

Ответы к задачам

Глава 1. Информация и информационные процессы

§ 1. Количество информации

- 1) 3
- 2) 0,25
- 3) 0,1; 0,2; 0,7
- 4) 0,1; 0,25; 0,75; 0,01
- 5) $\frac{10}{21}; \frac{1}{3}; \frac{2}{21}$
- 6) 2
- 7) 3
- 8) 4
- 9) 32
- 10) 1
- 11) 2
- 12) 5
- 13) 24
- 14) 2
- 15) 5
- 16) 30

Формула Шеннона

- 1) 0,3
- 2) $H = 0,3 \cdot \log_2 \frac{1}{0,3} + 0,7 \cdot \log_2 \frac{1}{0,7} \approx 0,881$ бита
- 3) $H = 0,3 \cdot \log_2 \frac{1}{0,3} + 0,4 \cdot \log_2 \frac{1}{0,4} + 0,3 \cdot \log_2 \frac{1}{0,3} \approx 1,571$ бита

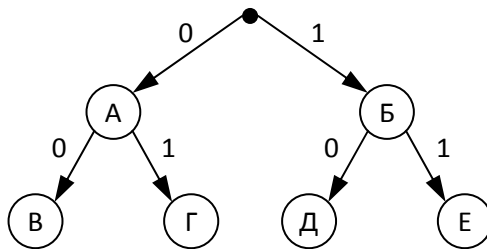
§ 2. Передача данных

- 1) 40 с
- 2) 3750 Кбайт
- 3) 48 с
- 4) 1296 с
- 5) 1875 Кбайт
- 6) 45 с
- 7) 1800 страниц
- 8) 100
- 9) 96
- 10) 150
- 11) 256000 бит/с
- 12) 16 бит
- 13) 3, 2
- 14) 01101, 10101, 11001, 11111, 11100; n
- 15) 12345

§ 3. Сжатие данных

- 1) 00000001 11000010 10000100 11000000 00000001 11000010 10000011 11000000
 1 В = 194 4 А = 192 1 В = 194 3 А = 192
 10001010 11010000
 10 Р = 208

- 2) 10 байтов
 3) 18 байтов
 4) НЕ ТОНЕТ ЕНОТ
 5) код не префиксный, потому что не все буквы – листья



- 6) М – 11, А – 10, Л – 00, пробел – 011, Ы – 0100, У 0101; $k = 42/34 \approx 1,24$; $k = 3,29$

§ 4. Информация и управление

Задач нет.

§ 5. Информационное общество

Задач нет.

Ответы к задачам

Глава 2. Моделирование

§ 6. Модели и моделирование

1) 20

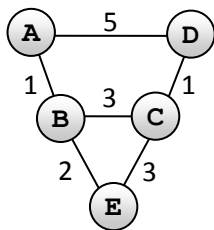
§ 7. Системный подход в моделировании

1) $9 \cdot (9-1) : 2 = 36$

2) $3 \cdot 3 \cdot (3-1) : 2 + 3 \cdot (3-1) : 2 = 12$; легче исследовать систему, разбив ее на подсистемы

3)

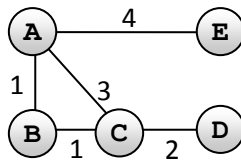
а)



	A	B	C	D	E
A	0	1	0	1	0
B	1	0	1	0	1
C	0	1	0	1	1
D	1	0	1	0	0
E	0	1	1	0	0

	A	B	C	D	E
A		1		5	
B	1		3		2
C		3		1	3
D	5		1		
E		2	3		

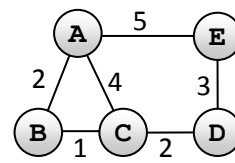
б)



	A	B	C	D	E
A	0	1	1	0	1
B	1	0	1	0	0
C	1	1	0	1	0
D	0	0	1	0	0
E	1	0	0	0	0

	A	B	C	D	E
A		1	3		4
B	1		1		
C	3	1		2	
D			2		
E	4				

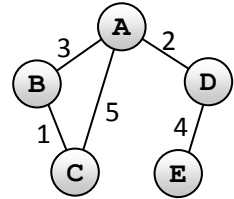
в)



	A	B	C	D	E
A	0	1	1	0	1
B	1	0	1	0	0
C	1	1	0	1	0
D	0	0	1	0	1
E	1	0	0	1	0

	A	B	C	D	E
A		2	4		5
B	2		1		
C	4	1		2	
D			2		3
E	5			3	

г)



	A	B	C	D	E
A	0	1	1	1	0
B	1	0	1	0	0
C	1	1	0	0	0
D	1	0	0	0	1
E	0	0	0	1	0

	A	B	C	D	E
A		3	5	2	
B	3		1		
C	5	1			
D	2				4
E				4	

4) 14

5) -

6) 16:10, прямой рейс

7) 12:50, через Мухино

8) 13:15, через Марьино

9) 12:25, через Озерное

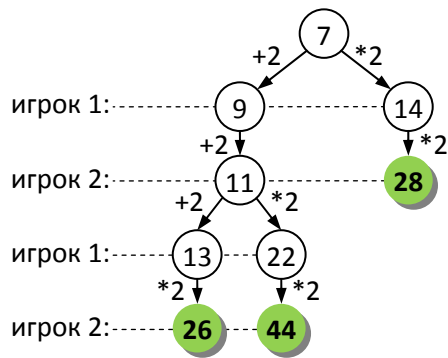
10) в

11) б, г

12) г

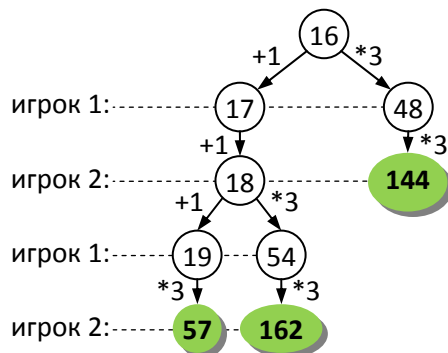
13)

1	2	3	4	5	6	7	8	9	1	1	1	1	1	1	1	1	1	1	1	2	2	2	2	2
									0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	
В	Х	Х	В	В	В	Х	Х	В	В	Х	Х	В	В	В	В	В	В	В	В	В	В	В	В	В
4	3	3	3	3	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	



