

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ТОРГОВО-ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра информационных систем и информационных технологий

Павлов В.Н.

**Задания для контрольных работ по дисциплине
«Информационные технологии и защита
информации»**

Санкт-Петербург 2014

Литература.....	2
1. Задания для контрольных работ теории и методам кодирования информации.....	2
1.1. Методические материалы. Основные определения.....	2
1.1.1. Пример 1 Вычисление энтропии источника.....	3
1.1.2. Пример 2 (продолжение примера 1) Равномерный код.....	4
1.1.3. Основные этапы кодирования Шеннона-Фано.....	4
1.1.4. Пример 3 (продолжение примера 2) Код Шеннона-Фано.....	5
1.1.5. Пример 4 (продолжение примера 3) Код Хаффмана.....	6
1.2. Текст задания.....	9

Литература

Павлов В.Н., Смородина Е.В., Трусков С.С. Информационные технологии и защита информации. Учебно-методическое пособие. СПбГТЭУ, 2014.

1. Задания для контрольных работ теории и методам кодирования информации

1.1. Методические материалы. Основные определения

Пусть имеется алфавит $A = \{a_1, \dots, a_m\}$, в котором нумерация букв задает некоторый порядок, и текст $T_n = b_1 b_2 \dots b_n$ длины n , такой что $b_s \in A$ для всякого $s = 1, \dots, n$, т.е. для всякого $s = 1, \dots, n$ найдется $k(s) : 1 \leq k(s) \leq m$ такое, что $b_s = a_{k(s)}$. Длину текста T будем обозначать $|T|$, так что $|T_n| = n$. Справедливо равенство: $T_n = T_{n-1} b_n$, где $T_{n-1} = b_1 b_2 \dots b_{n-1}$. Через $e = T_0$ обозначим пустой (не содержащий ни одной буквы) текст. Обозначим через A^* множество всевозможных текстов в алфавите A , имеющих конечные длины $|T| < \infty$.

Каждый текст $T \in A^*$ в теории информации иногда называется словом или сообщением.

Источником сообщений называется пара (A, p) из конечного алфавита A и распределения вероятностей $p(T)$ на множестве A^* конечных слов в алфавите A . Распределение вероятностей $p(T)$ должно удовлетворять естественным соотношениям:

1) для любого текста (слова) $T \in A^*$ справедливо равенство

$$\sum_{a \in A} p(Ta) = p(T);$$

2) для любого целого положительного числа n справедливо равенство

$$\sum_{|T|=n} p(T) = 1.$$

Скоростью создания информации источником называется средняя длина сообщения, создаваемого источником в единицу времени. Обозначим скорость создания информации источником через V .

Источник сообщений называется источником без памяти (источником Бернулли), если вероятность буквы в слове не зависит от контекста, т. е.

$$p(b_1 b_2 \dots b_n) = p(b_1) \cdot p(b_2) \cdot \dots \cdot p(b_n).$$

Величина

$$I(T) = -\log_2 p(T)$$

называется сложностью слова $T \in A^*$ относительно источника сообщений (A, p) или количеством информации, содержащемся в слове T .

Теперь ожидаемая сложность наугад выбранного слова длины n вычисляется:

$$H_n(A, p) = -\sum_{|T|=n} p(T) \cdot \log_2 p(T)$$

и называется энтропией слова длины n относительно источника сообщений (A, p) или ожидаемым количеством информации в наугад выбранном n -буквенном слове T , измеряемом в битах. В частности, энтропия слова длины 1 называется энтропией источника Бернулли (A, p) и обозначается $H(A, p)$. Энтропия источника характеризует ожидаемое количество информации, содержащееся в наугад выбранной букве алфавита в битах.

1.1.1. Пример 1 Вычисление энтропии источника

Вычислить энтропию источника без памяти в битах, если на выходе этого источника получено слово

$T = \text{ebgceaeagaggaecsgcesgaaeaeegbaccsescasegscgeagbfaeac}$

и энтропию этого слова T .

Решение.

