

Вариант 1. Микропроцессоры.

Цель работы: изучение принципов построения и организации экспертных систем, базирующихся на логике и правилах.

Таблица сравнительных характеристик микропроцессоров.

Процессор	Тактовая частота	FLOP *	Кэш данных	Кэш команд	Кэшпамять L2	Частота системной шины	SPECint **	SPECfp **	Ватт ***
Intel Pentium 4	1.3-3.2 ГГц	2	8 КБ	12 КБ	256-512 КБ на чипе	400-800 МГц	1620	1494	82 Вт
Intel Itanium	733, 800 МГц	4	16 КБ	16 КБ	96 КБ	266 МГц	365	701	116-130 Вт
AMD Athlon XP	1.333-2.2 ГГц	3	64 КБ	64 КБ	512 КБ на чипе	400 МГц	1080	982	60-68 Вт
AMD Athlon MP	0.852-1.33 ГГц	3	64 КБ	64 КБ	256 КБ на чипе	266 МГц	781	656	46.1-54.7 Вт
Sun UltraSPARC III	600-1200 МГц	2	64 КБ	32 КБ	до 16 МБ внешней	150 МГц	722	1118	70 Вт @ 750 МГц
IBM PowerPC 750FX	0.9-1 ГГц	1	32 КБ	32 КБ	512 КБ на чипе	200 МГц			5.7 Вт @ 900 МГц
SandCraft SR71000	500-800 МГц	2	32 КБ	32 КБ	512 КБ на чипе	133 МГц			4 Вт @ 600 МГц
Alpha 21264	0.5-1 ГГц	2	64 КБ	64 КБ	до 8 МБ	200 МГц	679	960	90 Вт @ 750 МГц
IBM Power 4	1.1-1.3 ГГц	4	32 КБ	64 КБ	от 0.5 до 16 МБ	400 МГц	814	1169	
HP PA-8700	650, 750 МГц	4	0.75 МБ	1.5 МБ	нет		569	581	12.5 Вт
SPARC64 GP	400-675 МГц	2	128 КБ	128 КБ	8 МБ внешней		478	509	
AMD Opteron	1.4-2 ГГц	2	64 КБ	64 КБ	1 МБ		1335	1339	84.7 Вт
Intel Xeon	1.4-2 ГГц		8 КБ	20 КБ	1 МБ	400-533 МГц	1243	1152	110 Вт
Intel Itanium 2	1.3-1.5 ГГц	4	32 КБ - общий для данных и команд		256 КБ	400 МГц	1322	2119	
Alpha 21364	1.15-1.77 ГГц	4	64 КБ	64 КБ	1.75 МБ		877	1482	155 Вт
Crusoe	667-1000 МГц		64 КБ	64 КБ	512 КБ				7.5 Вт
Intel Pentium M	900-1700 МГц		64 КБ	64 КБ	1 МБ	400 МГц			22 Вт

* FLOP = операций с плавающей точкой за такт.

** Лучшие результаты тестов SPECint2000 и SPECfp2000 для однопроцессорных конфигураций.
Результаты были взяты с сервера www.spec.org

*** Максимальное потребление электроэнергии.

Для того, чтобы решить задачу подбора необходимого микропроцессора воспользуемся методом дерева решений. С помощью языка программирования Python и среды разработки Jupyter Notebook я построил дерево решений для подбора необходимой модели микропроцессора за минимальное число вопросов:

```
In [1]: import pandas as pd
import numpy
```

```
In [2]: MP_data = pd.read_excel('ЛР4.xlsx')
MP_data
```

Out[2]:

	Низкая тактовая частота (ГГц)	Высокая тактовая частота (ГГц)	FLOP	Наличие общего кэш	Кэш- данных	Кэш- команд	Носитель кэш- памяти L2	Количество кэш памяти	Частота системной шины	Результаты тестов SPECint	Результаты тестов SPECfp	Максимальное потребление электроэнергии (Ватт)	№ модели
0	1.300	3.200	2	0	8	12	2	512	600	1620	1494	82.0	1
1	0.733	0.800	4	0	16	16	2	96	266	365	701	123.0	2
2	1.333	2.200	3	0	64	64	2	512	400	1080	982	64.0	3
3	0.850	2.133	3	0	64	64	2	256	266	781	656	50.0	4
4	0.600	1.200	2	0	64	32	1	16384	150	722	1118	70.0	5
5	0.900	1.000	1	0	32	32	2	512	200	914	1097	5.7	6
6	0.500	0.800	2	0	32	32	2	512	133	914	1097	4.0	7
7	0.500	1.000	2	0	64	64	2	8192	200	679	960	90.0	8
8	1.100	1.300	4	0	32	64	2	8192	400	814	1169	63.0	9
9	0.650	0.750	4	0	768	1536	0	0	323	569	581	12.5	10
10	0.400	0.670	2	0	128	128	1	8192	323	478	509	63.0	11
11	1.400	2.000	2	0	64	64	2	1024	323	1335	1339	84.7	12
12	1.400	2.000	3	0	8	20	2	1024	465	1243	1152	110.0	13
13	1.300	1.500	4	1	16	16	2	256	400	1322	2119	63.0	14
14	1.150	1.770	4	0	64	64	2	1792	323	877	1482	155.0	15
15	0.667	1.000	3	0	64	64	2	512	323	914	1097	7.5	16
16	0.900	1.700	3	0	64	64	2	1024	400	914	1097	22.0	17

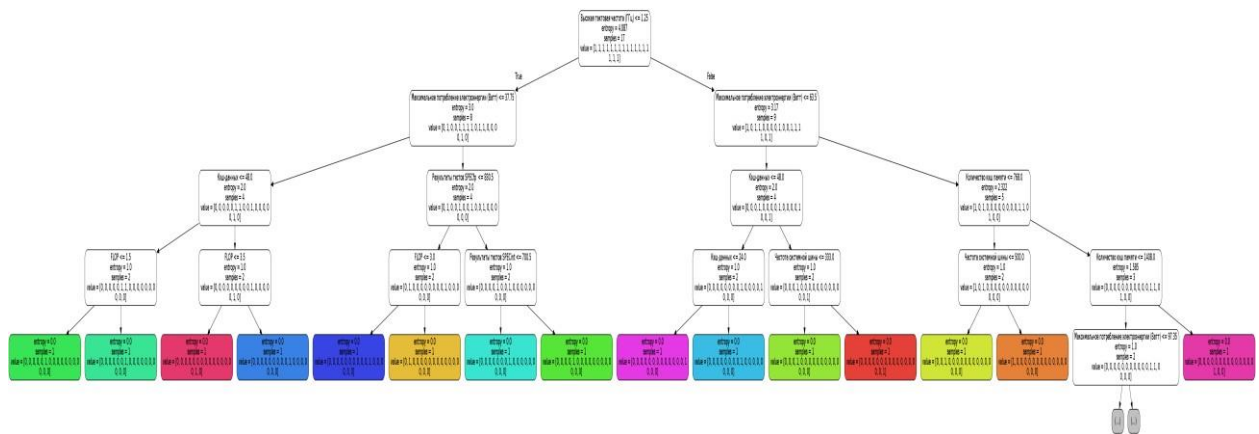
```
In [3]: from sklearn.tree import DecisionTreeClassifier
from sklearn import tree
from sklearn.tree import export_graphviz
```

```
In [4]: X_train = MP_data.iloc[:, :-1]
y_train = MP_data.iloc[:, -1].values
```