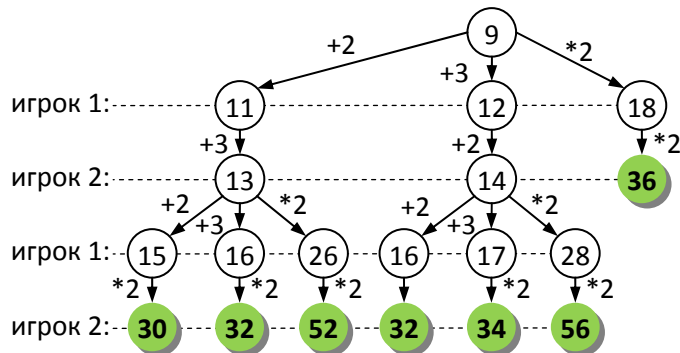
14)

1	2	3	4	5	6	7	8	9	1	1	1	1	1	1	1	1	1	1	2	2	...	5	
B	B	X	B	X	B	B	X	B	X ₅	B ₅	X ₄	B ₄	X ₃	B ₃	X ₂	B ₂	X ₁	B ₁	B ₁	B ₁	...	B ₁	
6	6	5	5	3	2	7	6	6															

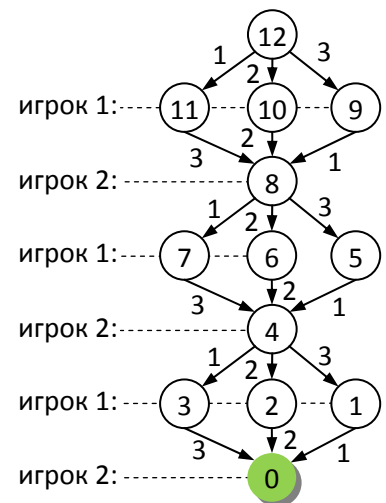


15)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	...	28	29
B ₄	X ₃	X ₃	B ₃	B ₃	B ₃	B ₂	X ₂	X ₂	B ₂	B ₂	B ₂	X ₁	X ₁	B ₁	B ₁	B ₁	...	B ₁	B ₁



16)



15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
B ₄	B ₄	B ₄	X ₃	B ₃	B ₃	B ₃	X ₂	B ₂	B ₂	B ₂	X ₁	B ₁	B ₁	B ₁

§ 8. Этапы моделирования

Задач нет.

§ 9. Моделирование движения

- 1) а) 3,90 с; 18,22 м; -18,45 м/с
 б) 4,1 с; 20,4 м; -20 м/с
 в) -
 г) 3,87 с; 18,35 м; -18,06 м/с
- 2) а) 2,76 с; 9,33 м; 34,76 м; 17,20 м/с
 б) 2,88 с; 10,19 м; 40,77 м; 20 м/с
- 3) а) -; б) 5,4 м/с; в) -

§ 10. Математические модели в биологии

- 1) через 3 интервала
- 2) а) 912, не зависит от начальной численности
 б) переходный процесс в начале развития
 в) 45
- 3) а) через 40 дней
 б) 980 человек; 20 человек
 в) 648 человек
- 4) а) 67 карасей, 33 щуки
 б) значения коэффициентов
 в) на переходный процесс
 г) например, при $b_z = 0,03$ (появляются отрицательные значения)
 г) например, при $b_z = 0,005$
- 5) а) да, потому что учитывает взаимное влияние видов
 б) 64 белки, 38 бурндуков
 в) коэффициенты
 г) на переходный процесс
 д) $D_M \approx 0,153$
 е)

$$N_{i+1} = \left[1 + K_N \cdot \frac{L_N - N_i}{L_N} \right] \cdot N_i - D_N \cdot M_i - E_N \cdot Q_i$$

$$M_{i+1} = \left[1 + K_M \cdot \frac{L_M - M_i}{L_M} \right] \cdot M_i - D_M \cdot N_i - E_M \cdot Q_i$$

$$Q_{i+1} = \left[1 + K_Q \cdot \frac{L_Q - Q_i}{L_Q} \right] \cdot Q_i - D_Q \cdot N_i - E_Q \cdot M_i$$

§ 11. Системы массового обслуживания

- 1) 13

Ответы к задачам

Глава 3. Базы данных

§ 12. Информационные системы

Задач нет.

§ 13. Таблицы

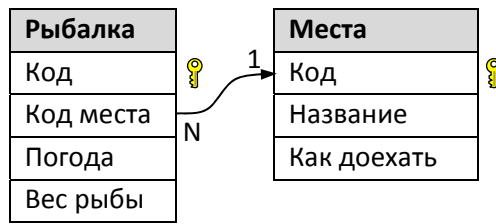
- 1) 1 048 576 сравнений при линейном и 21 – при двоичном поиске, выигрыш примерно в 5000 раз
- 2) первичный ключ – *Дата* и *Номер заказа* (если заказы нумеруются с 1 в течение каждого дня); можно построить индексы по каждому из полей
- 3) первичным ключом может быть номер паспорта или номер аттестата
- 4)

Номер	Дата	Номер	Товар	Номер	Количество, т
1	12.09.10	1	Ананасы	5	11
2	12.09.10	3	Ананасы	1	12
3	13.09.10	2	Апельсины	2	12
4	13.09.10	5	Апельсины	4	13
5	13.09.10	4	Бананы	3	15