- 1) Вычисляем длину полученного слова $|T| = 45$.
- 2) Заполняем следующую таблицу

Буква алфавита	Сколько раз встречается в слове T	Относительная частота
a	11	$P_1 = 11/45 = 0,245$
b	3	$P_2 = 3/45 = 0,067$
c	9	$P_3 = 9/45 = 0,2$
e	11	$P_4 = 11/45 = 0,245$
f	1	$P_5 = 1/45 = 0,021$
g	10	$P_6 = 10/45 = 0,222$

- 3) Вычисляем энтропию источника (A, p) по формуле (2.5) (при $n = 1$):

$$\begin{aligned} H(A, p) &= -0,245 \log_2 0,245 - 0,067 \log_2 0,067 - 0,2 \log_2 0,2 - \\ &- 0,245 \log_2 0,245 - 0,021 \log_2 0,021 - 0,222 \log_2 0,222 = \\ &= 0,245 \cdot 2,032 + 0,067 \cdot 3,907 + 0,2 \cdot 2,322 + 0,245 \cdot 2,032 + \\ &\quad + 0,022 \cdot 5,492 + 0,222 \cdot 2,170 = \\ &= 0,497 + 0,260 + 0,464 + 0,497 + 0,122 + 0,482 = 2,323 \end{aligned}$$

- 4) Вычисляем энтропию слова T : $H_{45}(A, p) = 45 \cdot H(A, p) = 104,522$ битов.

1.1.2. Пример 2 (продолжение примера 1) Равномерный код

Определение. Код называется равномерным, если длина кода каждой буквы алфавита одинакова.

Для слова $T = \text{ebgcesaegaggaecsgesgaaeaegebacссесасесгсгеагбфаес}$ из примера 1 определить длину оптимального однобуквенного равномерного кода $k(1)$, определить длину оптимального равномерного кода слова T , вычислить информационную избыточность оптимального равномерного кода слова T .

Решение. Так как $N(1)$ представляет собой количество различных букв в слове T , то, согласно составленной в примере 1 таблице, $N(1) = 6$. Целое число $k(1)$ находится из условий: $\log_2 N(1) \leq k(1) < \log_2 N(1) + 1$, т.е. $2,58 \leq k(1) < 2,58 + 1$. Этим условиям удовлетворяет единственное целое число $k(1) = 3$. Следовательно, длина равномерного кода каждой буквы слова T равна 3 бита. Так как в слове T всего 45 букв, то длина равномерного кода $|f(T)|$ слова T равна $3 \cdot 45 = 135$ битов, т.е. $|f(T)| = 135$.

Энтропия источника, вычисленная в примере 1, равна $H(A, p) = 2,323$. Таким образом, информационная избыточность равномерного кода равна:

$$S(f, T) = \left(1 - \frac{H(A, p) \cdot |T|}{|f(T)|} \right) \cdot 100 = \left(1 - \frac{2,323 \cdot 45}{135} \right) \cdot 100 = 22,576\%$$

1.1.3. Основные этапы кодирования Шеннона-Фано

1. Строится таблица частот встречаемости (вероятностей) символов алфавита в кодируемом слове.
2. Символы алфавита упорядочиваются по убыванию частот.
3. Символы алфавита A делят на две части A_0 и A_1 (левая A_0 и правая A_1) так, чтобы суммарные вероятности символов, вошедших в A_0 были максимально близки суммарной вероятности символов, вошедших в A_1 .
4. Каждому символу алфавита A , вошедшему в подмножество A_0 , назначается первая цифра двоичного кода «0», а вошедшему в подмножество A_1 — первая цифра двоичного кода «1».
5. Каждое из полученных подмножеств алфавита A_0 и A_1 снова делится на две части (левая и правая) по алгоритму шага 3. Пусть подмножество A_0 делится на A_{00} и A_{01} , а подмножество A_1 — на A_{10} и A_{11} . При этом, если подмножество A_0 или подмножество A_1 уже состоит из одного символа, то дальнейшее деление этого подмножества не происходит, так как единственный символ этого подмножества уже получил окончательный двоичный код.
6. Теперь каждому символу алфавита, вошедшему в A_{00} , назначаются первые две цифры двоичного кода «00», символам из A_{01} — цифры «01», символам из A_{10} — цифры «10», символам из A_{11} — цифры «11».
7. Выполняется дальнейшее деление каждого из полученных подмножеств по алгоритму шага 3. $A_{00} \rightarrow A_{000} + A_{001}$, $A_{01} \rightarrow A_{010} + A_{011}$, $A_{10} \rightarrow A_{100} + A_{101}$,

