15, x=10, y=20, z=15$, так как одна из сторон кирпича больше стороны отверстия: $y > a$.

Исправить ошибку в программе так, чтобы она соответствовала поставленной задаче.

Ответ: в программе не проверяется условие $(x \leq a) \text{ and } (z \leq a)$ ($10 \leq 15 \text{ and } 15 \leq 15$), которое дает ИСТИНУ, и кирпич может пройти через отверстие, в то время как проверка условий ($10 \leq 15 \text{ and } 20 \leq 15$) дает FALSE и ($20 \leq 15 \text{ and } 15 \leq 15$) тоже дает FALSE, т.е. кирпич пройти не сможет. В программу надо добавить еще одно условие, так как кирпич может проходить через отверстие тремя способами (сторонами $xу, хz, уz$).

Укажите, какая часть программы является лишней.

Ответ: $S=x*y$; — не требуется вычислять площадь ни одной из поверхностей кирпича.

| Правильная программа | |
|--|---|
| ПАСКАЛЬ | БЕЙСИК |
| <pre> Var a,x,y,z:real; Begin Writeln('введите размеры кирпича x, y, z'); Readln(x,y,z); Write('введите размер отверстия A='); Readln(a); If (x<=a) and (y<=a) or (y<=a) and (z<=a) or (x<=a) and (z<=a) then writeln('пройдет') else writeln(' не пройдет'); Readln; End.</pre> | <pre> INPUT "Введите размеры кирпича x, y, z: ";x,y,z Print "Введите размер отверстия A=";a If (x<=a) and (y<=a) or (y<=a) and (z<=a) or (x<=a) and (z<=a) then print "пройдет" else print " не пройдет" End if End</pre> |

6. Требовалось написать программу, чтобы поменять местами во введенной из файла **input.dat** строке символы, стоящие на четных и нечетных местах. Результат следовало поместить в файл **output.dat**. Программист написал программу с ошибкой.

| ПАСКАЛЬ | БЕЙСИК |
|---|--|
| <pre> program file_my; uses crt; Var txt:text; Str:string; {для введенной строки} Ch:char; I,k,n:integer;</pre> | <pre> DIM s\$(255) as string Defint I,k,n Defstr A,B,m,c 'или m\$ A:="input.dat" B:="output.dat" 'Открываем файл A для чтения как 1 OPEN a FOR INPUT AS #1</pre> |

Окончание табл.

| ПАСКАЛЬ | БЕЙСИК |
|--|---|
| <pre> Begin Assign(txt,'input.dat'); {связываем переменную txt с внешним файлом и открыва- ем файл для чтения} Reset(txt); Readln(txt,str); {считываем из файла} Close(txt); {закрываем файл} N:=length(str); For i:=1 to n do Begin Ch:=str[i*2]; str[i*2]:= str[i*2-1]; end; Assign(txt,'output.dat'); {связываем переменную txt с внешним файлом и открыва- ем файл для записи} Rewrite(txt); Writeln(txt,str); {записыва- ем в файл} Close(txt); {закрываем файл} Readln; End.</pre> | <pre> INPUT\$ #1,m 'm-набор символов. Читает строку сим- волов из файла CLOSE 'закрываем файл N=len(m) For i=1 to n S\$(i)= mid\$(m,I,1) Next i For i=1 to n C=S\$(i*2) S\$(i*2)= S\$(i*2-1) Next i 'Открываем файл А для записи как 1 Open b for OUTPUT AS #1 Write #1,m CLOSE 'закрываем файл end</pre> |

Привести пример такой строки, для которой програм-
ма дает неправильный результат.

Ответ: Неправильно для любых строк, где **неодинако-
вые символы на четных и нечетных местах**. Символы
на нечетном и четном местах не меняются местами, а
символ на нечетном месте удваивается.

Дано: травка зеленеет

Получаем: ттаакк еееееетт

Исправьте программу так, чтобы в ней не выполня-
лось лишних операций.

Ответ: В программе осуществляются лишние перестановки «пустых» символов, так как символы строки сравниваются парами, то можно цикл сократить в 2 раза. **For i:=1 to length(str) div 2 do**

Исправить ошибку в программе так, чтобы она соответствовала поставленной задаче.

Ответ: пропущена команда **str[i*2-1]:=ch;** в части программы, когда символы меняются местами (используется вспомогательная переменная ch).

Ch:=str[i*2]; {запоминается символ на четном месте}
str[i*2]:= str[i*2-1]; {на место символа с четным номером помещается символ с нечетным номером}

str[i*2-1]:=ch; {на место символа с нечетным номером помещается символ с четным номером} — эта команда была пропущена.

| Правильная программа | |
|--|--|
| ПАСКАЛЬ | БЕЙСИК |
| <pre> program file_my; uses crt; Var txt:text; Str:string; {для введенной строки} Ch:char; I,k,n:integer; Begin Assign(txt,'input.dat'); {связываем переменную txt с внешним файлом и открываем файл для чтения} Reset(txt); Readln(txt,str); {считываем из файла} Close(txt); {закрываем файл} </pre> | <pre> DIM s\$(255) as string Defint I,k,n Defstr A,B,m,c 'или m\$ A:="input.dat" B:="output.dat" 'Открываем файл A для чтения как 1 OPEN a FOR INPUT AS #1 INPUT\$ #1,m 'm-набор символов. Читает строку символов из файла CLOSE 'закрываем файл N=len(m) For i=1 to n S\$(i)= mid\$(m,I,1) </pre> |

Продолжение табл.

| ПАСКАЛЬ | БЕЙСИК |
|--|--|
| <pre> For i:=1 to length(str) div 2 do {два варианта сокращения программы} N:=length(str); For i:=1 to n div 2 do Begin Ch:=str[i*2]; str[i*2]:= str[i*2-1]; str[i*2-1]:=ch; {OK} end; Assign(txt,'output.dat'); {связа- ваем переменную txt с внешним файлом и открываем файл для записи} Rewrite(txt); Writeln(txt,str); {записываем в файл} Close(txt); {закрываем файл} Readln; End.</pre> | <pre> Next i For i=1 to n\2 C=S\$(i*2) S\$(i*2)= S\$(i*2-1) S\$ (i*2-1)=c Next i 'Открываем файл А для записи как 1 Open b for OUTPUT AS #1 Write #1,m CLOSE 'закрываем файл end</pre> |

Ответы на задания С2

7. Опишите на русском языке или на одном из языков программирования алгоритм перестановки элементов целого одномерного массива A размером N в обратном порядке, не используя дополнительных массивов.

Алгоритм решения задачи будет состоять из следующих блоков:

1. Описание одномерного массива любого типа.
2. Задание массива любым способом (ввод с клавиатуры, с использованием случайной функции) с использованием цикла. Вывод элементов массива на экран.
3. Организация работы цикла, переменная имеет начальное значение 1, конечное значение — целое от деления нацело

количества элементов в массиве, так как количество перестановок в 2 раза меньше количества элементов.

4. В теле цикла меняем местами симметричные элементы с концов массива, используя вспомогательную переменную.
5. Вывод массива на экран (в нем элементы следуют в обратном порядке по сравнению с исходным массивом).

Пример правильной программы на языке Паскаль (Turbo Pascal 7.0)

```

Const n=15;
Type
    mass=array[1..n] of integer;
Var
    x:mass;
    i,y:integer; {буферная переменная y}
Begin
    Randomize;
    For i:=1 to n do
        begin
            x[i]:=-10+random(40); {генерирование одномерного
массива целого типа}
            write(x[i]:5);
        end;
    writeln;
    for i:=1 to n div 2 do {заголовок цикла, n div 2 , количество
перестановок в 2 раза меньше количества элементов,
div — деление нацело. При нечетном количестве элементов
средний элемент переставлять не с чем}
        begin
            y:=x[i]; {запоминание в переменной y текущего элемента
массива}
            x[i]:=x[n-i+1]; {текущий элемент получает значение симме-
тричного элемента с конца массива}
            x[n-i+1]:=y;    {элемент с конца массива получает значение
симметричного элемента с начала массива}
        end; {конец цикла}
    for i:=1 to n do

```

Окончание таблицы

```
write(x[i]:5); {вывод массива X с переставленными  
элементами}  
writeln;  
readln;  
end.
```

Примечание: То же — для массива любого типа, изменить только описание типа

3. Опишите на русском языке или на одном из языков программирования алгоритм определения количества элементов двумерного массива A размером $N \times N$, отличных от максимального в каждой строке.

Алгоритм решения задачи будет состоять из следующих блоков:

1. Описание исходного двумерного массива $X[n,n]$ любого типа. Описание одномерного массива $X1[n]$ для хранения максимальных элементов из каждой строки матрицы.
2. Задание массива X любым способом (ввод с клавиатуры, с использованием случайной функции) с использованием вложенных циклов. Вывод элементов массива на экран. Задание во внешнем цикле элементам массива $X1$ начальных значений -32768 . Во внутреннем цикле надо сравнивать текущий элемент строки массива с элементом массива $X1$. В том случае когда элемент исходного массива больше, то текущий элемент $X1$ заменяется.
3. Вывод массива $X1$ — максимальных элементов строк исходного массива X .
4. Задать начальное значение счетчика количества элементов массива, отличных от максимального $K=0$.
5. Организовать сравнение элементов строк массива с найденными максимальными значениями, используя вложенные циклы. Если элемент не равен максимальному, то увеличить значение переменной счетчика на 1.
6. Вывести значение количества элементов массива, отличных от максимального в каждой строке, на экран.

