Задание 1

Дан набор данных:

tid	itemset
t_1	$ABCD$
t_2	$ACDF$
t_3	$ACDEG$
t_4	$ABDF$
t_5	BCG
t_6	DFG
t_7	ABG
t_8	$CDFG$

* будем считать не supp, а кол-во вхождений

A. Предположив, что минимальный уровень поддержки равен 3 / 8. Продемонстрируйте, как алгоритм Apriori перебирает данный набор данных.

Шаг 1

	A	B	C	D	E	F	G
t1	1	1	1	1	0	0	0
t2	1	0	1	1	0	1	0
t3	1	0	1	1	1	0	1
t4	1	1	0	1	0	1	0
t5	0	1	1	0	0	0	1
t6	0	0	0	1	0	1	1
t7	1	1	0	0	0	0	1
t8	0	0	1	1	0	1	1
	5	4	5	6	1	4	5

Шаг 2

	AB	AC	AD	AF	AG	BC	BD	BF	BG	CD	CF	CG	DF	DG	FG
t1	1	1	1	0	0	1	1	0	0	1	0	0	0	0	0
t2	0	1	1	1	0	0	0	0	0	1	1	0	1	0	0
t3	0	1	1	0	1	0	0	0	0	1	0	1	0	1	0
t4	1	0	1	1	0	0	1	1	0	0	0	0	1	0	0
t5	0	0	0	0	0	1	0	0	1	0	0	1	0	0	0
t6	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
t7	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0
t8	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
	3	3	4	2	2	2	2	1	2	4	2	3	4	3	2

Шаг 3

	ABC	ABD	ACD	CDG	DFG
t1	1	1	1	0	0
t2	0	0	1	0	0
t3	0	0	1	1	0
t4	0	1	0	0	0
t5	0	0	0	0	0
t6	0	0	0	0	1
t7	0	0	0	0	0
t8	0	0	0	1	1
	1	2	3	2	2

B. Предположив, что минимальный уровень поддержки равен 2 / 8. Продемонстрируйте, как алгоритм FPGRowth перебирает данный набор данных.

Шаг 1

	A	B	C	D	E	F	G
t1	1	1	1	1	0	0	0
t2	1	0	1	1	0	1	0
t3	1	0	1	1	1	0	1
t4	1	1	0	1	0	1	0
t5	0	1	1	0	0	0	1
t6	0	0	0	1	0	1	1
t7	1	1	0	0	0	0	1
t8	0	0	1	1	0	1	1
	5	4	5	6	1	4	5