$A_{11} \rightarrow A_{110} + A_{111}$. При этом, если подмножество уже состоит из одного символа, то дальнейшее деление этого подмножества не происходит, так как единственный символ этого подмножества уже получил окончательный двоичный код на предыдущем шаге.

8. Процесс последовательного деления подмножеств алфавита продолжается до тех пор, пока подмножеств, содержащих несколько элементов, не останется.

1.1.4. Пример 3 (продолжение примера 2) Код Шеннона-Фано

Найдем длину кода Шеннона-Фано для слова

$T = \text{ebgescaggaesgescgaaeaeagbассесасегсгеагbfаес}$

из примера 1.

Решение.

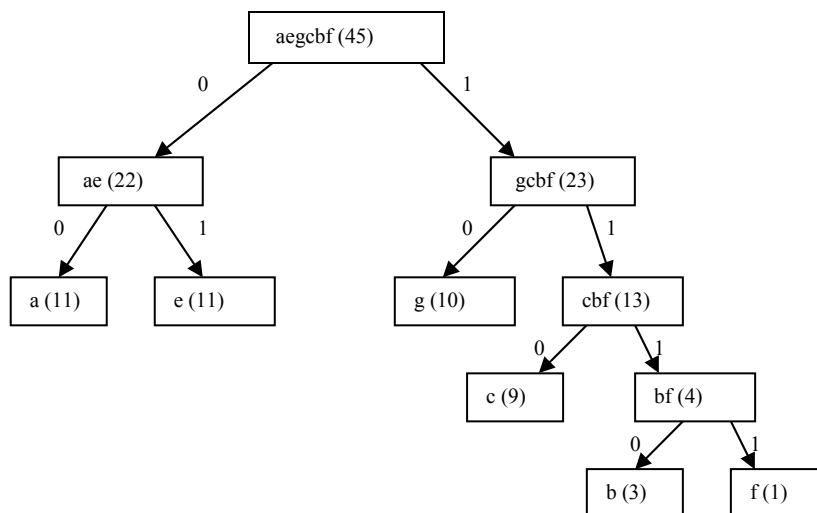
Исходные символы:

Буква	Сколько раз встречается в слове T
a	11
b	3
c	9
e	11
f	1
g	10

Шаг 1. Упорядочиваем таблицу

Буква	Сколько раз встречается в слове T
a	11
e	11
g	10
c	9
b	3
f	1

Построение кодового дерева начинаем от корневой вершины (45). Строим кодовое дерево:



На каждой стрелке дерева написана цифра 0 или 1. Двигаясь от корневого слова (45) до каждой буквы по стрелкам, переписываем последовательно встречающиеся на стрелках цифры.

Результаты сводим в итоговой таблице:

Буква	Полученный код
a	00
b	1110
c	110
e	01
f	1111
g	10

Код Шаннона-Фано слова T равен:

$f(T) = 0111101001110000110001010000111010011101000000100011011100011011001110011001101101001001001001011101111000001110$

Длина кода $f(T)$ равна $107 = 2 \cdot 11 + 4 \cdot 3 + 3 \cdot 9 + 2 \cdot 11 + 4 \cdot 1 + 2 \cdot 10$ битов. Количество битов, израсходованное на одну букву в среднем равно $107/45 = 2,378$ бит. Вывод. Расход битов на букву в коде Шеннона-Фано (2,378) больше энтропии источника, равной 2,323 (см. пример 1), но меньше, чем в равномерном коде. Вспомним из примера 2, что расход битов на букву в равномерном коде был равен 3.