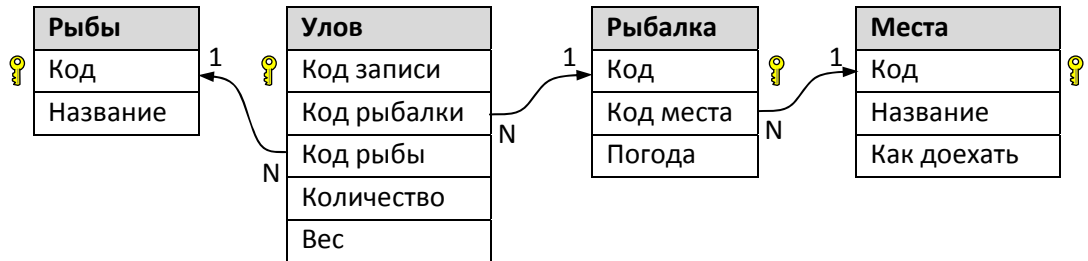
- 5) –

§ 14. Многотабличные базы данных

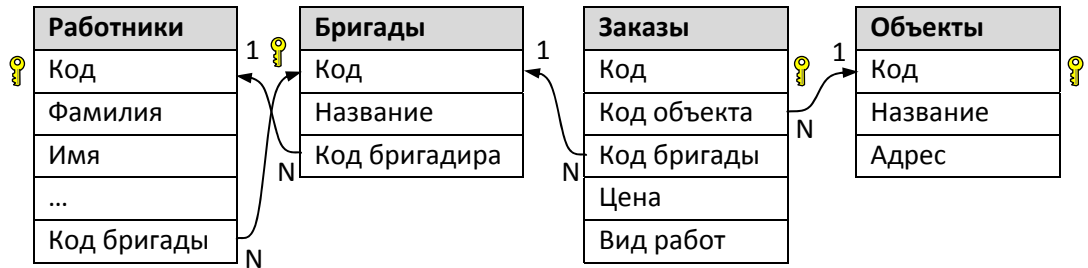
- 1) 2
- 2) а) Москва: 20 уп. бумаги
Санкт-Петербург: 25 уп. фломастеров
Пермь: 20 уп. бумаги, 30 уп. пеналов
Воронеж: –
Липецк: –
б) Москва: 20, Пермь: 20
в) Москва: 20 уп. бумаги = 3000 рублей
Санкт-Петербург: 25 уп. фломастеров = 8750 рублей
Пермь: 20 уп. бумаги + 30 уп. пеналов = 10500 рублей
- 3) а) Фукс Т.Н.
б) Фукс Н.А.
в) Черненко А.Н.
г) Иванов Т.М.
- 4) а) Сидоров А.Т.
б) Сизых А.И., Малых И.И.
в) Симоняк Т.Н.
г) Сизых И.Т.
- 5) а) Левитина Р.Б., Левитин Л.Б., Мурина С.А., Кузнецов П.А.
б) родная сестра П.А. Кузнецова – Мурина С.А. (см. [опечатку](#) в условии)
в) Кузнецов П.А.
г) Косарева Л.П.
- 6)



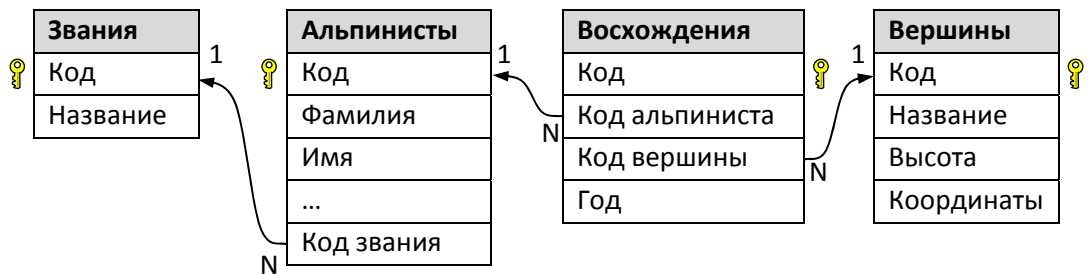
7)



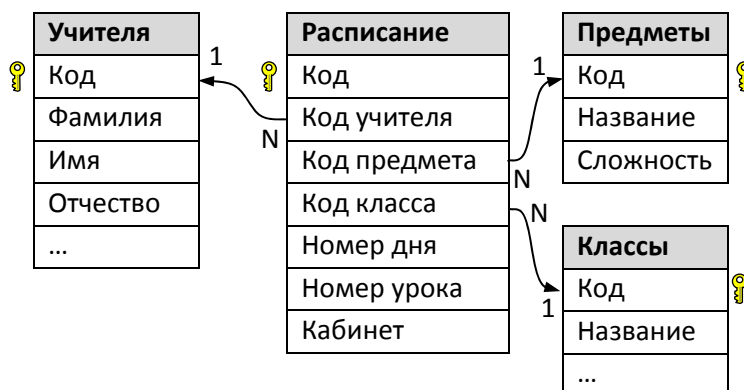
8)



9)