Пример правильной программы на языке Паскаль (Turbo Pascal 7.0)

```

Const n=15;
Type
    mass=array[1..n,1..n] of integer;
    mass1= array[1..n] of integer;
Var
    x:mass;x1:mass1; {описание двумерного X и одномер-
ного X1 массивов}
    i,j,k:integer; {описание целых переменных}
Begin
    Randomize;
    For i:=1 to n do
        begin
            x1[i]:=-32768; {задание начальных значений эле-
ментам массива X1. -32768 — минимальное целое число для
переменных типа Integer }
            For j:=1 to n do
                begin
                    x[i,j]:=-10+random(40); {получение исходного
массива с помощью генератора случайных чисел}
                    write(x[i,j]:5); {вывод исходного массива}
                    if x[i,j] > x1[i] then x1[i]:=x[i,j]; {поиск макси-
мальных элементов в строках }
                end;
            writeln;
        end;
    for i:=1 to n do
        writeln(i:5,' max=',x1[i]); {вывод максимальных элементов в
строках}
    k:=0;
    for i:=1 to n do
        for j:=1 to n do
            if x[i,j]<>x1[i] then k:=k+1; {если текущий элемент стро-
ки не равен X1, то значение счетчика K увеличивается на 1}
            writeln('k=',k); {вывод количества элементов, отличных
от максимального}
        readln;
    end.

```

4. Массив содержит несколько предложений, разделенных точкой. Опишите на русском языке или на одном из языков программирования алгоритм вывода на экран второго предложения.

Алгоритм решения задачи будет состоять из следующих блоков:

1. Описание переменных: символьной, целой, логической.
2. Ввод символьной строки.
3. Логическая переменная F получает значение FALSE.
4. Переменная K получает значение 1.
5. Организация цикла для поиска первой точки. Первая точка — признак конца первого предложения и начало второго, которое нужно вывести. Если найдена точка и $F=TRUE$, то K увеличивается на 1 или если найдена точка и $K=1$, то K увеличивается на 1 и становится равной 2. Если символ не точка и $K=2$, то только тогда начинается вывод второго предложения до точки (точка не выводится), переменная F получает значение TRUE.
Когда дошли до конца 2-го предложения и считали точку, то переменная K увеличивается на 1 и становится равной 3. Если $K>2$, то выводится символ точка, т.е. 2-е предложение существует. Конец цикла.
6. Если было введено только одно предложение ($K=2$ и $F=FALSE$), то вывести сообщение, что 2-го предложения нет.

Пример правильной программы на языке Паскаль (Turbo Pascal 7.0)

```

Var a:string; {описание символьной переменной}
I,k:integer;  {описание целых переменных}
f:boolean;    {описание логической переменной}
Begin
Write('a=');readln(a); {ввод предложений}
f:=false;           {логической переменной присвоено значение FALSE}
K:=1;              {признак первой точки k=1}
For i:=1 to length(a) do {организация цикла, конечное значение переменной цикла I равно количеству символов во введенных предложениях, используется функция length}

```

```

begin
if (a[i]='.') and f then k:=k+1; {увеличение К на единицу, если
выведено 2-е предложение. F=TRUE — признак вывода 2-го
предложения}
If (a[i]='.') and (k=1) then k:=k+1; {увеличение К на едини-
цу, если найдена точка после первого предложения (К=1), т.е.
К будет равно 2}
If (a[i]<>'.') and (k=2) then {если символ не точка, а К=2, т.е.
идут символы второго предложения и их надо выводить на
экран}
begin
write(a[i]);f:=true; {вывод символов второго предложения
на экран, переменная F получает значение TRUE}
end;
If (k>2) then
begin
writeln('.'); readln;exit; {выводим точку после второго
предложения}
end;
End;
if (k=2) and (not f) then writeln('2-го предложения нет');
{выводим сообщение, что 2-го предложения в тексте нет}
Readln;
End.

```

5. Опишите на русском языке или на одном из языков программирования алгоритм определения номера строки массива A размером $N \times M$, содержащего одинаковые элементы. Известно, что хотя бы одна такая строка есть.

Алгоритм решения задачи будет состоять из следующих блоков:

1. Описание массива и переменных.
2. Ввод массива либо с клавиатуры либо из файла.
3. Вывод элементов массива на экран.
4. Организовать два вложенных цикла для сравнения элементов в строке.
Внешний цикл работает столько раз, сколько строк в массиве.
Переменной K присваиваем значение 1.

Внутренний цикл на 1 меньше, чем количество столбцов, так как элементы в строке надо сравнивать попарно: 1 и 2, 2 и 3, ..., а последний элемент в строке сравнивать не надо. Если сравниваемые элементы равны, переменная *K* увеличивается на 1. После работы внутреннего цикла осуществляется проверка равенства *K* и количества элементов в строке. Если они равны, то выводится номер строки с равными элементами.

Пример правильной программы на языке Паскаль (Turbo Pascal 7.0)

```

Uses crt;
Const n=7; m=5;
Type
    mass=array[1..n,1..m] of integer;
Var
    x:mass;
    i,j,k:integer;
    txt: file of integer;
Begin
    {Ввод массива с клавиатуры*}
    For i:=1 to n do
    For j:=1 to m do
    begin
        writeln('Введите элементы массива по строкам');
        readln(x[i,j]);
    end;
    {Ввод массива из input.dat - внешнего файла*}
    Assign(txt,'input.dat'); {связываем переменную txt с внеш-
        ним файлом и открываем файл для чтения}
    Reset(txt);
    For i:=1 to n do
    For j:=1 to m do
    Read(txt,x[i,j]); {считываем данные из файла}
    Close(txt); {закрываем файл}
    {Вывод элементов массива на экран}
    For i:=1 to n do
    For j:=1 to m do
    write(x[i,j]:5);
    writeln;
    For i:=1 to n do

```

```
begin
K:=1;           {Кол-во равных элементов в строке}
For j:=1 to m-1 do {Сравниваем элементы попарно}
begin
if (x[I,j] = x[I,j+1]) then k:=k+1; {если элементы равны, то K
увеличивается на 1}
end;
if k=m then writeln('строка с равными элементами=',i); {вы-
вод номера строки с равными элементами}
end;
readln;
end.
```

* — массив в программе вводится только одним из приведенных способов.

6. Опишите на русском языке или на одном из языков программирования алгоритм поиска элементов массива X размером N , кратных данному целому числу B . Определите их количество и сумму.

Алгоритм решения задачи будет состоять из следующих блоков:

1. Описание одномерного массива целого типа и целых переменных.
2. Ввод с клавиатуры числа B .
3. Задание массива любым способом (ввод с клавиатуры, с использованием случайной функции) с использованием цикла. Вывод элементов массива на экран.
4. Обнуление переменных-счетчиков количества элементов кратных B и их суммы.
5. Организация цикла для анализа элементов массива.
6. Текущий элемент массива будет кратен числу B , если остаток от деления элемента массива нацело на число B будет равен нулю. Проверяем это условие для всех элементов массива.
Если условие выполняется, то переменная-счетчик увеличивается на 1, а к переменной накопления суммы прибавляется проверяемый элемент массива.
7. Вывод количества элементов, кратных B .
8. Вывод суммы элементов, кратных B .

Пример правильной программы на языке Паскаль (Turbo Pascal 7.0)

```

Const n=15;
Type
    mass=array[1..n] of integer;
Var
    x:mass;           {описание массива}
    i,j,min,b,k,s:integer;    {описание переменных цело-
го типа}
Begin
Randomize;
Writeln('введите целое число b'); {Ввод числа B}
Readln(b);
For i:=1 to n do
    begin
        x[i]:=-10+random(40); {генерирование одномерного
массива целого типа}
        write(x[i]:5);        {вывод элементов массива на
экран}
    end;
writeln;
K:=0;s:=0;    {обнуление переменных-счетчиков: K — коли-
чество, s — сумма}
for i:=1 to n do    {Заголовок цикла}
begin
IF x[i] mod b = 0 then begin {сравнение остатка от деления
элемента массива нацело на число B с нулем}
k:=k+1;           {увеличение счетчика количества эле-
ментов, удовлетворяющих условию, на 1}
s:=s+x[i];        {суммирование элементов, удовлетворяющих
условию}
end; {конец IF}
end; {конец цикла}
writeln;
writeln('kol-vo=',k);    {вывод количества элементов,
кратных B}
writeln('summa=',s);    {вывод суммы элементов, крат-
ных B}
readln;
end.

```

Ответы раздела С4

1. На вход программе подаются сведения о студентах некоторого вуза. В первой строке сообщается количество студентов N , которое больше 20, но меньше или равно 100, каждая из следующих N строк имеет следующий формат: <Фамилия> <Имя> <Оценка1> <Оценка2> <Оценка3> <Оценка4> <Оценка5>, где <Фамилия> — строка, состоящая не более чем из 20 символов, <Имя> — строка, состоящая не более чем из 15 символов, <Оценка1> <Оценка2> <Оценка3> <Оценка4> <Оценка5> — целое число от 1 до 5. <Фамилия> и <Имя>, а также <Имя> и <Оценка1> <Оценка2> <Оценка3> <Оценка4> <Оценка5> разделены одним пробелом. Входные данные не упорядочены никаким способом.