Вычислим информационную избыточность кода Шеннона-Фано.

$$S(f, T) = \left(1 - \frac{H(A, p) \cdot |T|}{|f(T)|} \right) \cdot 100 = \left(1 - \frac{2,323 \cdot 45}{107} \right) \cdot 100 = 2,316\%$$

Информационная избыточность кода Шеннона-Фано (2,316%) меньше информационной избыточности равномерного кода (22,576%), т.е. код Шеннона-Фано более экономичный, чем равномерный код.

1.1.5. Пример 4 (продолжение примера 3) Код Хаффмана

Найдем длину кода Хаффмана для слова
 $T = \text{ebgесаегагааесгесгааеаегбассесасегсгеагбфаес}$
из примера 1.

Решение.

Исходные символы:

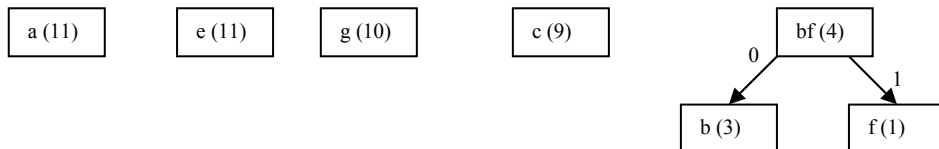
Буква	Сколько раз встречается в слове T
a	11
b	3
c	9
e	11
f	1
g	10

Построение кодового дерева Хаффмана начинаем с нижнего уровня. Завершающим этапом построения кодового дерева является корневая вершина (45). Упорядочим буквы по убыванию частоты и строим кодовое дерево.

Дерево Хаффмана на первом шаге алгоритма имеет вид:

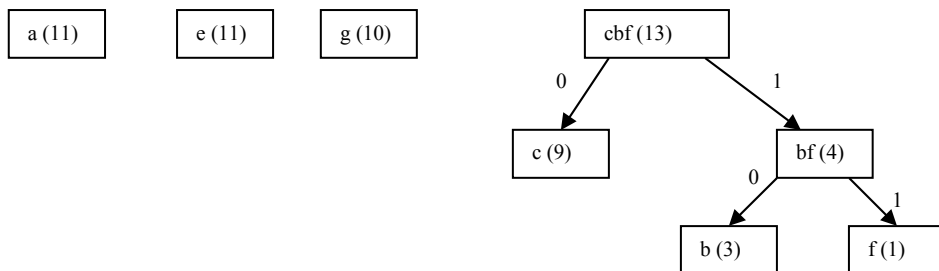


Свободными узлами дерева являются: a(11), e(11), g(10), c(9), b(3), f(1). Выбираем два свободных узла с наименьшими весами b и f и создаем свободный родительский узел {bf} с весом 4=1+3. Исправляем дерево:



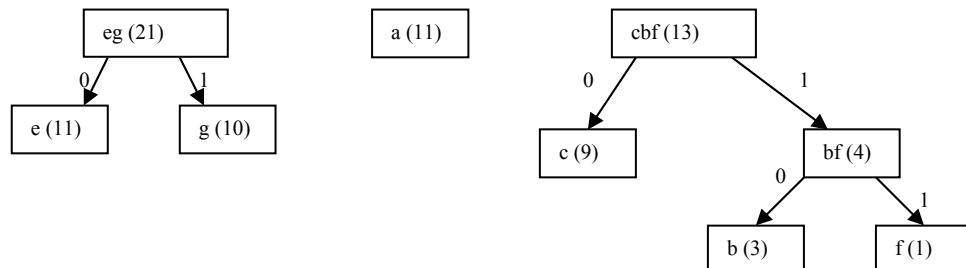
Теперь свободными узлами дерева являются: a(11), e(11), g(10), c(9), bf(4).

