Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«Пермский национальный исследовательский**

**политехнический университет»**

Факультет прикладной математики и механики

Кафедра «Вычислительная математика, механика и биомеханика»

Направление: 09.03.02 «Информационные системы и технологии»

**ЛАБОРАТОРНАЯ РАБОТА № 3**

**Проектирование базы знаний и реализация ЭС на основе семантических сетей и фреймов**

Выполнил:

студент гр. ИСТ-19-2Б, А.С. Бабикова

Проверил:

ассистент каф. ВММБ, Г.В. Нетбай

Пермь 2022

Описание метода

### ***Семантическая сеть*** – информационная модель предметной области, имеет вид ориентированного графа, вершины которого соответствуют объектам предметной области, а ребра задают отношения между ними. Объектами могут быть понятия, события, свойства, процессы.

### Структура семантической сети отображается графически с помощью узлов и дуг, соединяющих их. Узлы называются объектами, а дуги – связями или ребрами. Связи в семантической сети применяются для представления отношений, а узлы, как правило, для представления физических объектов, концепций или ситуаций.

### 

### 

Описание решаемой задачи

Перед тем как проектировать базу знаний в формате yaml, была спроектирована семантическая сеть в визуальном представлении. Рис.1.

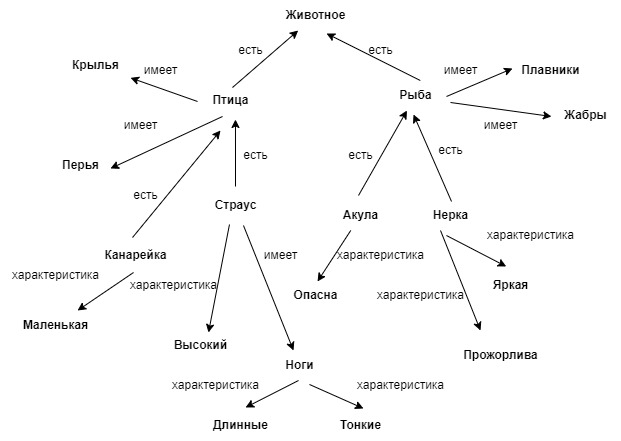


Рисунок 1 – Семантическая сеть.

### Затем данная сеть была прописана в файл формата yaml, который содержит список объектов(Objects) и список что с чем и по какой связи взаимодействует(Connection). Пример некоторых Objects и Connection представлен на рис. 2.

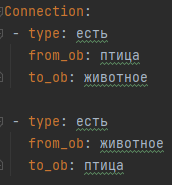
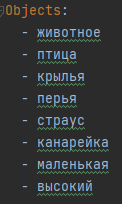


Рисунок 2 – база знаний в YAML файле.

### Принцип работы программы заключается в следующем:

### 1.Пользователю выводится список объектов и предлагается выбрать один из представленных. Рис.3.

### 

### Рисунок 3 – Список объектов.

### 2. Затем пользователю выводятся связи, и он выбирает одну из возможных. Рис.4.

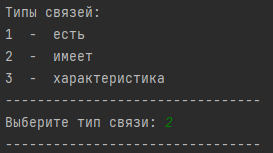


Рисунок 4 – Типы связей.

3. Из полученных данных программа формирует ответ путем перебора БЗ. Рис.5.

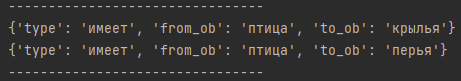


Рисунок 5 – Полученный ответ.

### 4. Затем программа для нового поиска ответа просит нажать enter. И происходят аналогичные действия соответствующие пунктам 1-3. Рис.6.



Рисунок 6 – Продолжение работы программы.

Вывод

### В ходе работы была спроектирована база знаний, которая хранится в формате yaml с представлением знаний в виде семантической сети, реализована оболочка экспертной системы, которая работает с данной базой знаний.

Приложение 1

### import yaml from pprint import pprint from numpy import array from pandas.core.common import flatten from collections import OrderedDict while True: with open('D:/Учеба ПНИПУ/4 курс/Столбов, Нетбай/ЛР\_3/base.yaml', encoding='utf-8') as f: data = yaml.safe\_load(f) #pprint(data)#вывод yaml ob=[];#массив с объектами ob.append(data.get('Objects')); #print(ob); ob = (list(array(ob).flat)) ob = (list(flatten(ob))) sc = 0; print("--------------------------------") print("Объекты: "); for i in range (len(ob)): sc = sc + 1; print(sc, " - ", ob[i]); print("--------------------------------") vybor = int(input("Выберите объект: ")) print("--------------------------------") type = []; #тип связи from\_ob = [];#от какого объекта to\_ob = [];#к какому объекту for item in range(len(data['Connection'])): type.append(data['Connection'][item]['type']); from\_ob.append(data['Connection'][item]['from\_ob']); to\_ob.append(data['Connection'][item]['to\_ob']); type = (list(array(type).flat)) type = (list(flatten(type))) type = list(OrderedDict.fromkeys(type))#удалить дубликаты #print(type); #print(from\_ob); #print(to\_ob); sc\_1 = 0; print("Типы связей: "); for i in range (len(type)): sc\_1 = sc\_1 + 1; print(sc\_1, " - ", type[i]); print("--------------------------------") vybor\_2 = int(input("Выберите тип связи: ")) print("--------------------------------") #print(vybor); #print(vybor\_2); #answer = type[vybor\_2-1] #print("ОТВЕТ", answer); #answer\_2 = ob[vybor-1] #print("ОТВЕТ №2", answer\_2) for j,doc in data.items(): if(j == "Connection"): i=0 while (i< len(doc)): tr = (data['Connection'][i]['type']); tr2 = (data['Connection'][i]['from\_ob']); tr3 = (data['Connection'][i]['to\_ob']); if(tr == type[vybor\_2-1]) and (tr2 == ob[vybor-1]): print(data['Connection'][i]); i = i + 1; print("--------------------------------") input("Для продолжения нажмите Enter")