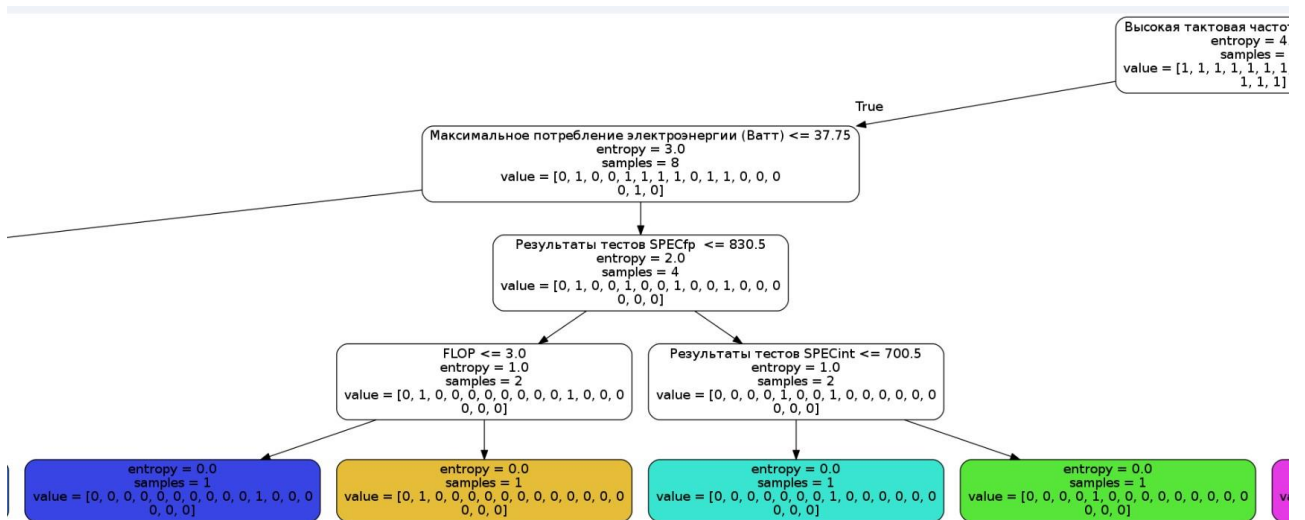
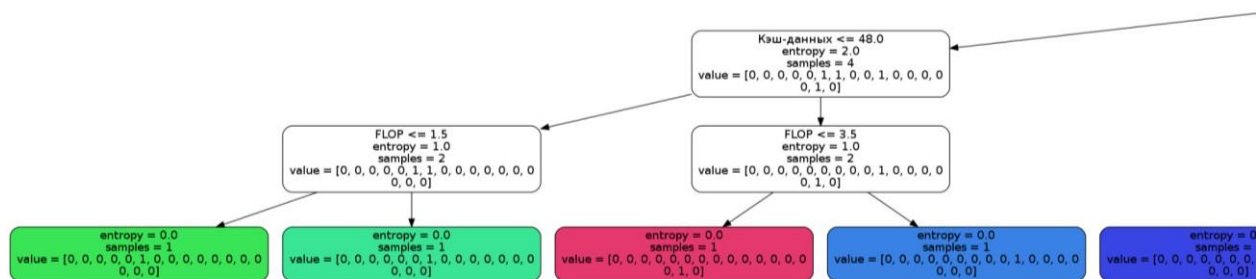
```
In [8]: decision_tree = tree.DecisionTreeClassifier(criterion='entropy', min_samples_split=3)
decision_tree.fit(X_train, y_train)
```