Пример входных строк:

25

Федорова Ирина 5 4 5 5 4

Иванов Сергей 3 2 3 5 5

Светлов Андрей 5 5 5 5 5

Требуется написать программу, которая определит стипендию студента и будет выводить на экран фамилии и имена студентов и их стипендию, которая начисляется следующим образом: имеющие все 5 — максимальная стипендия 6000 р. Студентам, получившим только 5 и 4 — 4000 р., получившим одну 3 — 2000 р. Остальные студенты стипендию не получают. Алфавитный порядок при выводе фамилий можно не соблюдать.

Пример выходных строк:

Федорова Ирина 4000 р.

Иванов Сергей 0 р.

Светлов Андрей 6000 р.

Разбор задачи 1.

Алгоритм решения задачи.

1. Задать значения стипендий для отличников, хорошистов и студентов с одной тройкой (stip1=6000; stip2=4000;stip3=2000;)

2. Для представления единой структурой совокупности неоднородных данных (фамилия — символы, оценки, стипендия — целые числа) будем использовать записи (record). Для ссылки на поле записи используется форма: <имя переменной типа запись>.<имя поля>.
 p:массив[1..100] = запись
 имя: символьный тип;
 оценки:массив[1..5] целый тип; {5 оценок}
 стипендия: целый тип;
3. Ввод количества студентов n.
4. Начало цикла с параметром i от 1 до n для чтения полей записей.
 - a. Переменной fam присвоить пустой символ.
 - b. Повторять {цикл с постусловием для считывания фамилии}
 - c. Ввод символа.
 - d. Накапливание символов фамилии в переменной fam.
 - e. Пока символ не будет пробелом.
 - f. Повторять {цикл с постусловием для считывания имени}.
 - g. Ввод символа.
 - h. Накапливание символов имени в переменной fam.
 - i. Пока символ не будет пробелом.
 - j. Обнуление переменной K3 для подсчета оценок 3, 2, 1, 0 и суммы оценок ms для данного студента.
 - k. Начало цикла по j от 1 до 5 для считывания оценок
 - i. Считывание текущей оценки.
 - ii. Если оценка 2 или 1 или 0, то K3 увеличивается на 2 (так как 1 тройка дает минимальную стипендию).
 - iii. Если оценка равна 3, то K3 увеличивается на 1.
 - iv. Накапливание оценок студента в переменной ms.
 - l. Если сумма оценок равна 25, то в поле записи <Стипендия> вводится максимальная стипендия.

- м. Если сумма оценок меньше 25 и кол-во троек равно нулю, то в поле записи стипендия вводится стипендия для хорошистов.
 - п. Если сумма оценок меньше 25 и кол-во троек равно единице, то в поле записи стипендия вводится минимальная стипендия.
 - о. Если сумма оценок меньше 25 и кол-во троек больше единицы, то в поле записи стипендия вводится ноль — отсутствие стипендии.
 - р. Конец цикла по i — считаны все записи и заполнено поле Стипендия.
5. Начало цикла с параметром по i от 1 до n .
- а. Вывод фамилии студентов и их стипендии.
 - б. Конец цикла.
6. Конец программы.

Ответ:

Пример правильной и эффективной программы на языке Паскаль (Borland Turbo Pascal 7.0)

```
Const stip1=6000; stip2=4000;stip3=2000;
var p:array[1..100] of record
    name:string[32];
    m:array[1..5] of integer;
    stip:integer;
end;
c:char;
i,n,j,ms,k3:integer;
m: array[1..5] of integer;
begin
    readln(n);
    for i:=1 to n do
        begin
            p[i].name:='';
            repeat
                read(c);
                p[i].name:=p[i].name+c
            until c=''; {считана фамилия}
            repeat
                read(c);
```

```

    p[i].name:=p[i].name+c
    until c=' '; {считано имя}
k3:=0; ms:=0; {обнулено кол-во троек, 2, 1, 0 и сумма оценок}
for j:=1 to 5 do begin
    read(p[i].m[j]);    {считывание оценок}
    if (p[i].m[j]=2) or (p[i].m[j]=1) or (p[i].m[j]=0) then k3:=k3+2;
    if p[i].m[j]=3 then k3:=k3+1;
    ms:=ms+p[i].m[j];
end; {цикла j}
If ms=25 then p[i].stip:=stip1;
If (ms<25) and (k3=0) then p[i].stip:=stip2;
if (ms<25) and (k3=1) then p[i].stip:=stip3;
if (ms<25) and (k3>1) then p[i].stip:=0;
End;
for i:=1 to n do {вывод студентов и стипендии}
    with p[i] do begin
        writeln(name,stip:8,' p. ');
    end;
Writeln;
Readln; readln;
End.

```

2. На вход программе подаются сведения об учениках 9-х и 11-х классов школы (не более 300 чел.) и о десяти предметах, сдаваемых по выбору. Названия предметов заданы в программе. В первой строке сообщается количество учащихся N, каждая из следующих N строк имеет формат: <Фамилия> <Имя> <Название класса> <Предмет>, <Балл>, где <Фамилия> — строка, состоящая не более чем из 20 символов, <Имя> — строка, состоящая не более чем из 12 символов, <Название класса> — 3 символа (например, 11а, 09б, 11б), <Предмет> — не более 12 символов, <Балл> — от 0 до 100, не более 3 символов. <Фамилия>, <Имя>, <Название класса>, <Предмет> и <Балл> разделены одним пробелом. Ввод строки заканчивается точкой.

Пример входных строк:

150

Кузнецов Сергей 096 история. 45

Светлов Андрей 11а информатика. 62

Кузнецов Сергей 096 химия. 56

Требуется написать как можно более эффективную программу (укажите используемую версию языка программирования, например, Borland Pascal 7.0), которая определит, какой предмет выбрали самое большое количество учащихся (при равенстве учащихся вывести все эти предметы), выведет на экран информацию о количестве учащихся, сдающих предметы из известного перечня, а также список учащихся с указанием фамилии, класса и балла. Вывести сообщение о наличии или отсутствии учащихся, набравших 100 баллов.

Пример выходных строк:

Есть учащиеся, получившие 100 баллов

Предмет Химия 12 учащихся

Иванов 11а 89

Петров 11в 56

...

Предмет Биология 12 учащихся

Степанов 11а 78

Антонов 11в 100

...

Предмет География 2 учащихся

Денисов 9в 56

Петров 9а 56

Самое большое количество учащихся — 12 — выбрали предметы:

Химия

Биология

Ответ:

Пример правильной и эффективной программы на языке Паскаль (Borland Turbo Pascal 7.0)

```
const d:array[1..10] of string [3] = ('mat', 'rus', 'lit', 'bio',
'geo', 'fiz', 'him', 'ist', 'ikt', 'lan'); {названия предметов по-
английски}
const m=10; {количество предметов}
{const d:array[1..10] of string [12] = ('математика', 'русский',
'литература', 'биология', 'география', 'физика', 'химия',
'история', 'информатика', 'ин.яз'); {названия предметов по-
русски. Не все версии Паскаля поддерживают кириллицу,
поэтому при отладке программы на компьютере используйте
только латиницу.}}
```

```
var p:array[1..300] of record
    famm:string[20];
    name: string[12];
    klass: string[3];
    subs: string[3];
    bal:integer;
end;
c:char;
I,j,n,kmax:integer;
Flag:Boolean;
K:array[1..10] of integer;
begin
for i:=1 to m do
k[i]:=0;
readln(n); {количество уч-ся}
for i:=1 to n do
begin
p[i].famm:= "";
repeat
read(c);
p[i].famm:=p[i].famm+c
until c=' '; {считана фамилия}
p[i].name:= "";
repeat
read(c);
p[i].name:=p[i].name+c
until c=' '; {считано имя}
```

```

p[i].klass:='';
repeat
  read(c);  p[i].klass:=p[i].klass+c
until c=' '; {считан класс}
p[i].subs:='';
repeat
  read(c);
  p[i].subs:=p[i].subs+c
until c='.'; {считано названия предмета}
flag:=false; {признак отсутствия уч-ся со 100 баллами}
read(p[i].bal);
if p[i].bal=100 then flag:=true;
for j:=1 to m do
  if p[i].subs=d[j] then
    k[j]:=k[j]+1; {количество уч-ся, сдающих j-тый предмет}
  end;           {конец цикла i}
kmax:=0;
for i:=1 to m do begin
  writeln('Предмет ',d[i],', ',k[i],', уч-ся'); {вывод предмета и
  кол-ва уч-ся }
  if k[i] > kmax then kmax:=k[i]; {определение max кол-ва уч-
  ся}
  for j:=1 to n do begin
    if p[j].subs=d[i] then writeln(p[j].famm,', ', p[j].klass, ', ',p[j].
    bal); {вывод учащихся, сдающих данный предмет и их балл}
  end;
  writeln;
end;
writeln('Самое большое количество учащихся -', kmax, '
выбрали предметы:');
for i:=1 to m do
  if k[i]=kmax then writeln(d[i]);
if flag then writeln('Есть уч-ся, получившие 100 баллов ') else
writeln('Никто из уч-ся не получил 100 баллов ');
readln;
end.

```

3. На вход программе подаются сведения об осадках (миллиметры атмосферных осадков) в городах Московской области в течение трех месяцев. В первой строке сообщается количество записей (не более 99), каждая из следующих N строк имеет формат: <Город> <осадки1>, <осадки2> , <осадки3>, где <Город> — строка, состоящая не более чем из 20 символов; <осадки1>, <осадки2> , <осадки3> — целые числа. <Город> <осадки1> <осадки2> <осадки3> разделены одним пробелом.