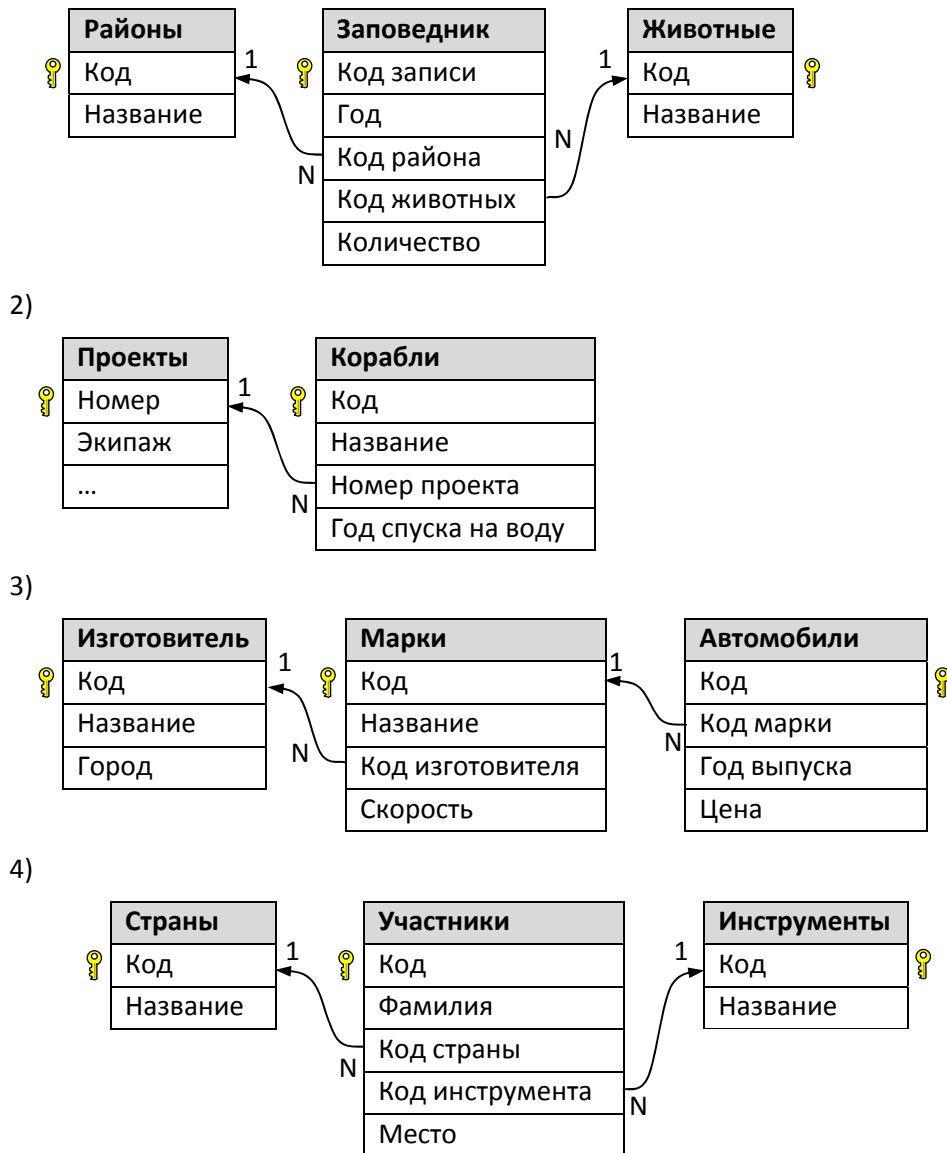


10)



§ 15. Реляционная модель данных

1)



§ 16. Работа с таблицей

- 1) Фильтры б) и в) совпадают.
- 2) а) 0
б) 2
в) 1
г) 2
д) 3
е) 5
ж) 5
з) 3
- 3) а) 1
б) 6
в) 1
г) 3
д) 2
е) 1

§ 17. Создание однотабличной базы данных

Задач нет.

§ 18. Запросы

Задач нет.

§ 19. Формы

Задач нет.

§ 20. Отчёты

Задач нет.

§ 21. Работа с многотабличной базой данных

Задач нет.

§ 22. Нереляционные базы данных

Задач нет.

§ 23. Экспертные системы

Задач нет.

Ответы к задачам

Глава 4. Создание веб-сайтов

§ 24. Веб-сайты и веб-страницы

Задач нет.

§ 25. Текстовые веб-страницы

1)

`index.htm:`

```
<a href="goods/goods.htm">...</a>
<a href="goods/food/food.htm">...</a>
<a href="goods/show/show.htm">...</a>
<a href="history/hist.htm">...</a>
<a href="history/before/before.htm">...</a>
```

`goods.htm:`

```
<a href="../index.htm">...</a>
<a href="food/food.htm">...</a>
<a href="show/show.htm">...</a>
<a href="../history/hist.htm">...</a>
<a href="../history/before/before.htm">...</a>
```

`food.htm:`

```
<a href="../../index.htm">...</a>
<a href="../goods.htm">...</a>
<a href="../show/show.htm">...</a>
<a href="../../history/hist.htm">...</a>
<a href="../../history/before/before.htm">...</a>
```

`show.htm:`

```
<a href="../../index.htm">...</a>
<a href="../goods.htm">...</a>
<a href="../food/food.htm">...</a>
<a href="../../history/hist.htm">...</a>
<a href="../../history/before/before.htm">...</a>
```

`hist.htm:`

```
<a href="../index.htm">...</a>
<a href="../goods/goods.htm">...</a>
<a href="../goods/food/food.htm">...</a>
<a href="../goods/show/show.htm">...</a>
<a href="before/before.htm">...</a>
```

`before.htm:`

```
<a href="../../index.htm">...</a>
<a href="../../goods/goods.htm">...</a>
<a href="../../goods/food/food.htm">...</a>
<a href="../../goods/show/show.htm">...</a>
<a href="../hist.htm">...</a>
```

- 2) а) откроется начало страницы
- б) откроется страница на метке `resume`
- в) начнётся скачивание файла
- г) запустится почтовый клиент

3) –

§ 26. Оформление документа

Задач нет.

§ 27. Рисунки

Задач нет.

§ 28. Мультимедиа

Задач нет.

§ 29. Таблицы

Задач нет.

§ 30. Блоки

Задач нет.

§ 31. XML и XHTML

Задач нет.

§ 32. Динамический HTML

Задач нет.

§ 33. Размещение веб-сайтов

- 1) –
- 2) нельзя, трафики составит около 1,2 Гбайт
- 3) –
- 4) –

Ответы к задачам

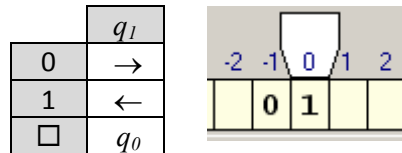
Глава 5. Элементы теории алгоритмов

Программное обеспечение:

- машина Тьюринга: <http://kpolyakov.spb.ru/prog/turing.htm>
- машина Поста: <http://kpolyakov.spb.ru/prog/post.htm>
- алгоритмы Маркова: <http://kpolyakov.spb.ru/prog/nma.htm>

§ 34. Уточнение понятия алгоритма

- 1) а) каретка идет влево до первого пробела и останавливается на ячейке справа от него закичивается, если слово (из 0 и 1) не заканчивается
 б) каретка идет влево до первого непробельного символа и останавливается справа от него закичивается, если слева от каретки только пробелы
 в) каретка идет влево до первого непробельного символа и стирает слово до следующего пробела; закичивается, если слева от каретки только пробелы
- 2) программа, которая закичивается:



- 3) Уменьшение двоичного числа на 1. Каретка над числом.

Уменьшить число, записанное в двоичной системе счисления, на 1. Каретка стоит над числом.

← -16 -15 -14 -13 -12 -11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4 5 6 7 8 →

1 0 1 1 1

Алфавит [01]

	Q_1	Q_2	Q_3	Q_4
0	0 → Q_1	1 ← Q_2	0 ↓ Q_3	— → Q_4
1	1 → Q_1	0 ← Q_3	1 ↓ Q_3	1 ↓ Q_3
—	— ← Q_2	— ↓ Q_3	— → Q_4	— ↓ Q_4

Комментарий
 Q_1 - движение вправо до конца числа
 Q_2 - уменьшение числа на 1.
 Q_3 - проверка лидирующего нуля
 Q_4 - удаление лидирующих нулей

- 4) В десятичной системе:

Увеличение на 1

Уменьшение на 1

← -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4 5 6 7 8 →

9 9 9

Алфавит [0123456789]

	Q_1	Q_2
0	0 → Q_1	1 ↓ Q_2
1	1 → Q_1	2 ↓ Q_2
2	2 → Q_1	3 ↓ Q_2
3	3 → Q_1	4 ↓ Q_2
4	4 → Q_1	5 ↓ Q_2
5	5 → Q_1	6 ↓ Q_2
6	6 → Q_1	7 ↓ Q_2
7	7 → Q_1	8 ↓ Q_2
8	8 → Q_1	9 ↓ Q_2
9	9 → Q_1	0 ← Q_2
—	— ← Q_2	1 ↓ Q_2

Комментарий
 Q_1 - движение вправо до конца числа
 Q_2 - увеличение числа на 1.