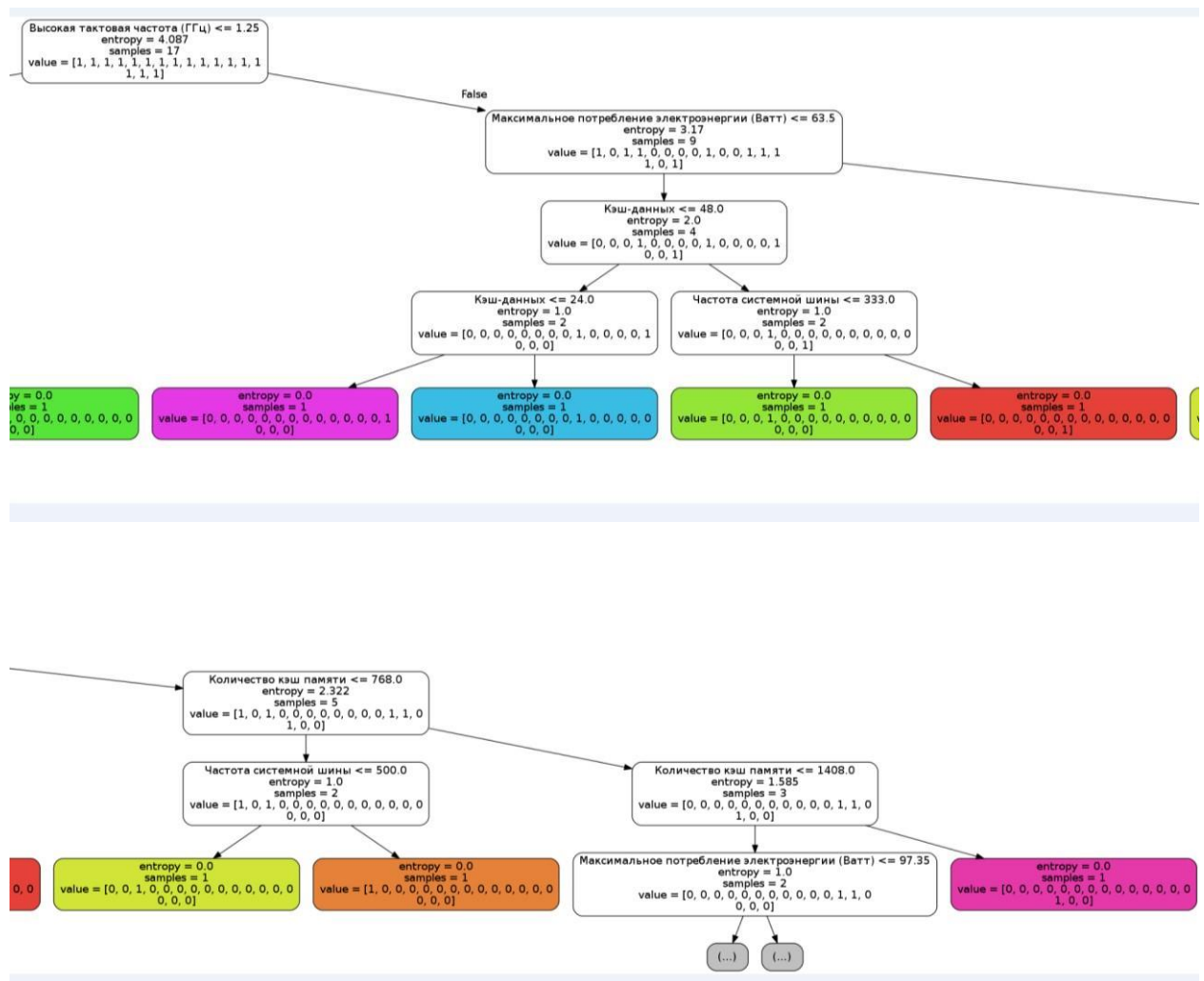
```
Out[8]: DecisionTreeClassifier(ccp_alpha=0.0, class_weight=None, criterion='entropy',
max_depth=None, max_features=None, max_leaf_nodes=None,
min_impurity_decrease=0.0, min_impurity_split=None,
min_samples_leaf=1, min_samples_split=3,
min_weight_fraction_leaf=0.0, presort='deprecated',
random_state=None, splitter='best')
```

```
In [9]: export_graphviz(decision_tree, out_file="tree.dot", max_depth = 4,
impurity = True, feature_names = X_train.columns,
rounded = True, filled= True )
```



Фрагменты:





За 4 вопроса мы можем точно подобрать 15 из 17 микропроцессоров. А за 5 вопросов подберём микропроцессор из всех моделей. По построенному дереву решений осталось написать программу, в основе которой будет лежать классический алгоритм ветвления.