Снова выбираем два свободных узла с наименьшими весами: c(9) и bf(4), и строим родительский узел {cbf}. Исправляем дерево:



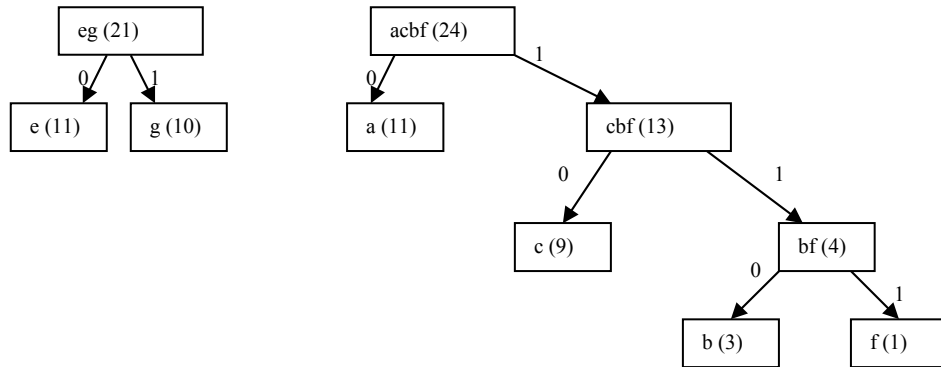
Теперь свободными узлами дерева являются: a(11), e(11), g(10), cbf(13).

Снова выбираем два свободных узла с наименьшими весами: e(11) и g(10), и строим родительский узел {eg}. Исправляем дерево:



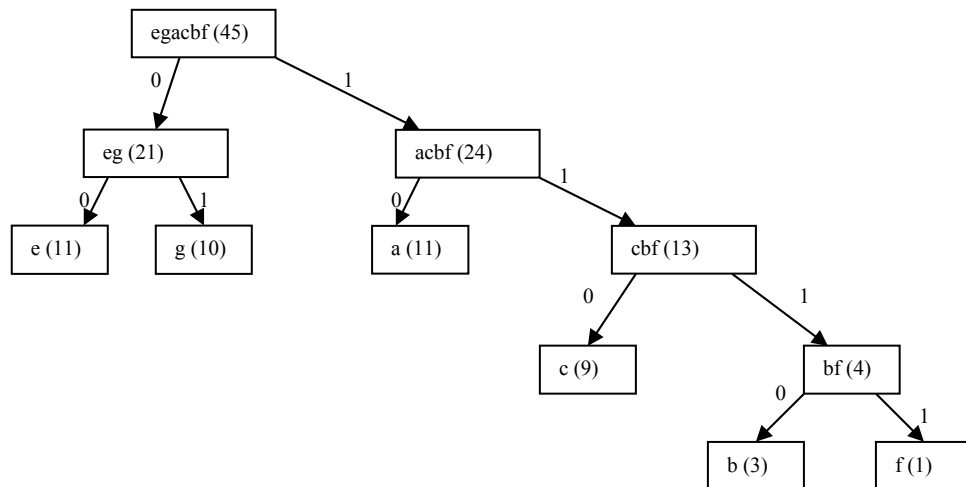
Теперь свободными узлами дерева являются: eg(21), a(11), cbf(13).

Снова выбираем два свободных узла с наименьшими весами: a(11) и cbf(13), и строим родительский узел {acbf}. Исправляем дерево:



Теперь свободными узлами дерева являются: eg(21), acbf(24).

Выбираем два последних свободных узла: eg(21) и acbf(24), и строим родительский узел {egacbf}. Исправляем дерево:



Построение дерева Хаффмана закончено.

Видим, что кодовое дерево Хаффмана отличается от кодового дерева Шеннона-Фано.