← -12 -11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4 5 6 7 8 9 10 11 12 →

2 1 0 0 0

Алфавит [0123456789]

	Q_1	Q_2	Q_3	Q_4
0	0 → Q_1	9 ← Q_2	0 ↓ Q_3	— → Q_4
1	1 → Q_1	8 ← Q_3	1 ↓ Q_3	1 ↓ Q_3
2	2 → Q_1	7 ↓ Q_3	2 ↓ Q_3	2 ↓ Q_3
3	3 → Q_1	6 ↓ Q_3	3 ↓ Q_3	3 ↓ Q_3
4	4 → Q_1	5 ↓ Q_3	4 ↓ Q_3	4 ↓ Q_3
5	5 → Q_1	4 ↓ Q_3	5 ↓ Q_3	5 ↓ Q_3
6	6 → Q_1	3 ↓ Q_3	6 ↓ Q_3	6 ↓ Q_3
7	7 → Q_1	2 ↓ Q_3	7 ↓ Q_3	7 ↓ Q_3
8	8 → Q_1	1 ↓ Q_3	8 ↓ Q_3	8 ↓ Q_3
9	9 → Q_1	0 ← Q_3	9 ↓ Q_3	9 ↓ Q_3
—	— ← Q_2	— ↓ Q_3	— → Q_4	— ↓ Q_4

Комментарий
 Q_1 - движение вправо до конца числа
 Q_2 - уменьшение числа на 1.
 Q_3 - проверка на лидирующий 0
 Q_4 - стереть лидирующие нули

5) Сложение двоичных чисел:

Алфавит 01+

	Q ₁	Q ₂	Q ₃	Q ₄	Q ₅
0	1 ← Q ₁	0 ← Q ₂	1 → Q ₄	0 → Q ₄	
1	0 ← Q ₂	1 ← Q ₂	0 ← Q ₃	1 → Q ₄	_ → Q ₅
+	_ → Q ₅	+ ← Q ₃		+ → Q ₄	
⌊			1 → Q ₄	_ ← Q ₁	_ ↓ ⓧ

Комментарий

Q1 - вычитание 1 из B
 Q2 - находим "+" и становимся на младший разряд A
 Q3 - прибавление 1 к A
 Q4 - находим пробел после примера и возвращаемся на младший разряд B
 Q5 - стираем остатки числа B

Вычитание двоичных чисел:

Алфавит 01-

	Q ₁	Q ₂	Q ₃	Q ₄	Q ₅	Q ₆	Q ₇
0	1 ← Q ₁	0 ← Q ₂	1 ← Q ₃	0 → Q ₆	_ → Q ₅	0 → Q ₆	
1	0 ← Q ₂	1 ← Q ₂	0 ← Q ₄	1 → Q ₆	1 → Q ₆	1 → Q ₆	_ → Q ₇
-	_ → Q ₇	- ← Q ₃			- → Q ₆	- → Q ₆	
⌊			_ → Q ₆	_ → Q ₅		_ ← Q ₁	_ ↓ ⓧ

Комментарий

Q1 - вычитание 1 из B
 Q2 - находим "-" и становимся на младший разряд A
 Q3 - вычитание 1 из A
 Q4 - проверка лидирующего нуля
 Q5 - стираем лидирующий 0
 Q6 - находим пробел после примера и возвращаемся на младший разряд B
 Q7 - стираем остатки числа B

6) Сложение десятичных чисел

Алфавит 0123456789+

	Q ₁	Q ₂	Q ₃	Q ₄	Q ₅
0	9 ← Q ₁	0 ← Q ₂	1 → Q ₄	0 → Q ₄	
1	0 ← Q ₂	1 ← Q ₂	2 ← Q ₄	1 → Q ₄	_ → Q ₅
2	1 ← Q ₂	2 ← Q ₂	3 ← Q ₄	2 → Q ₄	_ → Q ₅
3	2 ← Q ₂	3 ← Q ₂	4 ← Q ₄	3 → Q ₄	_ → Q ₅
4	3 ← Q ₂	4 ← Q ₂	5 ← Q ₄	4 → Q ₄	_ → Q ₅
5	4 ← Q ₂	5 ← Q ₂	6 ← Q ₄	5 → Q ₄	_ → Q ₅
6	5 ← Q ₂	6 ← Q ₂	7 ← Q ₄	6 → Q ₄	_ → Q ₅
7	6 ← Q ₂	7 → Q ₂	8 ← Q ₄	7 → Q ₄	_ → Q ₅
8	7 ← Q ₂	8 → Q ₂	9 ← Q ₄	8 → Q ₄	_ → Q ₅
9	8 ← Q ₂	9 → Q ₂	0 ← Q ₃	9 → Q ₄	_ → Q ₅
+	_ → Q ₅	+ ← Q ₃		+ → Q ₄	
⌊			1 → Q ₄	_ ← Q ₁	_ ↓ ⓧ

Комментарий

Q1 - вычитание 1 из B
 Q2 - находим "+" и становимся на младший разряд A
 Q3 - прибавление 1 к A
 Q4 - находим пробел после примера и возвращаемся на младший разряд B
 Q5 - стираем остатки числа B