Пример первых входных строк:

20

Подольск 50 55 68

Мытищи 45 50 55

Нарофоминск 55 60 65

Требуется написать как можно более эффективную программу (укажите используемую версию языка программирования, например, Borland Pascal 7.0), которая будет выводить на экран информацию отдельно о городах с минимальным и максимальным уровнем осадков за трехмесячный период (если городов с одинаковыми минимальными (максимальными) значениями в году несколько, то вывести все эти города).

Примечание: в Московской области в месяце бывает в среднем 50 — 60 мм атмосферных осадков.

Пример выходных строк:

Максимальные осадки в городе/дах

Нарофоминск 180 мм

Минимальные осадки в городе/дах

Мытищи 150 мм

Подольск 150 мм

Ответ:

Пример правильной и эффективной программы на языке Паскаль (Borland Turbo Pascal 7.0)

```

var rain: array[1..99] of record
    town:string;           {город}
    nc1:integer;           {осадки 1 месяца в городе}
    nc2:integer;           {осадки 2 месяца в городе}
    nc3:integer;           {осадки 3 месяца в городе}
end;
sum:array[1..99] of integer; {массив для хранения суммар-
ных осадков городов}
c:char;
i, k, n,min,max: integer;
begin
    readln(n); {считывается количество записей}
    max:=0;
    min:=1000;
    for i:=1 to n do
        sum[i]:=0; {обнуление суммарных осадков города за 3 м-ца}
        for i:=1 to N do
            with rain[i] do
                begin
                    town:="";
                    repeat
                        read(c);
                        town:=town+c
                    until c=' '; {считано название города}
                    read(nc1); {считаны осадки 1 месяца}
                    read(nc2); {считаны осадки 2 месяца}
                    read(nc3); {считаны осадки 3 месяца}
                    sum[i]:=nc1+nc2+nc3;
                    if max<sum[i] then max:=sum[i];
                    if min> sum[i] then min:=sum[i];
                end;
            {цикл для вывода городов с max и min осадками по годам}
        writeln('Минимальные осадки в городе/дах');
        for i:=1 to n do
            with rain[i] do
                begin
                    if sum[i]=min then
                        writeln(town,' ',min,' мм');
                end;
            end;

```

Окончание программы

```

writeln('Максимальные осадки в городе/дах');
for i:=1 to n do
  with rain[i] do
    begin
      if sum[i]=max then
        writeln(town,' ',max,' мм');
      end;
    readln;
  end.

```

4. На вход программе подаются сведения о пациентах поликлиники. В первой строке сообщается количество пациентов N , которых больше 10, но не более 300, во второй строке — текущий год, каждая из следующих N строк имеет следующий формат: <Фамилия> <Имя> <Отчество> <Номер истории болезни> <Пол> <Год рождения> <Код заболевания> <Наблюдение пациента>, где <Фамилия> — строка, состоящая не более чем из 20 символов, <Имя> — строка, состоящая не более чем из 15 символов, <Отчество> — строка, состоящая не более чем из 15 символов, <Номер истории болезни> — целое число, <Пол> — буква “м” или “ж”, <Год рождения> — целое число, <Код заболевания> — 4 символа, <Наблюдение пациента> — логическая переменная: 1 — пациент находится под наблюдением. 0 — нет. <Фамилия>, <Имя>, <Отчество>, <Номер истории болезни>, <Пол>, <Год рождения>, <Код заболевания>, <Наблюдение пациента> разделены одним пробелом.

Пример входной строки:

200

2009

| | | | | | |
|-----------------------------|------|---|------|------|---|
| Федорова Ирина Павловна | 345 | ж | 1967 | F450 | 0 |
| Иванов Сергей Петрович | 1029 | м | 1940 | D500 | 1 |
| Петров Сергей Иванович | 4567 | м | 1993 | S030 | 1 |
| Волохова Зинаида Евгеньевна | 345 | ж | 1992 | H005 | 0 |

Требуется написать программу, которая будет выводить на экран списки пациентов по следующим категориям: девочки-подростки до 18 лет, юноши-подростки до 18 лет, пенсионеры-женщины (не моложе 55 лет), пенсионеры-мужчины (не моложе 60 лет). Вывести номер истории болезни, фамилию, инициалы, код заболевания и сведения о наблюдении. Известно, что в каждой из групп есть пациенты. Исходные данные вводятся корректно.

Результат работы программы может выглядеть так:

Девушки-подростки:

| Номер ИБ | Фамилия И.О. | Код заболевания | Учет |
|----------|---------------|-----------------|------|
| 1234 | Волохова З.Е. | H005 | Нет |
| 235 | Сергеева А.А. | K900 | Да |

Юноши-подростки:

| | | | |
|-----|-------------|------|----|
| 123 | Петров С.И. | S030 | Да |
|-----|-------------|------|----|

Пенсионеры:

| | | | |
|------|---------------|------|-----|
| 2345 | Иванов С.П. | D500 | Да |
| 222 | Смирнова А.Т. | E006 | Нет |

Пример правильной и эффективной программы на языке Паскаль (Borland Turbo Pascal 7.0)

```
Const tr='Да',fl='Нет';
{Const tr='Yes'; Const fl='No';} {Вариант записи по-
английски. См. замечание к задаче C4-1}
var p:array[1..300] of record
    name: string[50];
    kp:integer;
    pol: char;
    god: integer;
    kod:string[4];
    flag:integer;
end;
m18,d18,p55:array[1..100] of integer;
```

```

fio1: array[1..300] of string[27];
c:char; fio:string;flag_p:string;
  sm,sd,sp,i,k,n,m,j,year:integer;
begin
  readln(n);
  readln(year);
  for i:=1 to 100 do begin {обнуление массивов групп пациентов}
    d18[i]:=0;
    m18[i]:=0;
    p55[i]:=0;
  end;
  sp:=0;sd:=0;sm:=0;
  for i:=1 to n do
  with p[i] do begin
    fio:= ""; name:= "";
    repeat
      read(c);name:=name+c;
      fio:= fio +c
    until c=' '; {считана фамилия}
    read(c); name:=name+c;  fio:= fio +c+'. '; {добавление 1-го
инициала и точки}
    repeat
      read(c); name:=name+c;
    until c=' '; {считано имя, можно запомнить name}
    read(c); name:=name+c;
    fio:= fio +c+'. '; {добавление 2-го инициала и точки}
    {fio — фамилия с инициалами}
    repeat
      read(c); name:=name+c;
    until c=' '; {считано отчество, можно запомнить name }
    fio1[i]:=fio;
    read(kp); read(c); {считан номер истории болезни (ИБ)} {счи-
тан пробел}
    read(c);pol:=c; read(c); {считан пол м или ж} {считан пробел}
    read(god); read(c); {считан год рождения} {считан пробел}
    kod:= "";
    repeat
      read(c);
      kod:=kod+c
    until c=' '; {считан код болезни }
    read(flag); {считан flag}
  end;
end;

```

```
{юноши}
If (pol='m') and ((year-god)<=18) then begin
Sm:=sm+1;m18[sm]:=kp;end;
{девушки}
If (pol='d') and ((year-god)<=18) then begin
Sd:=sd+1;d18[sd]:=kp;end;
{пенсионеры}
If (pol='d') and ((year-god)>=55) then begin
Sp:=sp+1;p55[sp]:=kp;end;
If (pol='m') and ((year-god)>=60) then begin
Sp:=sp+1;p55[sp]:=kp;end;
end; {конец цикла i}
{вывод фамилий пациентов по группам}
Writeln('Девушки-подростки:');
Writeln('Номер ИБ Фамилия И.О. Код заболевания Учет');
For i:=1 to sd do
With p[i] do
Begin
If flag=1 then flag_p:=tr else flag_p:=fl;
writeln(d18[i]:10,fio1[i], ' ',kod, ' ',flag_p);
end;
Writeln('Юноши-подростки:');
Writeln('Номер ИБ Фамилия И.О. Код заболевания
Учет');
For i:=1 to sm do
With p[i] do
Begin
If flag=1 then flag_p:=tr else flag_p:=fl;
writeln(m18[i]:10,fio1[i], ' ',kod, ' ',flag_p);
end;
Writeln('Пенсионеры: ');
Writeln('Номер ИБ Фамилия И.О. Код заболевания Учет');
For i:=1 to sp do
With p[i] do
Begin
If flag=1 then flag_p:=tr else flag_p:=fl;
writeln(p55[i]:10,fio1[i], ' ',kod, ' ',flag_p);
end;
Readln;
end.
```

5. На вход программе подаются сведения об учениках школы (не более 300 чел.) В первой строке сообщается количество учащихся N, каждая из следующих N строк имеет формат: <Фамилия> <Имя> <Название класса>, где <Фамилия> — строка, состоящая не более чем из 20 символов, <Имя> — строка, состоящая не более чем из 12 символов, <Номер класса> — целое число (например, 11, 9, 10). <Фамилия>, <Имя>, <Номер класса> разделены одним пробелом. Примечание: фамилии и имена вводятся латинскими буквами.