Результаты сводим в итоговой таблице начиная от корневого узла (45) также, как это делали в примере 3:

Буква	Полученный код
A	10
B	1110
C	110
E	00
F	1111
G	01

Код Хаффмана слова T равен:

f(T)=001110010011010000110010110001100100110011010001000011110101101100
0110 1011000011100100100111101111101000110

Длина кода f(T) равна 107 битов. Количество битов, израсходованное на одну букву в среднем равно $107/45=2,378$ бита. Вывод. Расход битов на букву в коде Хаффмана (2,378) больше энтропии источника из примера 1, равной 2,323, и равен расходу кода Шеннона-Фано из примера 3 (2,378).

Вычислим коэффициент избыточности кода Хаффмана:

$$S(f, T) = \left(1 - \frac{H(A, p) \cdot |T|}{|f(T)|} \right) \cdot 100 = \left(1 - \frac{2,323 \cdot 45}{107} \right) \cdot 100 = 2,316\%$$

Видим, что коэффициент избыточности кода Хаффмана (2,316%) равен коэффициенту избыточности кода Шеннона-Фано (2,316%) из примера 3 для рассмотренного слова T, т.е. код Хаффмана и код Шеннона-Фано одинаково экономичны.

1.2. Текст задания

Дано слово T, сгенерированное стационарным источником без памяти.

- 1) Вычислить энтропию этого слова (см Пример 1).
- 2) Построить равномерный код этого слова (см. Пример 2).
- 3) Построить код Шеннона-Фано этого слова (см. пример 3).
- 4) Построить код Хаффмана этого слова (см. пример 4).
- 5) Сравнить между собой экономичность каждого кода также, как это сделано в примере 4, и оценить информационную избыточность каждого кода.

Вариант исходного слова T выбирается по последним двум цифрам номера зачетной книжки.

Например. Если последние две цифры в номере Вашей зачетной книжки равны 25, то Вы выбираете слово из варианта 25.

Варианты исходных слов.