Вычитание десятичных чисел

	Q ₁	Q ₂	Q ₃	Q ₄	Q ₅	Q ₆	Q ₇
0	9 ← Q ₁	0 ← Q ₂	9 ← Q ₃	0 → Q ₄	→ Q ₅	0 → Q ₆	
1	0 ← Q ₂	1 ← Q ₂	0 ← Q ₄	1 → Q ₆	1 → Q ₆	1 → Q ₆	
2	1 ← Q ₂	2 ← Q ₂	1 ← Q ₄	2 → Q ₆	2 → Q ₆	2 → Q ₆	
3	2 ← Q ₂	3 ← Q ₂	2 ← Q ₄	3 → Q ₆	3 → Q ₆	3 → Q ₆	
4	3 ← Q ₂	4 ← Q ₂	3 ← Q ₄	4 → Q ₆	4 → Q ₆	4 → Q ₆	
5	4 ← Q ₂	5 ← Q ₂	4 ← Q ₄	5 → Q ₆	5 → Q ₆	5 → Q ₆	
6	5 ← Q ₂	6 ← Q ₂	5 ← Q ₄	6 → Q ₆	6 → Q ₆	6 → Q ₆	
7	6 ← Q ₂	7 ← Q ₂	6 ← Q ₄	7 → Q ₆	7 → Q ₆	7 → Q ₆	
8	7 ← Q ₂	8 ← Q ₂	7 ← Q ₄	8 → Q ₆	8 → Q ₆	8 → Q ₆	
9	8 ← Q ₂	9 ← Q ₂	8 ← Q ₄	9 → Q ₆	9 → Q ₆	9 → Q ₆	→ Q ₇
-	→ Q ₇	← Q ₃			→ Q ₆	→ Q ₆	
←			→ Q ₆	→ Q ₅		← Q ₁	

Алфавит 0123456789-

Комментарий

Q1 - вычитание 1 из B
 Q2 - находим "-" и становимся на младший разряд A
 Q3 - вычитание 1 из A
 Q4 - проверка лидирующего нуля
 Q5 - стираем лидирующий 0
 Q6 - находим пробел после примера и возвращаемся на младший разряд B
 Q7 - стираем остатки числа B

- 7) а) каретка идет вправо и ставит метки на ленте через одну, зацикливается
 б) каретка идет вправо и ставит метки во всех ячейках, пока не дойдёт до существующей метки
 в) каретка идет вправо до первой свободной ячейки и ставит там метку, затем останавливается
- 8) Увеличение числа на 1 Уменьшение числа на 1

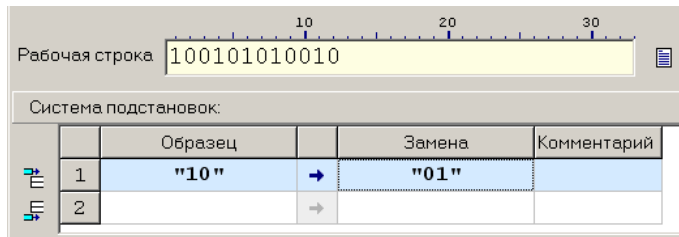
	Команда	Переход	Комментарий
1	<		
2	?	1, 3	
3	>		
4	1		
5	.		

	Команда	Переход	Комментарий
1	>		
2	?	3, 1	
3	<		
4	0		
5	.		

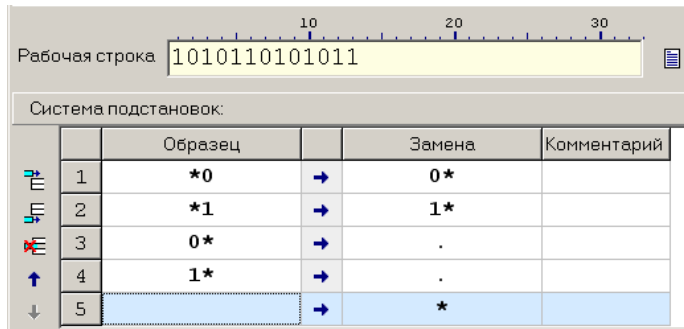
- 9) Сложение чисел в унарной системе

	Команда	Переход	Комментарий
1	<		вышли на первое слагаемое
2	?	7, 3	если там ничего нет, выход
3	0		стереть метку
4	>		перейти на пустую секцию
5	1		поставить метку
6	<	1	перейти на пустую секцию и начать заново
7	!		УРА!

- 10) а) бесконечно увеличивает количество нулей на месте первого нуля или (если нулей нет) бесконечно) добавляет единицы
 б) прогоняет маркер «*» в конец строки и ставит там знак «=»
 в) удваивает каждую цифру
- 11) Отсортировать цифры двоичного числа

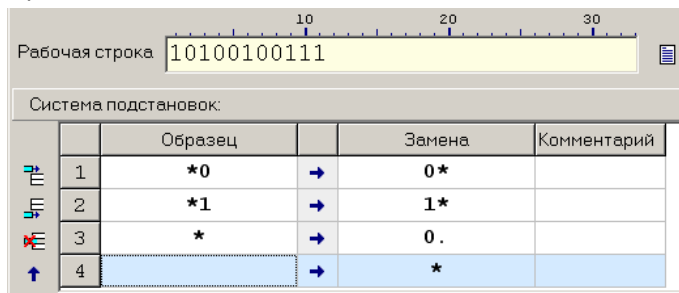


12) Удалить из двоичной строки последний символ (см. [опечатку](#) в условии):



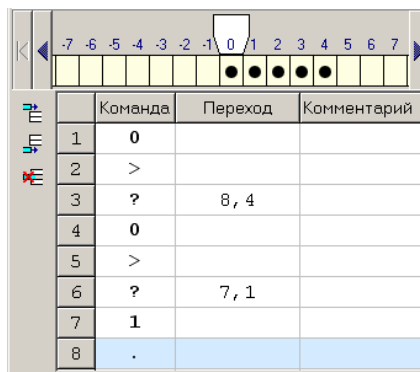
Алгоритм выполняет деление числа на 2 с отбрасыванием остатка.