Пример входных строк:

150

Кузнецов Сергей 9

Светлов Андрей 11

Кузнецов Сергей 9

Требуется написать как можно более эффективную программу (укажите используемую версию языка программирования, например, Borland Pascal 7.0), которая определит, есть ли в школе однофамильцы. Вывести сообщение о наличии учащихся с одинаковыми фамилиями и их количество. Фамилии Иванов и Иванова, Репин и Репина, Одоевская и Одоевский считать различными.

Пример выходных строк:

Иванов 2 учащихся

Сидоров 3 учащихся

Петров 2 учащихся

Петрова 3 учащихся

Пример правильной и эффективной программы на языке Паскаль (Borland Turbo Pascal 7.0)

```
var p:array[1..300] of record
    famm:string[20];
    name: string[12];
    klass: integer;
end;
```

```
c:char;
l,j,n,k:integer;
Fm:string;
Fam:array[1..300] of string[20];
begin
  readln(n); {количество учащихся}
  for i:=1 to n do
    With p[i] do
      Begin
        famm:="";
        repeat
          read(c);
          famm:=famm+c
        until c=' '; {считана фамилия}
        fam[i]:=famm;
        name:="";
        repeat
          read(c);
          name:=name+c
        until c=' '; {считано имя}
        read(klass); {считан класс}
        readln; {Обязательно, чтобы перед следующей символьной
переменной не было ВК и ПС}
      end; {end i}
    for i:=1 to n do
      writeln(fam[i]); {Вывод списка фамилий из массива для упо-
рядочения элементов в порядке возрастания}
    {упорядочение по алфавиту А->Я}
    {сортировка методом обмена}
    For i:=1 to n-1 do
      For j:=i+1 to n do
        If fam[i]>fam[j] then
          begin
            Fm:= fam[i];
            fam[i]:= fam[j];
            fam[j]:=Fm;
          End; {end if}
    for i:=1 to n do
      writeln(fam[i]); {вывод упорядоченного массива фамилий}
    K:=1; {Поиск одинаковых элементов в массиве и их количества}
```

Окончание программы

```

for i:=1 to n-1 do begin
if fam[i]=fam[i+1] then k:=k+1 else
  if k<>1 then begin writeln(fam[i-1], ' - ', k, 'уч-ся'); k:=1; end;
end;
if k<>1 then writeln(fam[n]); {Вывод последней фамилии списка,
если она является повторяющейся фамилией}
readln;
end.

```

6. На вход программе подаются сведения о спортсменах соревнования по бегу на длинные дистанции (не более 300 чел.) В первой строке сообщается количество спортсменов N , каждая из следующих N строк имеет формат: <Фамилия> <Имя> <Страна> <Время старта> <Время финиша>, где <Фамилия> — строка, состоящая не более чем из 20 символов, <Имя> — строка, состоящая не более чем из 12 символов, <Страна> — строка, состоящая не более чем из 20 символов, <Время старта> <Время финиша> имеют формат чч.мм, где чч — часы, мм — минуты, разделенные двоеточием (например, 14:12). <Фамилия>, <Имя>, <Класс>, <Время старта> и <Время финиша> разделены одним пробелом.

Пример входных строк:

150

Кузнецов Сергей USA 14:10 15:10

Светлов Андрей Russia 12:12 13:18

Петров Сергей Canada 10:04 10:55

Требуется написать как можно более эффективную программу (укажите используемую версию языка программирования, например Borland Pascal 7.0), которая выводит нумерованный список спортсменов в порядке увеличения времени, затраченного на дистанцию, и название страны победителя (если таких спортсменов несколько, также вывести названия их стран). Время вывести в минутах.

Пример выходных строк:

1. Петров Сергей 51 мин. Canada
2. Кузнецов Сергей 60 мин.
3. Светлов Андрей 66 мин.

Пример правильной и эффективной программы на языке Паскаль (Borland Turbo Pascal 7.0)

```

uses crt;
var p:array[1..300] of record
    fam:string[20];
    name:string[12];
    country:string[20];
    time:integer;
end;
c,m:char;
l,j,n,k,time1,tmin:integer;
Fm:string;
Fam:array[1..300] of string[20];
nz:array[1..300] of integer;
begin
    readln(n); {количество спортсменов}
    tmin:=180;
    for i:=1 to n do
        With p[i] do
            Begin
                nz[i]:=i;      {заполнение массива индексов записей}
                fam:="";
                repeat
                    read(c);
                    fam:=fam+c
                until c=' ';    {считана фамилия}
                name:="";
                repeat
                    read(c);
                    name:=name+c
                until c=' ';    {считано имя}
                country:="";
                repeat
                    read(c);
                    country:=country +c
                until c=' ';    {считана страна}
            end;
        end;
    end;

```

```

read(c,m); {считаны часы старта}
time:=60*((ord(c)-ord('0'))*10+ord(m)-ord('0'));
read(c,c,m); {пропущено двоеточие, считаны минуты старта}
time:=time+(ord(c)-ord('0'))*10+ord(m)-ord('0');
read(c,c,m); {пропущен пробел, считаны часы финиша}
time1:=60*((ord(c)-ord('0'))*10+ord(m)-ord('0'));
{Readln - обязательно, чтобы перед следующей символьной
переменной не
было ВК и ПС}
readln(c,c,m); {считаны минуты финиша}
time1:=time1+(ord(c)-ord('0'))*10+ord(m)-ord('0');
time:=time1-time;
if time<tmin then tmin:=time;
end; {end i}
{упорядочение по возрастанию результатов соревнования}
{сортировка методом обмена с учетом массива индексов
записей}
For i:=1 to n-1 do
  For j:=i+1 to n do
    If p[i].time>p[j].time then
      begin
        time1:=p[i].time;k:=nz[i];
        p[i].time:= p[j].time;nz[i]:=nz[j];
        p[j].time:=time1;nz[j]:=k;
      End; {end if}
  for i:=1 to n do
    begin
      K:=nz[i];
      if p[i].time=tmin then writeln(I,' ',p[k].fam,p[k].name,p[i].
time:5,'
',p[k].country) else
        writeln(I,' ',p[k].fam,p[k].name,p[i].time:5); {вывод
упорядоченного
массива
фамилий и результатов соревнований + страна для
победителей}
    end;
  repeat until keypressed;
end.

```

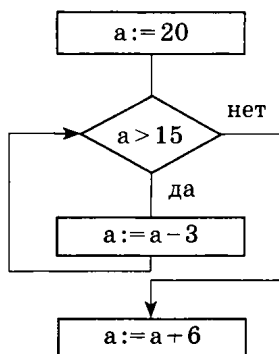

ТЕМАТИЧЕСКИЙ ТЕСТ

Тест состоит из трех частей и содержит 13 заданий. Время выполнения теста — 80–120 минут (в зависимости от навыков программирования). За каждое правильно выполненное задание части 1 и 2 можно набрать 1 балл, за выполнение заданий части 3 можно набрать от 2 до 4 баллов. Если задание вызывает затруднение, перейдите к следующему. К невыполненным заданиям можно будет вернуться, если останется время.

Часть 1

При выполнении заданий этой части необходимо поставить «х» в клетке с номером ответа, который вы считаете правильным.

1. Задан фрагмент блок-схемы



Ему соответствует фрагмент программы

| | БЕЙСИК | ПАСКАЛЬ | ПСЕВДОКОД |
|----|---|--|---|
| 1) | 10 a = 20 20 if a < 15 then a = a - 3 30 a = a + 6 | a := 20; if a < 15 then a := a - 3; a := a + 6; | a := 20 <u>если</u> a < 15 <u>то</u> a := a - 3 <u>все</u> a := a + 6 |

Окончание табл.

| | БЕЙСИК | ПАСКАЛЬ | ПСЕВДОКОД |
|----|---|---|--|
| 2) | 10 a = 20 20 if a < 15 then a = a + 6 30 a = a - 3 | a := 20; if a < 15 then a := a + 6; a := a - 3; | a := 20 <u>если</u> a < 15 то a := a + 6 <u>все</u> a := a - 3 |
| 3) | 10 a = 20 20 while a > 15 30 a = a - 3 40 wend 50 a = a + 6 | a := 20; While a > 15 do a := a - 3; end; a := a + 6; | a := 20 <u>пока</u> a > 15 a := a - 3 <u>кц</u> a := a + 6 |
| 4) | 10 a = 20 20 while a < 15 30 a = a + 6 40 wend 50 a = a - 3 | a := 20; While a < 15 do a := a + 6; end; a := a - 3; | a := 20 <u>пока</u> a < 15 a := a + 6 <u>кц</u> a := a - 3 |

1) 1

2) 2

3) 3

4) 4

2. Имеется таблица, состоящая из трех строк и четырех столбцов. Чему будет равна сумма элементов первого столбца таблицы после выполнения фрагмента программы?

| БЕЙСИК | ПАСКАЛЬ | ПСЕВДОКОД |
|---|--|--|
| 10 for i = 1 to 3 20 for j = 1 to 4 30 if j > i then 60 40 a(i, j) = i - j 50 goto 70 60 a(i, j) = i + j 70 next j 80 next i | for i := 1 to 3 do for j := 1 to 4 do if j > i then A[i, j] := i + j else A[i, j] := i - j; | <u>нц для</u> i <u>от</u> 1 <u>до</u> 3 <u>нц для</u> j <u>от</u> 1 <u>до</u> 4 <u>если</u> j > i то A[i, j] := i + j <u>иначе</u> A[i, j] := i - j <u>все</u> <u>кц</u> <u>кц</u> |