Вариант 00

T = bcdijgbidhfiebjiggfbfbjgghifheibjjbcejeie

Вариант 01

T = igcgjcgciicgbhaebjjibjedibecgciafjbgejdg

Вариант 02

T = icddbbgghiddfbgdbdbdedifedeeaaieehgdeigfdea

Вариант 03

T = eггесgecefcgeeeeгсгаегgdeeaeggdcfcсееегgag

Вариант 04

T = ehfffeifbiaiidbidccibdfdfhddfaadhfgddgeg

Вариант 05

T = hddgacфегibibibgacacdiihbbhafbehbhffffhiegad

Вариант 06

T = bbhbbiabbbbgebgagfbaggiabageeabghaaghiebd

Вариант 07

T = hcgibigcchbgaafagbcafbbhiihbicfgccihgiaci

Вариант 08

T = iidfgijddcjdhfhjicciihgccfjagcigedcdjjcid

Вариант 09

T = kgcik kibgkigbkbcihakikggchfchghikjiaaieaikba

Вариант 10

T = fdbagdfbcggfagahbabfbhfefeehedceeeeadecad

Вариант 11

T = fegfbdfbchfigeegggfbgfdgсgeghefibbeaicgcbfe

Вариант 12

T = bfbcfcdfafdfcdcebibefbdafdcfadedddcfeddec

Вариант 13

T = dichfececicbdddabcgbgcdcibchcdgegeddiibcdgg

Вариант 14

T = aikjcgjiiibddaijaebcekakibefkfaaacacfceidac

Вариант 15

T = iaabhgeccbbbfhjfeffhijheccgfhbbccabfciddec

Вариант 16

T = hgcehadaagahfcgbachdehfabcegbchhahccfgba

Вариант 17

T = jgjhjeeidiaddffdjffiehefigaijdgjgiedgaj

Вариант 18

T = ibbcgbbaadbbbbahbabibiaiebfeggdbbcbbebi

Вариант 19

T = jffjdiddjjcdgcigijijfdjdgcfjjdidcdgfdgfb

Вариант 20

T = aifdfhhafehdciiddidciaaddidhdihdhggcfadc

Вариант 21

T = ggedagecefeefhgcccgcchaacgfcbedfgbegddgcgbg

Вариант 22

T = deedeihdeeddbgccdceheeiidddeehgehghgegedg

Вариант 23

T = hcdhdddbfhhaehbedddbaadhaabd hfgddhbegaa

Вариант 24

T = jidkchjhcicdffedjicbicjjbbffdjkc dididjjkh

Вариант 25

T = hdgddddbedddhedfhhdhbgbgghdghdebhdhdbhbgd

Вариант 26

T = ibeifbefchbgbbjjehghcbggjhgiahcegcajcjheh

Вариант 27

T = jgfbijaadigjfhgddggiggficcjfidfcchjdajhgjib

Вариант 28

T = hhcfhjccjbejcgihkhaeccajiafehdihefcgcchc

Вариант 29

T = dhdcahhcdgdgaiabbhgeghhagdgfacadiggbdfeegg

Вариант 30

T = hhhbehgcddbihbdcbibihdjbdhegcadjhhihhjdbhhjh

Вариант 31

T = ifjafjgbjgaejfjjajiifabfcejieiciejgagibjf

Вариант 32

T = jhfdjdjddgfjfeiehajddjdfjecjjjedjdefjfejja

Вариант 33

T = edhhgdecebgfbfgeecdbbghdahbhfchcfdiccahafdch

Вариант 34

T = dccccagfghaidhgdicaaahceggghacchadbacihha

Вариант 35

T = cdgdhhdbeccdehheebhcceecdhaacehadahehchcbcgh

Вариант 36

T = dgdceffgdabgefdhfgdfaedbbeafhhgdgfbddhd

Вариант 37

T = ibadhgehbdhdecbbdhhdccdbdddhhaihddhadciddihed

Вариант 38

T = ehhcfehdhebbahcdbhadibacbigdijhdddajdahadha

Вариант 39

T = gsegjegabgbhgeegbjhgbeeagbccbfjcfaggegcbgc

Вариант 40

T = acgggghagfggchddfdcgfgdbbcfchbccfaaeaefff

Вариант 41

T = fgfjibjifgiffiffijjeigaeehjajaieijaiajfe

Вариант 42

T = ibgeeciaidbbhdcbiibbcdhabdhiachaggdaehdde

Вариант 43

T = daifdaidagidgaidgiaaddaiighbgbagagdiifbddidd

Вариант 44

T = faceceaeffdhbeehieehbhcagbgacgghheheaec

Вариант 45

T = gbfeeicgdhgbfceggbfigbhicfechbdcfghfgbicf

Вариант 46

T = ejedfbidbifgicegafjiceehhhbfccahebbhddajii

Вариант 47

T = ggechhchcceecfebfccbhffecghcecfceegcbeec

Вариант 48

T = aacaehcaadcdcchchbiabaeajgcdahcabgihcdcdhhif

Вариант 49

T = ghhefcbaeagdcfecfcfgcccgegbbffcfahdchgecgcbc