Листинг программы:

```
print('Вводите y или n при ответе на вопросы.')
print('Верхний порог тактовой частоты больше 1.25ГГц?')
a = input()
if a == "y":
    print('Максимальное потребление электроэнергии больше чем 63.5 Ватт? (Прогресс подбора: 47%)')
    a = input()
    if a == "y":
        print('Количество кэш памяти L2 свыше 768 КБ? (Прогресс подбора: 71%)')
        a = input()
        if a == "y":
            print('Количество кэш памяти L2 свыше 1408 КБ? (Прогресс подбора: 83%)')
            a = input()
            if a == "y":
                print('Максимальное потребление электроэнергии больше чем 97.35 Ватт? (Прогресс подбора: 88%)')
                a = input()
                if a == "y":
                    print('Микропроцессор найден!')
```

```

        a = input()
    if a == "y":
        print('Вам подходит Alpha 21364.')
    else:
        print('Максимальное потребление электроэнергии
        больше чем 97.35 Ватт? (Прогресс подбора: 88%)')
        a = input()
        if a == "y":
            print('Вам подходит Intel Xeon.')
        else:
            print('Вам подходит AMD Opteron.')
    print('Частота съёмной шины больше 500 МГц? (Прогресс подбора: 88%)')
    a = input()
    if a == "y":
        print('Вам подходит Intel Pentium 4.')
    else:
        print('Вам подходит AMD Athlon XP.')
    print('Кэш данных больше 48 КБ? (Прогресс подбора: 76%)')
    a = input()
    if a == "y":
        print('Частота системной шины больше 333 МГц? (Прогресс подбора:
        88%)')
    a = input()
    if a == "y":
        print('Вам подходит Intel Pentium M')
    else:
        print('Вам подходит AMD Athlon MP.')
    else:
        print('Кэш данных больше 24 КБ? (Прогресс подбора:
        88%)')
    a = input()
    if a == "y":
        print('Вам подходит IBM Power 4')
    else:

```

```

        print('Вам подходит Intel Itanium 2.') else:
    print('Максимальное потребление электроэнергии больше чем 37.75
Ватт? (Прогресс подбора: 53%)')        a = input()        if a == "y":
    print('Результат тестов SPECfp лучше 830.5? (Прогресс подбора:
76%)')

        a = input()
    if a == "y":
        print('Результат тестов SPECint лучше 700.5? (Прогресс подбора:
88%)')

            a = input()
            if a == "y":
                print('Вам подходит Sun UltraSPARC III.')
            else:
                print('Вам подходит Alpha 21264.') else:        print('FLOP (опер. с
плав. точкой за такт) больше 3?
(Прогресс подбора: 88%)')

                a = input()
                if a == "y":
                    print('Вам подходит Intel Itanium.')
                else:
                    print('Вам подходит SPARC64 GP.') else:
                        print('Кэш данных больше 48 КБ? (Прогресс подбора: 76%)')
                        a = input() if
                        a == "y":
                            print('Кэш память L2 может отсутствовать? (Прогресс подбора: 88%)')
                                a = input()
                                if a == "y":
                                    print('Вам подходит HP PA-8700.')
                                else:
                                    print('Вам подходит Crusoe.') else:        print('FLOP (опер. с плав.
точкой за такт) равно 2?
(Прогресс подбора: 88%)')

                                    a = input()
                                    if a == "y":

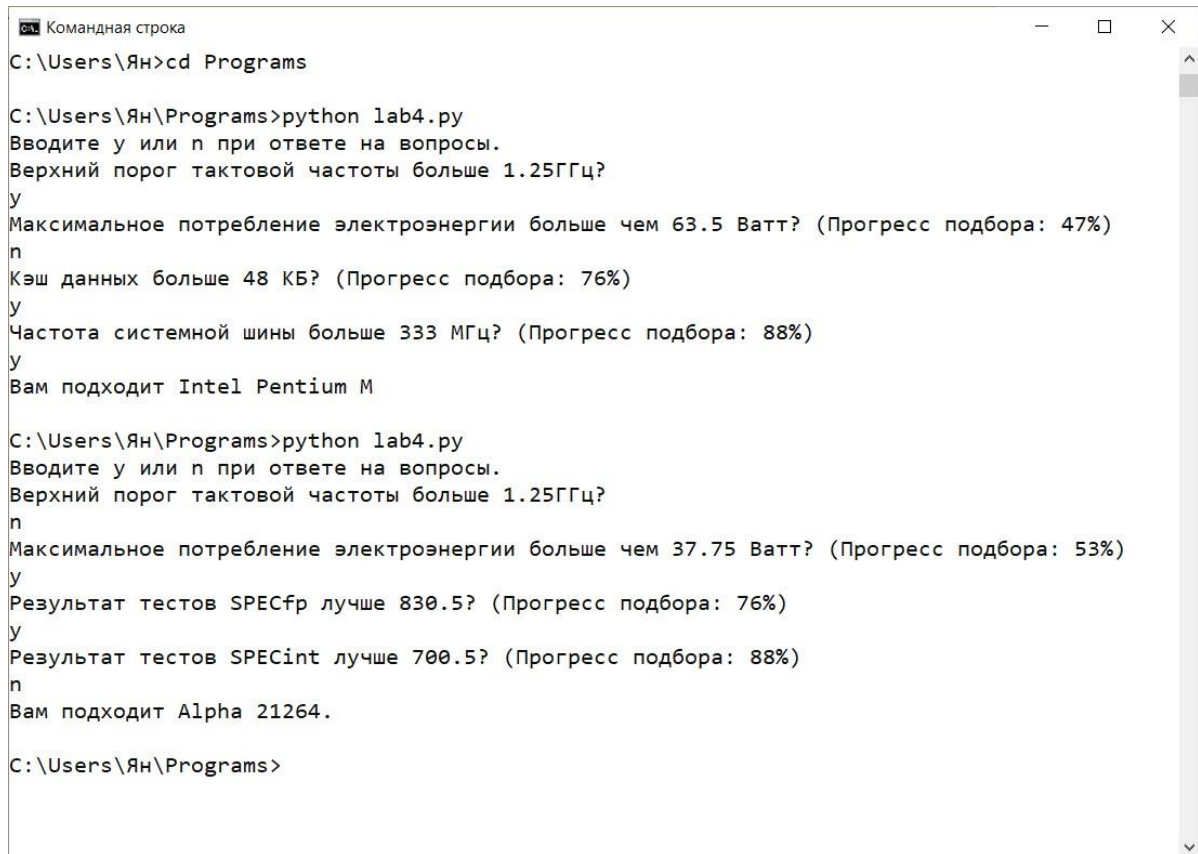
```

```
print('Вам подходит SandCraft SR71000.')
```

else:

```
print('Вам подходит IBM PowerPC 750FX.')
```

Пример работы разработанной экспертной системы:



```
Командная строка
C:\Users\Ян>cd Programs

C:\Users\Ян\Programs>python lab4.py
Вводите у или н при ответе на вопросы.
Верхний порог тактовой частоты больше 1.25ГГц?
у
Максимальное потребление электроэнергии больше чем 63.5 Ватт? (Прогресс подбора: 47%)
н
Кэш данных больше 48 КБ? (Прогресс подбора: 76%)
у
Частота системной шины больше 333 МГц? (Прогресс подбора: 88%)
у
Вам подходит Intel Pentium M

C:\Users\Ян\Programs>python lab4.py
Вводите у или н при ответе на вопросы.
Верхний порог тактовой частоты больше 1.25ГГц?
н
Максимальное потребление электроэнергии больше чем 37.75 Ватт? (Прогресс подбора: 53%)
у
Результат тестов SPECfp лучше 830.5? (Прогресс подбора: 76%)
у
Результат тестов SPECint лучше 700.5? (Прогресс подбора: 88%)
н
Вам подходит Alpha 21264.

C:\Users\Ян\Programs>
```