1) 1

2) 2

3) 3

4) 4

3. Задан фрагмент программы, обрабатывающий массив X размером N. После выполнения фрагмента программы

| БЕЙСИК | ПАСКАЛЬ | ПСЕВДОКОД |
|--|---|--|
| <pre> Imax:=1;imin:=1 For i:=2 to n If X(i)<x(imin) then imin=i else If X(i)>x(imax) then imax=i End if End if Next i y:= X(imax) x(imax)=x(imin) x(imin)=y </pre> | <pre> Imax:=1;imin:=1; For i:=2 to n do Begin If X[i]<x[imin] then imin=i else If X[i]>x[imax] then imax=i End; y:= X[imax]; x[imax]:=x[imin]; x[imin]:=y; </pre> | <pre> Imax:=1 imin:=1 Нц для I от 2 до n если X[i]<x[imin] то imin:=I иначе если X[i]>x[imax] то imax:=I все кц y:= X[imax] x[imax]:=x[imin] x[imin]:=y </pre> |

| | |
|----|--|
| 1) | Массив остался без изменений |
| 2) | Максимальный и минимальный элементы поменялись местами |
| 3) | Максимальный элемент стал первым в массиве |
| 4) | Минимальный элемент стал первым в массиве |

4. В клетки полоски в один ряд записывают 4 символа по следующему правилу: на первом месте может быть один из символов А,Б,Г. На третьем месте — символы А,Г,Д,Е, не стоящие на первом и втором месте. На последнем месте — символы В,Г,Е, не стоящие на третьем месте.

1) ГАДЕ 2) БААГ 3) АГАВ 4) БГЕЕ

5. Даны 3 числа: $A=100$, $B=60$, $C=20$. Были выполнены следующие действия: сравнили A и B , запомнили меньшее число. Затем сравнили полученное число и число C и запомнили большее из них. Какое число запомнили?

1) 20 2) 60 3) 100 4) 160

6. Исполнитель РОБОТ может передвигаться на одну клетку **вверх**, **вниз**, **вправо** и **влево** прямоугольного клетчатого поля, на котором расположены горизонтальные и вертикальные стенки. Двигаться вперед он может только тогда, когда стенок перед ним нет.

Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится РОБОТ:

| | | | |
|--------------------|-------------------|-------------------|--------------------|
| сверху свободно | снизу свободно | слева свободно | справа свободно |
|--------------------|-------------------|-------------------|--------------------|

Цикл

ПОКА $\langle \text{условие} \rangle$ делать
команда1

...

командаN

Конец

выполняется, пока условие истинно, иначе происходит переход на следующую строку.

Сколько клеток приведенного лабиринта соответствуют требованию, что, выполнив предложенную ниже программу, РОБОТ остановится в той же клетке, с которой он начал движение?

НАЧАЛО

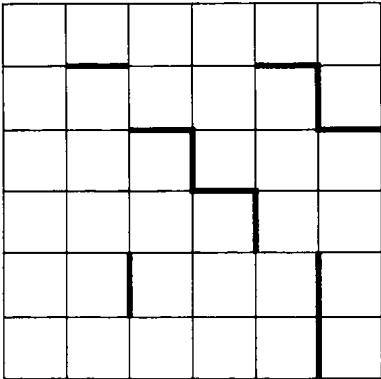
ПОКА $\langle \text{сверху свободно} \rangle$ вверх

ПОКА $\langle \text{справа свободно} \rangle$ вправо

ПОКА $\langle \text{снизу свободно} \rangle$ вниз

ПОКА $\langle \text{слева свободно} \rangle$ влево

КОНЕЦ



1) 1

2) 2

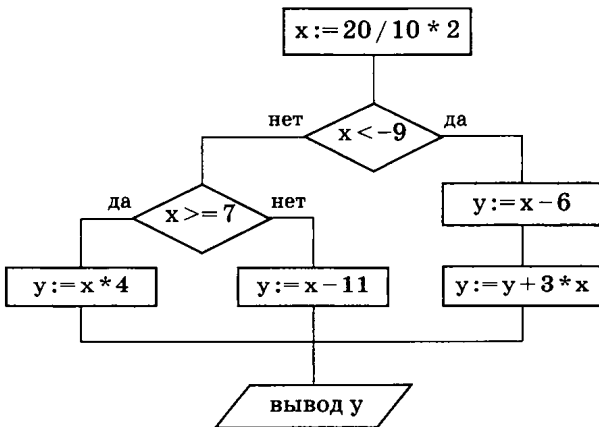
3) 3

4) 4

Часть 2

Во второй части теста необходимо самостоятельно получить ответ в виде числа или последовательности букв и записать его в специально отведенное для него место. Каждый символ следует писать в отдельной клетке, начиная с первой.

1. Запишите значение переменной y после выполнения фрагмента алгоритма



2. Запишите значение переменной f после выполнения фрагмента алгоритма для переменной целого типа $x=555$.

| БЕЙСИК | ПАСКАЛЬ | ПСЕВДОКОД |
|---|--|---|
| $B = x \bmod 100 \backslash 10$ $C = x \bmod 100 \backslash 10$ $D = x \bmod 10$ $Y = 100 * c + 10 * b + d$ $f = x - y$ | $B := x \bmod 100 \div 10;$ $C := x \bmod 10;$ $D := x \div 100;$ $Y := 100 * c + 10 * b + d;$ $f := x - y;$ | $B := \text{div}(\text{mod}(x, 100), 10)$ $C := \text{div}(\text{mod}(x, 100), 10)$ $D := \text{mod}(x, 10)$ $Y := 100 * c + 10 * b + d$ $f := x - y$ |

Ответ: _____

3. Исполнитель Черепашка перемещается на экране компьютера, оставляя след в виде линии. В каждый конкретный момент известно положение исполнителя и направление его движения. У исполнителя существуют две команды:

Вперед n , где n — целое число, вызывает передвижение черепашки на n шагов в направлении движения.

Направо m , где m — целое число, вызывает изменение направления движения на m градусов по часовой стрелке (от 0° до 359°).

Запись **Повтори 5 [Команда1 Команда2]** означает, что последовательность команд в скобках повторится 5 раз.

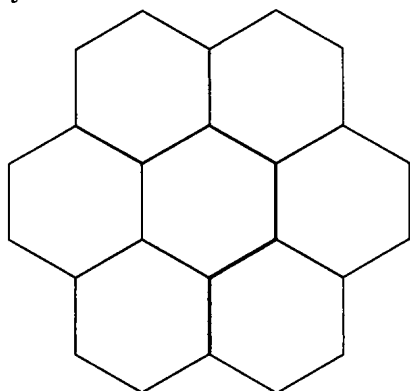
Исполнитель интерпретирует эту запись как одну команду.

Черепашке был дан для исполнения следующий алгоритм:

Повтори 10 [Повтори 6 [Вперед 100 Направо X] Направо Y вперед 100 Налево Z]

На экране появилась следующая фигура из семи шестиугольников, плотно прилегающих друг к другу.

Определить и записать в ответе значения X , Y и Z через запятую.



Ответ: _____

4. Строки (цепочки арабских цифр) создаются по следующему правилу.

Первая строка состоит из одного символа — цифры «1». Каждая из последующих цепочек создается следующими действиями: в очередную строку записывается предыдущая цепочка цифр, затем приписывается число — номер строки по порядку, и далее предыдущая цепочка цифр.

Вот первые 4 строки, созданные по этому правилу:

- (1) 1
- (2) 121
- (3) 1213121
- (4) 121312141213121
- (5) 1213121412131215121312141213121

Какая цифра находится на 128-м месте в восьмой строке? Сколько троек находится в этой строке? Перечислите их через запятую, например: 0,20.

Ответ: _____

Часть 3

В третьей части теста также необходимо самостоятельно получить ответ и записать полное решение и полученный ответ в бланк ответов № 2.

Ответы записывайте четко и разборчиво.

Для проверки правильности работы программы используйте соответствующую среду программирования.

1. Требовалось написать программу, чтобы вычислить квадратный корень из числа a с точностью e . Программист написал программу с ошибкой.

| ПАСКАЛЬ | БЕЙСИК |
|--|--|
| <pre> Var I,k,n:integer; {n — кол-во итераций, шагов} A,b,b0,e:real; {b0-начальное приближение} Begin {b-искомое приближение корня} Write('введите число a='); Readln(a); Write('введите точность e='); Readln(e); Write('введите число b=');Readln(b); If a>1 then b0:=1 else b0:=0.01; {b0 — начальное приближение} N:=1; {N — счетчик итераций} B:=0.5*(b0+a/b0); While (b-b0)>e do Begin B0:=b; {заменяем старое приближение на новое} B:=0.5*(b0+a/b0); {вычисляем новое приближение} N:=n+1; end; Writeln('b=',b); {результат} Writeln('n=',n); {кол-во итераций} Readln; End.</pre> | <pre> Defint I,k,n Input "введите число a=";a Input "введите точность e=";e Input "введите число b=";b If a>1 then c=1 else c=0.001 'c — начальное приближение End if N=1 'N — счетчик итераций B=0.5*(c+a/c) 'b — искомое приближение корня While (b-c) > e C=b 'заменяем старое приближение на новое B=0.5*(c+a/c) 'вычисляем новое приближение N=n+1 Wend Print "b=";b 'результат Print "n=";n 'кол-во итераций end</pre> |

- Привести пример таких данных **a** и **e**, для которых программа дает неправильный результат.
 - Исправить ошибку в программе так, чтобы она соответствовала поставленной задаче.
 - Укажите, какая часть программы является лишней.
2. Найти сумму положительных элементов массива размерности **N** и вывести ее значение.
3. На вход программе подаются сведения о сотрудниках фирмы (не более 300 чел.) В первой строке сообщается количество сотрудников **N**, каждая из следующих **N** строк имеет формат: <Фамилия> <Имя> <Отчество> <Профессия>, где <Фамилия> — строка, состоящая не более чем из 20 символов, <Имя> — строка, состоящая не более чем из 12 символов, <Отчество> — строка, состоящая не более чем из 20 символов, <Профессия> — строка, состоящая не более чем из 12 символов. <Фамилия>, <Имя>, <Отчество>, <Профессия> разделены одним пробелом. Ввод профессии оканчивается точкой.