Шаг 2

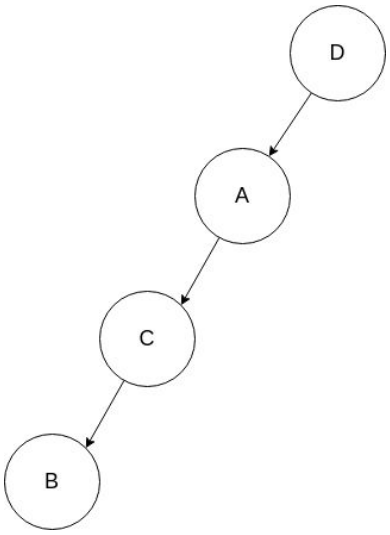
D	6
A	5
C	5
G	5
B	4
F	4

Шаг 3

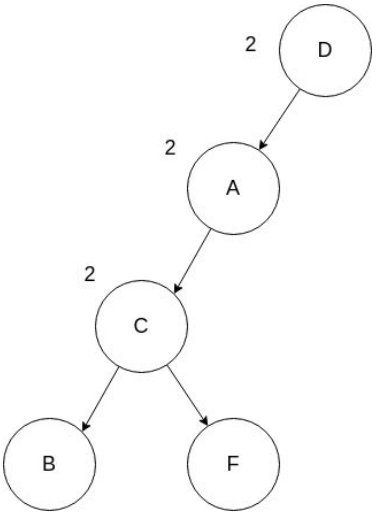
ti1	DACB
ti2	DACF
ti3	DACG

ti4	DABF
ti5	CGB
ti6	DGF
ti7	AGB
ti8	DCGF

Шаг 4

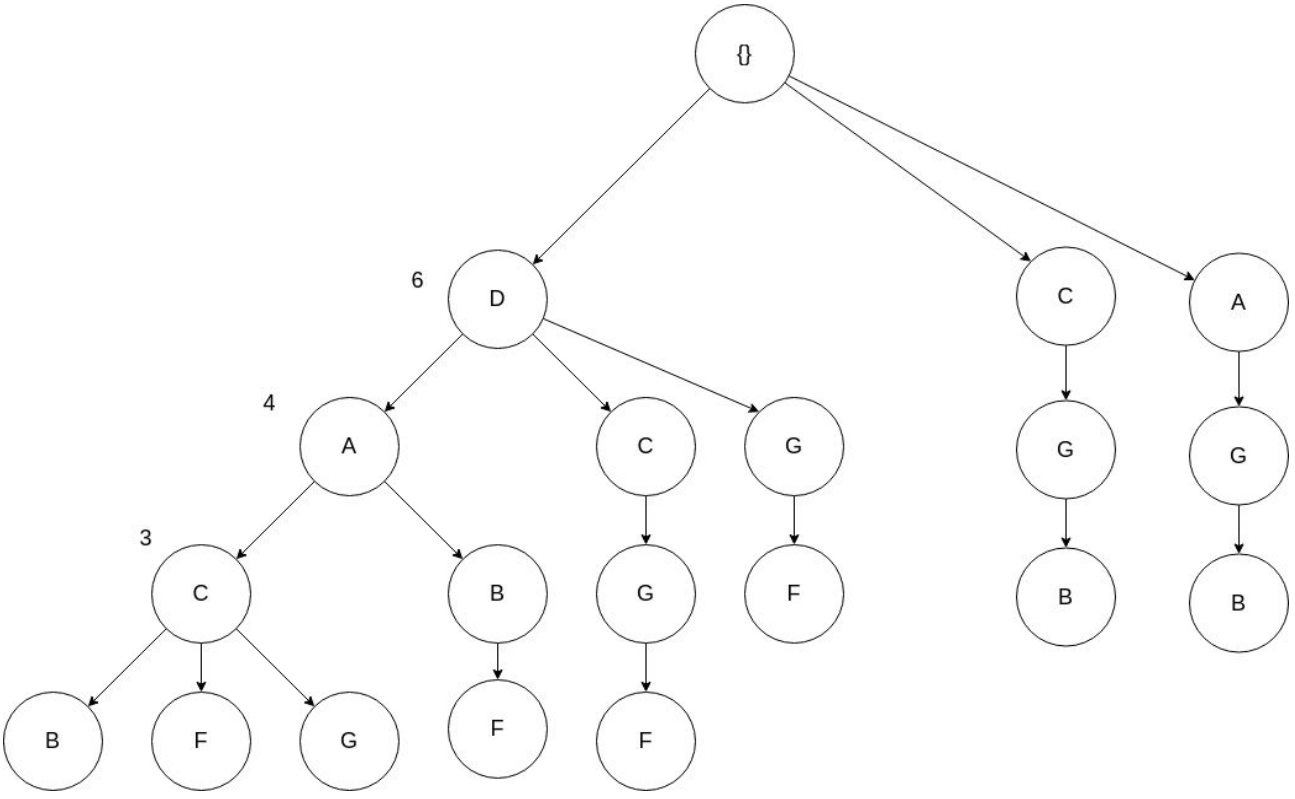


Шаг 4.1



Шаг 4.2

…



Шаг 4.8

Шаг 5

Шаг 5.1 (A)
(AD, 4)

Шаг 5.2 (C)
(CAD, 3), (CD, 1) => (CAD, 3), (AC, 3), (CD, 4)