Вариант 50

T = gaedggeahaahbeghhdbehhdihfdgaabffbaeahdcj

Вариант 51

T = deadgbegdgiigbfigiidbeaaededdaeggdegdddb

Вариант 52

T = bfbggaefajbbejfaifbgjijfggffeffifehejjgbeb

Вариант 53

T = cgfbdbcdihcvgfbhedchcddbcgdbceeeefdedicb

Вариант 54

T = ifbdddhhbbeegaeddggeiiidfeaeahaheieaeffd

Вариант 55

T = affbbafbbababadbgbfegaddgcggbcgfbdacbaccaa

Вариант 56

T = ecdcdbbjicfdgaffdcbebijicjehdcehfafefef

Вариант 57

T = bbahiidiaehiaidfeciiiiiedafhadfeeiidfcffefa

Вариант 58

T = egdahgdahcaddheaddagddafnhfefhdefdaehhbeadhed

Вариант 59

T = fiiichiagbjafiaefbaiejbbbjfecbbciabbcbadhaib

Вариант 60

T = igbghahehgigiafgbbbfaiigaibccgbihdvgfvgb

Вариант 61

T = hgdedheehageehedeecdeciegchggghhcgdcddaeeegg

Вариант 62

T = dhfjdjihiefibdcfdddaiiddgfjabggjcaafidfb

Вариант 63

T = dgicijaaifdaifgcajjikbfeadigbdjdibijiaigjgfij

Вариант 64

T = bbhafkfikeybbffbkkihfhdbidjaikdbakabfjkfbd

Вариант 65

T = abjajgfcjiaiejhjihjjjgfcgfbjfbjhjfajahfj

Вариант 66

T = fedchchccecbhbhfcbbgcbgbgfcfcfbfbefecbheb

Вариант 67

T = gbfdibibjfdbjgaabijiiiabfjjifcfgediigie

Вариант 68

T = hedfgfjdaahagcadaddhdhacdadehahgeaceacbgjdh

Вариант 69

T = agjfcfihffbcfbfhjbaifabajjcbdjhajagcgchh

Вариант 70

T = hhhbidhefbgebcihgbdeehfbfhffdihcdbiggibdiibhjh

Вариант 71

T = dhhaahdcddcageaebahffdfgbdgcihdaadeagdgecgehb

Вариант 72

T = ehacfbdheedhcbdhcghbcbfcdfbbagedfdbhfaddbfdh

Вариант 73

T = feadggiajajdajiheadcdhhijafaifeaidfdaiaj

Вариант 74

T = bkdckddkgibkkcgbbegdkdkiiikkigckgkicdfigke

Вариант 75

T = dfahbfhhacadjdddifcjfhbahcgffadahchjfdced

Вариант 76

T = eajaeghgkkhjdggjcegbdckjcebafigjkkcjfggjabha

Вариант 77

T = egfedgdjgjjfeeajeeehcejckcdjcdchicbbicg

Вариант 78

T = bebbccabbfbcbaabcheabaebaehaachbahccgeabbbg

Вариант 79

T = bhfbbaggbfbhgagahagfbaehbeegbifbbgbhafgbghafa

Вариант 80

T = ghhdhghffehhcgffefcgccccgffafachhhfdhhif

Вариант 81

T = ehdddchgehchbhghahcefgfiaacccdddedgddgfcddbc

Вариант 82

T = bffggfghhbhbggcfggfdgbfhffbggghbgbbfhhefbaff

Вариант 83

T = hdgckaiebcccckgagdgddgcccgfifdcedddeeaieccdgj

Вариант 84

T = hbiagijabifjfahgiaijhbjabhahbahjaahihai

Вариант 85

T = beaeddhhaeedheahcbaebcheabeeghegeheda

Вариант 86

T = dfigacbgiiigciiicgeibfbbcibicbeecfedfdigai

Вариант 87

T = affabebhbadhhehdhhdhddhfbbebhshhhhhhhbegbbg

Вариант 88

T = eadgchgaggcgadcgafaagdgaahgdgdgggeddghddefg

Вариант 89

T = bbaceaaecgfcagbefbeaieijcbffbjcefghacjceb

Вариант 90

T = jgbbicfgfjjbicjbgcgiihjhjcggdjjigbcbij

Вариант 91

T = cgaaggbeahacfdihgbchkjaebcahaaceajfageaefhcf

Вариант 92

T = eaccchddefccaaccafaeecbceeahhfecdfeacefehaacc

Вариант 93

T = dddbfcdddegccgdddabecefeechbdeecchbehcachd

Вариант 94

T = hcifaifahaahbcagfahciafbaifhffahffaichch

Вариант 95

T = fibdaaibiajhffhjhjdbgjhhijhdgagaaddjhbdg

Вариант 96

T = ihciagehgegeeiegeeggaggaaagggedigghaggaegbe

Вариант 97

T = hibhchbaabfahiihicbffcbaiahicffebechgicb

Вариант 98

T = cbbgbaiiaaheihahiiifjjghiaaihdciaifbfaafbi

Вариант 99

T = ebcgdgadiaeagiidedcdhhdhdcdbchidaedhaaciieah