Пример входных строк:

150

Кузнецов Сергей Иванович инженер.

Светлов Андрей Петрович лаборант.

Кузнецов Сергей Сергеевич журналист.

Требуется написать как можно более эффективную программу (укажите используемую версию языка программирования, например Borland Pascal 7.0), которая выведет списки сотрудников (фамилия, инициалы) по профессиям с указанием их количества.

Пример выходных строк:

Профессия инженер — 5 чел.

Иванов А.Р.

Сидоров М.С.

Кузнецов С.И.

Павлов М.Д.

Денисов П.Л.

Профессия журналист — 1 чел.

Кузнецов С.С.

ОТВЕТЫ

1. Информация и ее кодирование

Часть 1

| № задания | Ответ | № задания | Ответ |
|-----------|-------|-----------|-------|
| A1 | 1 | A11 | 1 |
| A2 | 2 | A12 | 3 |
| A3 | 3 | A13 | 2 |
| A4 | 3 | A14 | 3 |
| A5 | 1 | A15 | 3 |
| A6 | 2 | A16 | 2 |
| A7 | 4 | A17 | 4 |
| A8 | 3 | A18 | 3 |
| A9 | 2 | A19 | 2 |
| A10 | 3 | A20 | 4 |

Часть 2

| № | Ответ |
|----|-------------------|
| B1 | 6 |
| B2 | 1024×768 |
| B3 | 536870912 |
| B4 | 110101110 |
| B5 | 1 |
| B6 | 2 |
| B7 | 1024 |

2. Основы логики

Часть 1

| № задания | Ответ |
|-----------|-------|
| 1 | 1 |
| 2 | 4 |
| 3 | 1 |
| 4 | 1 |
| 5 | 2 |
| 6 | 4 |
| 7 | 4 |
| 8 | 3 |
| 9 | 2 |

Часть 2

| № | Ответ |
|---|----------------|
| 1 | 9 |
| 2 | 4 |
| 3 | РГПИ |
| 4 | 4Л, 1Т, 3А, 2К |
| 5 | АБ, НР, ТГ, ДЗ |

Часть 3

1. Ответ: Выигрывает первый игрок. Первым ходом следует поставить любой символ («крестик» или «нолик») в центральную клетку. После этого придерживаться следующей тактики: ставить соответствующий символ в клетку симметричную той, в которую поставил символ второй игрок.

| | | |
|---|---|---|
| х | | |
| | 0 | |
| | | х |

2. Ответ: Выигрывает первый игрок, своим первым ходом он должен поставить фишку в точке с координатами (3,4). Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке координаты фишки на каждом этапе игры.

| 1 ход | 2 ход | 3 ход | 4 ход | 5 ход |
|----------------------------|------------------------------|--------------------------|--|-----------------------------|
| Позиция после первого хода | II игрок (все варианты хода) | I игрок (выигрышный ход) | II игрок (все варианты хода) | I игрок (один из вариантов) |
| 3,4 | 6,4 | 6,6 | 9,6 | 12,6 |
| | | | 6,8 | 9,8 |
| | | | 6,10 | 9,10 |
| | 3,6 | 6,6 | Те же варианты 4–5 хода | |
| | 3,8 | 3,12 | Первый игрок выигрывает ответным ходом | |

Таблица содержит *все возможные* варианты ходов второго игрока. Из нее видно, что при любом ответе второго игрока у первого имеется ход, приводящий к победе. [1]

3. «Информационные и коммуникационные технологии» и «Моделирование и компьютерный эксперимент»

Часть 1

| № задания | Ответ | № задания | Ответ | № задания | Ответ |
|-----------|-------|-----------|-------|-----------|-------|
| 1 | 4 | 7 | 3 | 13 | 2 |
| 2 | 2 | 8 | 2 | 14 | 3 |
| 3 | 2 | 9 | 4 | 15 | 4 |
| 4 | 3 | 10 | 3 | 16 | 2 |
| 5 | 2 | 11 | 1 | 17 | 1 |
| 6 | 2 | 12 | 4 | 18 | 3 |

Часть 2

| № | Ответ |
|----------|----------------------|
| 1 | ВГАБ |
| 2 | 4123 |
| 3 | 3142 или 3124 |
| 4 | 80 |

Ответы
на тест по разделу
«Алгоритмизация и программирование»

| | | | | | | | |
|----------------|-------------------|-----------|----------|------------------|-------------|----------|-----------|
| Часть 1 | № задания | 1 | 2 | 3 | 4 | 5 | 6 |
| | код ответа | 3 | 3 | 2 | 1 | 2 | 2* |
| Часть 2 | № задания | 1 | 2 | 3 | 4 | | |
| | ответ | -7 | 0 | 60,120,60 | 8,32 | | |

| | | | | | |
|--|--|--|--|--|---|
| | | | | | |
| | | | | | * |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | * |

Ответ на часть 3

Ответ на № 1.

| Исходный тест программы с ошибкой | |
|---|--|
| Паскаль | Бейсик |
| <pre> Var I,k,n:integer; {n — кол-во итераций, шагов} A,b,b0,e:real; {a0 — начальное приближение} Begin {b — искомое приближение корня} Readln(a); Readln(e); Readln(b); If a>1 then b0:=1 else b0:=0.01; N:=1; B:=0.5*(b0+a/b0); While (b-b0)>e do Begin B0:=b; {заменяем старое приближение на новое} B:=0.5*(b0+a/b0); {вычисляем новое приближение} N:=n+1; end; Writeln('b=',b); {результат} Writeln('n=',n); Readln; End.</pre> | <pre> Defint I,k,n Input "введите число a=";a Input "введите точность e=";e Input "введите число b=";e If a>1 then c=1 else c=0.001 'c — начальное приближение End if N=1 'N — счетчик итераций B=0.5*(c+a/c) 'b — искомое приближение корня While (b-c) > e C=b 'заменяем старое приближение на новое B=0.5*(c+a/c) 'вычисляем новое приближение N=n+1 Wend Print "b="; b 'результат Print "n="; n 'кол-во итераций end</pre> |

- Привести пример таких данных **a** и **e**, для которых программа дает неправильный результат.

Ответ: например, для $a=4$ и $e=0.001$, при которых $b=2.05$ и заданная точность e не достигнута (см. таблицу пошагового выполнения алгоритма):

| n | b | b0 | b-b0 |
|---|------|-----|---|
| 1 | 2.5 | 1 | $1.5 > e$ |
| 2 | 2.05 | 2.5 | $-0.45 < e$ — выход из цикла, хотя точность не достигнута — ошибка! |

Ответ на 1 вопрос: для $a=4$ и $e=0.001$ мы получим результат $b=2.05$, при этом $b-b0=-0.45 < e$, не достигнута заданная точность вычисления корня.

Теперь исправим ошибку. Будем вычислять модуль разности приближений корня на соседних итерациях: $\text{abs}(b-b0) > e$, т.е. сравниваем модуль разности с заданной точностью. Получим:

| n | b | b0 | Abs(b-b0) |
|---|------------|--------|--|
| 1 | 2.5 | 1 | $1.5 > e$ |
| 2 | 2.05 | 2.5 | $0.45 > e$ — цикл продолжается |
| 3 | 2.0006 | 2.05 | $0.0494 > e$ — цикл продолжается |
| 4 | 2.00000009 | 2.0006 | $0.00059991 < e$ — выход из цикла, точность достигнута |

- Исправить ошибку в программе так, чтобы она соответствовала поставленной задаче.

Ответ: Заменить `While (b-b0)>e do` на команду `While abs(b-b0)>e do`.

- Укажите, какая часть программы является лишней.

Ответ: `Readln(b);` — лишняя часть, так как переменная b — это результат работы программы (корень из числа a).

| Правильная программа | |
|---|--|
| Паскаль | Бейсик |
| <pre> Var I,k,n:integer; {n — кол-во итераций, шагов} A,b,b0,e:real; {a0 — начальное приближение} Begin {b — искомое приближе- ние корня} Readln(a); Readln(e); If a>1 then b0:=1 else b0:=0.01; N:=1; B:=0.5*(b0+a/b0); While abs(b-b0)>e do Begin B0:=b; {заменяем старое при- ближение на новое} B:=0.5*(b0+a/b0); {вычисляем новое приближение} N:=n+1; end; Writeln('b=',b); {результат} Writeln('n=',n); Readln; End.</pre> | <pre> Defint I,k,n Input "введите число a="; a Input "введите точность e="; e If a>1 then c=1 else c=0.001 'c — начальное приближение End if N=1 'N — счетчик итера- ций B=0.5*(c+a/c) 'b — иско- мое приближение корня While abs(b-c) > e C=b 'заменяем старое при- ближение на новое B=0.5*(c+a/c) 'вычисляем новое приближение N=n+1 Wend Print "b="; b 'результат Print "n="; n 'кол-во ите- раций end</pre> |

Ответ на № 2.