13) Приписать 0 в конец числа

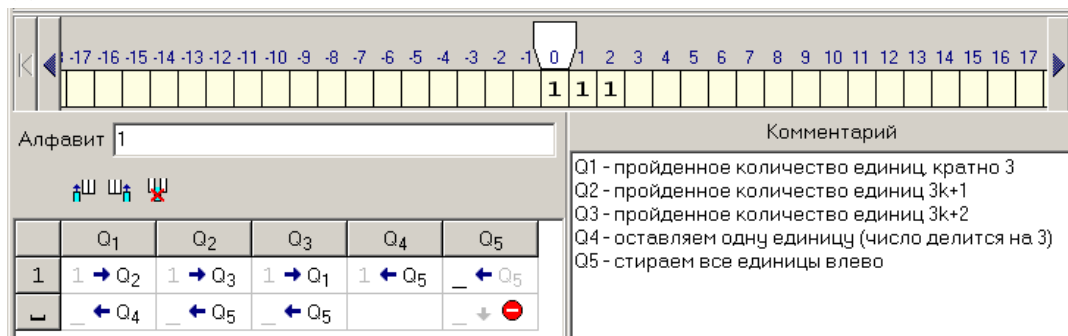


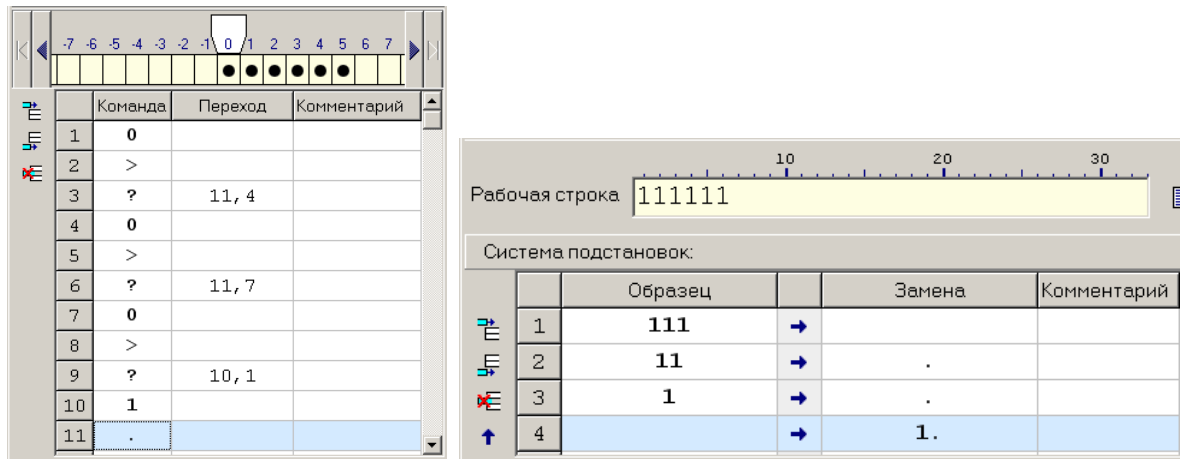
§ 35. Алгоритмически неразрешимые задачи

1) Программа оставляет на ленте одну метку, если меток было число нечётное, и стирает все метки, если это число чётное



2) Делимость на 3:





- 3) Предположим, что проблема останова разрешима, то есть существует программа Z, которая по любой программе P и её исходным данным X определяет, остановится ли эта программа. Тогда, подав на вход этой программы Z её собственный код, мы смогли бы определить, останавливается ли она, то есть получилось бы, что проблема самоприменимости разрешима. Получили противоречие, поэтому проблема останова алгоритмически неразрешима.

§ 36. Сложность вычислений

- 1) а) поиск делителей числа

```

цел N = 15, i
нц для i от 1 до N
  если mod(N,i)=0 то
    вывод i, нс
  все
кц

```

в цикле N шагов, число сравнений и вычисления остатка от деления

$$T_c(N) = T_{\text{mod}}(N) = N - 2$$

асимптотическая сложность $O(N)$

- б) нахождение минимального и максимального элемента

```

цел M, m
M:=A[1]; m:=A[1]
нц для i от 2 до N
  если A[i]>M то M:=A[i] все
  если A[i]<m то m:=A[i] все
кц

```

в цикле N шагов, число сравнений $T_c(N) = 2(N - 1)$

в худшем случае число присваиваний $T_a(N) = N - 1$ (два условных оператора на одном шаге цикла сработать не могут!)

асимптотическая сложность $O(N)$

- в) определение количества положительных элементов массива

```

нц для i от 1 до N
  если A[i]>0 то
    count:=count+1
  все
кц

```

в цикле N шагов, число сравнений $T_c(N) = N - 2$

в худшем случае число присваиваний $T_a(N) = N - 2$

асимптотическая сложность $O(N)$

г) проверка числа на простоту

```

цел N = 23, i
лог простое = да
нц для i от 2 до N-1
  если mod(N,i)=0 то
    простое:=нет
  выход
все
кц

```

в цикле N-2 шага, в худшем случае число сравнений и вычисления остатка от деления

$$T_c(N) = T_{\text{mod}}(N) = N - 2$$

асимптотическая сложность $O(N)$

2) Один из простых алгоритмов (без сохранения промежуточных результатов в массиве)

```

цел i, j, count
лит s = "1234512323"
нц для i от 1 до длин(s)
  count:=0
  нц для j от длин(s) до i+1 шаг -1
    если s[j]=s[i] то
      count:=count+1
      удалить(s, j, 1)
    все
  кц
  если count>0 то
    вывод s[i], нс
  все
кц

```

N = длина строки

Количество шагов внутреннего цикла в худшем случае (когда все символы разные)

$$T(N) = (N-1) + (N-2) + \dots + 2 + 1 = \frac{N(N-1)}{2} = \frac{1}{2}N^2 - \frac{1}{2}N$$

$$\text{Число сравнений } T_c(N) = \frac{1}{2}N^2 - \frac{1}{2}N + N = \frac{1}{2}N^2 + \frac{1}{2}N$$

Количество срабатываний условного оператора в худшем случае (все символы одинаковые)

$$T_{if}(N) = N - 1$$

$$\text{Число присваиваний } T_a(N) = N - 1 + N = 2N - 1$$

Асимптотическая сложность $O(N^2)$

3) Рекурсивный перебор

```

алг Рекурсивный перебор
нач
  лит word = '...'; | из n символов
  TumbaWords('ЫШЧО', word, 0) | алфавит из k символов
кон
алг TumbaWords(лит A, w0, цел N)
нач
  если N = длин(w0) то | слово построено
    вывод w0, нс

```

```

        выход
    все
    цел i
    лит w
    w := w0
    нц для i от 1 до длин(A)
        w[N+1] := A[i]
        TumbaWords(A, w, N+1) | рекурсия
    кц
    кон