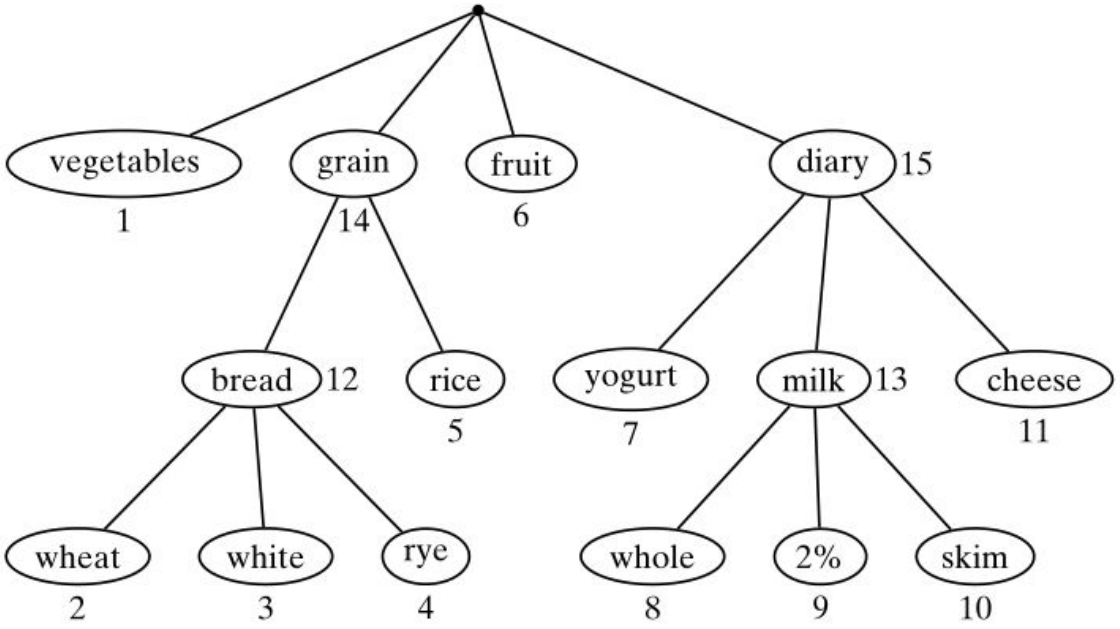
Шаг 5.3 (G)
(GCAD, 1), (GCD, 1) (GD, 1), (GC, 1), (GA, 1) => (GCD, 2), (GD, 3), (GC, 3), (GA, 2)

Шаг 5.4 (B)
(BCAD, 1), (BAD, 1), (BGC, 1), (BGA, 1) => (BAD, 2), (BG, 2), (BA, 3), (BD, 2), (BC, 2)

Шаг 5.5 (F)
(FCAD, 1), (FBAD, 1), (FGCD, 1), (FGD, 1) => (FAD, 2), (FCD, 2), (FGD, 2), (AF, 2), (CF, 2), (DF, 4), (GF, 2)

Задание 2

На рисунке представлена классификация различных продуктов. Каждый лист дерева это конкретный продукт, внутренний узел дерева представляет категорию продукта более верхнего уровня.



A. Каков размер области поиска наборов элементов, если ограничиваться только наборами, состоящими из простых элементов?
x =

$$\sum_{k=1}^{n}\binom{n}{k} = 2^n - 1$$

= 2^11 - 1 = 2047

B. Предположив, что минимальный уровень поддержки = 7/8. Найдите все часто встречающиеся наборы элементов, состоящие только из элементов высокого уровня в таксономии. Имейте в виду, что если в транзакции появляется простой элемент, предполагается, что все его предки высокого уровня также присутствуют в транзакции.

```
itemsets = pd.Series(['2 3 6 7',
                      '1 3 4 8 11',
                      '3 9 11',
                           '1 5 6 7',
                           '1 3 8 10 11',
                      '3 5 7 9 11',
                      '4 6 8 10 11',
                           '1 3 5 8 11'])
for i in itemsets.keys():
    itemsets[i] = itemsets[i].split()
    for j in range(len(itemsets[i])):
        if 2 <= int(itemsets[i][j]) <= 5:
            itemsets[i][j] = '14'
        if 7 <= int(itemsets[i][j]) <= 11:
            itemsets[i][j] = '15'

te = TransactionEncoder()
te_ary = te.fit_transform(itemsets)
df = pd.DataFrame(te_ary, columns=te.columns_)

frequent_itemsets = apriori(df, min_support=7 / 8,
                            use_colnames=True)
print(frequent_itemsets)
```

	support	itemsets
0	1.0	(14)
1	1.0	(15)
2	1.0	(14, 15)