2. Найти сумму положительных элементов массива размерности N и вывести ее значение.

Алгоритм решения задачи будет состоять из следующих блоков:

1. Описание одномерного массива и целых переменных.
2. Обнуление значения переменной s для хранения суммы элементов.
3. В цикле выполнить формирование элементов массива с помощью функции получения случайных чисел, вывод элементов и проверку, что элемент является положительным.

4. Если элемент массива положителен, то он накапливается в переменной *s*.
5. Конец цикла.
6. Вывод значения суммы положительных элементов на экран.

**Пример правильной программы
на языке Паскаль (Borland Turbo Pascal 7.0)**

```
Const m=100;
Type mas=array[1..m] of integer;
Var x:mas; { Описание одномерного массива }
I,j,s:integer; { Описание целых переменных }
Begin
Randomize;
s:=0; {Обнуление значения переменной s для хранения
суммы элементов }
For i:=1 to n do {Начало цикла}
begin
x[i]:=-30+random(50); {Получение случайных чисел для
заполнения массива}
Write(x[i]:5); {Вывод элементов массива}
if (x[i] > 0) then s:=s+x[i]; {Проверка положительности
элемента массива и накапливание его в переменной s}
end; {Конец цикла}
Writeln;
write('summa=', s:10); {Вывод значения суммы
положительных элементов на экран}
end.
```

Ответ на № 3.

Условие задачи

3. На вход программе подаются сведения о сотрудниках фирмы (не более 300 чел.) В первой строке сообщается количество сотрудников *N*, каждая из следующих *N* строк имеет формат: <Фамилия> <Имя> <Отчество> <Профессия>, где <Фамилия> — строка, состоящая не более чем из 20 символов; <Имя> — строка, состоящая не более чем из 12 символов; <Отчество> — строка, состоящая не более чем из 20 символов; <Профессия> — строка, состоящая не более чем из 12 символов (всего 10 профессий). <Фамилия>, <Имя>, <Отчество>, <Профессия> разделены одним пробелом.

Пример входных строк:

150

Кузнецов Сергей Иванович инженер

Светлов Андрей Петрович лаборант

Кузнецов Сергей Сергеевич журналист

Требуется написать как можно более эффективную программу (укажите используемую версию языка программирования, например Borland Pascal 7.0), которая выведет списки сотрудников (фамилия, инициалы) по профессиям с указанием их количества.

Пример выходных строк:

Профессия инженер — 5 чел.

Иванов А.Р.

Сидоров М.С.

Кузнецов С.И.

Павлов М.Д.

Денисов П.Л.

Профессия журналист — 1 чел.

Кузнецов С.С.

Алгоритм решения задачи**Пример правильной программы на языке Паскаль (Borland Turbo Pascal 7.0)**

{список профессий сотрудников}

```
const prf:array[1..10] of string [12] = ('prf1','prf2','prf3','prf4',
'prf5','prf6','prf7','prf8','prf9','prf10');
```

{названия профессий по-английски}

const m=10; {количество профессий}

```
{const d:array[1..10] of string [12] = ('инженер', 'програм-
мист', 'журналист', 'уборщица', 'охранник', 'директор',
'замдиректора', 'секретарь', 'бухгалтер', 'экономист');
```

{названия профессий по-русски. Не все версии Паскаля поддерживают кириллицу, поэтому при отладке программы на компьютере используйте только латиницу}

```
var kp:array[1..10] of integer; {массив кол-ва работников по
профессиям}
fio:array[1..300] of string[25]; {массив ФИО сотрудников}
c:char; fio1:string[25];
i,k,n,j:integer;
begin
  readln(n);
  for j:=1 to m do {обнуление массива кол-ва сотрудников по
профессиям}
    kp[j]:=0;
  for i:=1 to n do
    with p[i] do begin
      fio1:= ''; name:= '';
      repeat
        read(c); name:=name+c;
        fio1:= fio1 +c
      until c=' '; {считана фамилия}
      read(c); name:=name+c;  fio1:= fio1 +c+'.'; {добавление 1-го
инициала и точки}
      repeat
        read(c); name:=name+c;
      until c=' '; {считано имя, можно запомнить name}
      read(c); name:=name+c;
      fio1:= fio1 +c+'.'; {добавление 2-го инициала и точки}
      repeat
        read(c); name:=name+c; {в name сохраняются полностью
ФИО}
      until c=' '; {считано отчество, можно запомнить name}
      fio[i]:=fio1;
    {fio — фамилия с инициалами}
    read(c);
    while c<>'.' do
      begin
        prof:=prof+c;    {считывание названия профессии}
        read(c);
      end;                {end of While}
```

```
readln;
for j:=1 to m do
If prof=prf[j] then kp[j]:=kp[j]+1; {подсчет кол-ва сотрудников
по профессиям}
end;    {конец цикла i}
{вывод фамилий сотрудников по профессиям}
for i:=1 to m do begin
writeln('Профессия ',prf[i],' ',kp[i],' чел. '); {Вывод профес-
сии и кол-ва сотрудников }
for j:=1 to n do begin
if p[j].prof=prf[i] then writeln(fio[j]); {вывод сотрудников
данной профессии}
end;    {конец цикла j}
writeln;
end;    {конец цикла i}
Readln;
end.
```

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. *Андреева Е.В., Лещинер В.Р., Самылкина Н.Н., Якушкин П.А., Крылов С.С.* Методические рекомендации по оцениванию заданий с развернутыми ответами. Информатика. ЕГЭ 2009. — М.: Федеральный институт педагогических измерений, 2009.

2. *Бешенков С.А., Ракитина Е.А.* Информатика. Систематический курс. Учебник для 10 класса. — М.: Лаборатория базовых знаний, 2001. — 432 с. с илл.

3. Единый государственный экзамен 2006. Информатика. Учебно-тренировочные материалы для подготовки учащихся/ Рособрназор, ИСОП — М.: Интеллект-Центр, 2006. — 136 с.

4. *Лихтарников Л.М.* Занимательные логические задачи. — СПб.: Лань, МИК, 1996. — 126 с.

5. *Макарова Н.В., Николайчук Г.С., Титова Ю.Ф.* Информатика и информационно-коммуникационные технологии. 10-й класс. Базовый уровень. — СПб.: Питер, 2006. — 238 с. с илл.

6. *Самылкина Н.Н.* Построение тестовых заданий по информатике. Методическое пособие. — М.: БИНОМ. Лаборатория знаний, 2003. — 176 с. с илл.

7. *Самылкина Н.Н., Русаков С.В., Шестаков А.П., Баданина С.В.* Готовимся к ЕГЭ по информатике. Элективный курс: учебное пособие. — М.: БИНОМ. Лаборатория знаний, 2008. — 298 с. с илл.

8. *Смаллиан Р.* Принцесса или тигр?: Пер. с англ./ Под ред. и с предисл. Ю.И. Манина. — Мир, 1985. — 221 с.

9. *Сэм Ллойд.* Математическая мозайка. Сост. и ред. М. Гарднер / Пер. с англ. — М.: РИПОЛ, 1995. — 352 с. с илл.

10. www.ege.edu.ru/ Демонстрационный вариант по информатике за 2005–2010 гг.

СОДЕРЖАНИЕ

| | |
|---|------------|
| Введение | 3 |
| Как работать с изданием | 4 |
| 1. ИНФОРМАЦИЯ И ЕЕ КОДИРОВАНИЕ | 6 |
| 2. ОСНОВЫ ЛОГИКИ | 20 |
| 3. «ИНФОРМАЦИОННЫЕ И КОММУНИКАЦИОННЫЕ ТЕХНОЛОГИИ» И «МОДЕЛИРОВАНИЕ И КОМПЬЮТЕРНЫЙ ЭКСПЕРИМЕНТ» | 34 |
| 4. АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ | 49 |
| Ответы | 138 |
| Список использованной литературы | 149 |

Издание для дополнительного образования

Для старшего школьного возраста

ЕГЭ. СДАЕМ БЕЗ ПРОБЛЕМ

Островская Екатерина Михайловна

Самылкина Надежда Николаевна

ЕГЭ 2012

ИНФОРМАТИКА

Сдаем без проблем!

Ответственный редактор *А. Жилинская*

Ведущий редактор *Т. Судакова*

Художественный редактор *Е. Брынчик*

Технический редактор *Л. Зотова*

Компьютерная верстка *И. Кондратюк*

Корректор *Е. Анищенко*

ООО «Издательство «Эксмо»

127299, Москва, ул. Клары Цеткин, д. 18/5. Тел. 411-68-86, 956-39-21.

Home page: www.eksmo.ru E-mail: info@eksmo.ru

Подписано в печать 01.09.2011.

Формат 60×90^{1/16}. Гарнитура «Школьная». Печать офсетная.

Бумага тип. Усл. печ. л. 10.0.

Доп. тираж 3000 экз. Заказ № 4102501

Отпечатано с готовых файлов заказчика

в «НИЖПОЛИГРАФЕ»,

филиале ОАО «Первая Образцовая типография»

603950, г. Нижний Новгород, ГПС - 123, ул. Варварская, 32.

ISBN 978-5-699-51291-1



9 785699 512911 >