```

Асимптотическая сложность $O(k^n)$

§ 37. Доказательство правильности программ

$$1) c := L + \text{div}(R-L, 2) = \text{div}(2L, 2) + \text{div}(R-L, 2) = \text{div}(2L-R-L, 2) = \text{div}(L+R, 2)$$

Вариант $\text{div}(L+R, 2)$ включает меньше операций, но возможно переполнение при сложении.

$$2) \text{ Контрпример: } a=3, b=1, c=2$$

Исправление программы:

```

M := a
если b > a то M := b все
если c > M то M := c все

```

$$3) \text{ Программа некорректная, контрпример: } a=2, b=1, c=3$$

Переменная M будет неопределённой при нарушении всех условий

$$a > b, b > c \text{ и } c > a,$$

то есть при выполнении условия $a \leq b \leq c \leq a$; это возможно только в случае равенства значений всех переменных

4)

```

если a > b то поменять (a, b) все | отсортировать a и b
если b > c то поменять (b, c) все | c = max(a, b, c)
если a > b то поменять (a, b) все | a = min(a, b, c)

```

5) Инвариант: равное количество камней в обеих кучках. Если сначала в кучках разное количество камней, то выигрывает первый, иначе – второй.

6) Инвариант: по крайней мере, один нужный элемент массива находится в диапазоне индексов $[L, R-1]$. Программа находит номер последнего элемента, равного X (он будет записан в переменную L). Для того, чтобы найти первый элемент, равный X, нужно изменить программу, например, так:

```

L := 0; R := n
нц пока L < R-1
    c := div(L+R, 2)
    если X <= A[c] то
        R := c
    иначе
        L := c
    все
кц
вывод R

```

Инвариант цикла: по крайней мере, один нужный элемент массива находится в диапазоне индексов $[L+1, R]$.

- 7) а) инвариант цикла $b = a^k$, вычисляется a^n
б) инвариант цикла $b = a^{n-k}$, вычисляется a^n
- 8) а) $Q: n > 0; R: S =$ сумма делителей числа n
б) $Q: n > 0; R: F =$ «да», если n – простое число; $F =$ «нет», если n – составное число
в) $Q: S$ – непустая строка; $R: n =$ количество слов в строке S
г) $Q: n > 0$, массив A отсортирован по неубыванию;
 $R: L =$ номер элемента массива, равного X
д) $Q: n > 0; R:$ элементы массива A переставлены в обратном порядке
е) $Q: S$ – строка, содержащая цифры и, возможно, первый знак «-»; $R: X$ – целое число, записанное в строке S
- 9) при $n > 0$ можно принять $b := a; k := n - 1; p := a;$
- 10) число шагов цикла $T(n) = \log_2 n = m$, асимптотическая сложность $O(\log n)$

Ответы к задачам

Глава 6. Алгоритмизация и программирование

§ 38. Целочисленные алгоритмы

- 1) Один из делителей всегда находится в диапазоне $[2; \sqrt{k}]$, а второй – в диапазоне $[\sqrt{k}; k-1]$; если нет первого, то нет и второго.
- 2) –
- 3) в ячейках массива хранятся 6-значные числа, при умножении на $k \leq 100$ получается число не более, чем из 8 разрядов, что меньше, чем максимально допустимое число для 32-битных переменных $2^{31}-1 = 2\,147\,483\,647$
- 4) нет, так как нужно, чтобы в ячейке массива поместилось значение элемента, умноженного на 100, а в 4-байтовую ячейку помещаются числа, не большие, чем $2\,147\,483\,647$
- 5) 24 нуля
- 6) –
- 7) –
- 8) Один из вариантов процедуры для вывода числа в 6 позициях:

```

алг Write6 (цел x)
нач
  лит s = "000000", s1
  цел xx, k=6
  xx:= x
  нц пока xx > 0
    s1:=цел_в_лит(mod(xx,10))
    s[k]:=s1[1]
    k:=k-1
    xx:=div(xx,10)
  кц
  вывод s
кон

```

- 9) –
- 10) –
- 11) –
- 12) –

§ 39. Структуры (записи)

–

§ 40. Динамические массивы

–

§ 41. Списки

–

§ 42. Стек, очередь, дек

–

§ 43. Деревья

–

§ 44. Графы

- 1) –
- 2) алгоритм Прима-Крускала в приведенном варианте: $O(N^3)$; более эффективные реализации можно найти в литературе
- 3) –
- 4) –
- 5) алгоритм Дейкстры: $O(N^2)$, алгоритм Флойда-Уоршелла: $O(N^3)$
- 6) –
- 7) –

§ 45. Динамическое программирование

- 1) –
- 2) –
- 3) –
- 4) –
- 5) –
- 6) 14
- 7) 22
- 8) –

Ответы к задачам

Глава 7. Объектно-ориентированное программирование

§ 46. Что такое ООП?

–

§ 47. Объекты и классы

–

§ 48. Создание объектов в программе

–

§ 49. Скрытие внутреннего устройства

–

§ 50. Иерархия классов

–

§ 51. Программы с графическим интерфейсом

–

§ 52. Основы программирования в RAD-средах

–

§ 53. Использование компонентов

–

§ 54. Совершенствование компонентов

–

§ 55. Модель и представление

–

Ответы к задачам

Глава 8. Компьютерная графика и анимация

§ 56. Основы растровой графики

- 1) на экране (96 ppi): 5,29 x 2,65 см
на печати (300 ppi): 1,69 x 0,85 см
- 2) 3508 x 2480 пикселей; 8,7 Мпикселей

§ 57. Ввод изображений

Задач нет.

§ 58. Коррекция фотографий

Задач нет.

§ 59. Работа с областями

Задач нет.

§ 60. Фильтры

Задач нет.

§ 61. Многослойные изображения

Задач нет.

§ 62. Каналы

Задач нет.

§ 63. Иллюстрации для веб-сайтов

Задач нет.

§ 64. Анимация

Задач нет.

§ 65. Контуры

Задач нет.

Ответы к задачам

Глава 9. Трёхмерная графика.

§ 66. Введение

Задач нет.

§ 67. Работа с объектами

Задач нет.

§ 68. Сеточные модели

Задач нет.

§ 69. Модификаторы

Задач нет.

§ 70. Кривые

Задач нет.

§ 71. Материалы и текстуры

Задач нет.

§ 72. Рендеринг

Задач нет.

§ 73. Анимация

Задач нет.

§ 74. Язык VRML

Задач нет.