ПРИЛОЖЕНИЕ

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ В.Ф. УТКИНА»

Кафедра «Вычислительная и прикладная математика»

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ «Проектирование интеллектуальных информационных систем»

Направление подготовки

09.03.03 «Прикладная информатика»

Направленность (профиль) подготовки

«Прикладная информатика в экономике»

Уровень подготовки – бакалавриат

Квалификация выпускника – бакалавр

Форма обучения – очная, заочная

Рязань

1 ОБЩИЕ ПОЛОЖЕНИЯ

Оценочные материалы – это совокупность учебно-методических материалов и процедур, предназначенных для оценки качества освоения обучающимися данной дисциплины как части основной образовательной программы.

Цель – оценить соответствие знаний, умений и уровня приобретенных компетенций, обучающихся целям и требованиям основной образовательной программы в ходе проведения текущего контроля и промежуточной аттестации.

Основная задача – обеспечить оценку уровня сформированности компетенций, приобретаемых обучающимся в соответствии с этими требованиями.

Контроль знаний обучающихся проводится в форме промежуточной аттестации – экзамена.

2 ОПИСАНИЕ ПОКАЗАТЕЛЕЙ И КРИТЕРИЕВ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ

Сформированность каждой компетенции в рамках освоения данной дисциплины оценивается по трехуровневой шкале:

1) пороговый уровень является обязательным для всех обучающихся по завершении освоения дисциплины;

2) продвинутый уровень характеризуется превышением минимальных характеристик сформированности компетенций по завершении освоения дисциплины;

3) эталонный уровень характеризуется максимально возможной выраженностью компетенций и является важным качественным ориентиром для самосовершенствования.

Уровень освоения компетенций, формируемых дисциплиной

а) описание критериев и шкалы оценивания тестирования:

Шкала оценивания	**Критерий**
3 балла (эталонный уровень)	уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 85 до 100%
2 балла (продвинутый уровень)	уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 75 до 84%
1 балл (пороговый уровень)	уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 60 до 74%
0 баллов	уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 0 до 59%

б) описание критериев и шкалы оценивания теоретического вопроса:

Шкала оценивания	**Критерий**
3 балла (эталонный уровень)	выставляется студенту, который дал полный ответ на вопрос, показал глубокие систематизированные знания, смог привести примеры, ответил на дополнительные вопросы преподавателя.
2 балла (продвинутый уровень)	выставляется студенту, который дал полный ответ на вопрос, но на некоторые дополнительные вопросы преподавателя ответил только с помощью наводящих вопросов.
1 балл (пороговый уровень)	выставляется студенту, который дал неполный ответ на вопрос в билете и смог ответить на дополнительные вопросы только с помощью преподавателя.
0 баллов	выставляется студенту, который не смог ответить на вопрос

в) описание критериев и шкалы оценивания практического задания:

Шкала оценивания	**Критерий**

3 балла (эталонный уровень)	Задание решено верно
2 балла (продвинутый уровень)	Задание решено верно, но имеются технические неточности в выполнении
1 балл (пороговый уровень)	Задание решено верно, с дополнительными наводящими вопросами преподавателя
0 баллов	Задание не решено

На экзамен выносится: тестовое задание, 1 практическое задание и 1 теоретический вопрос. Студент может набрать максимум 9 баллов. Итоговый суммарный балл студента, полученный при прохождении промежуточной аттестации, переводится в традиционную форму по системе «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Шкала оценивания	**Критерий**	
отлично (эталонный уровень)	8 – 9 баллов	Обязательным условием является выполнение всех предусмотренных в течение семестра практических заданий и лабораторных работ.
хорошо (продвинутый уровень)	6 – 7 баллов	
удовлетворительно (пороговый уровень)	4 – 5 баллов	
неудовлетворительно	0 – 3 баллов	Студент не выполнил всех предусмотренных в течение семестра текущих заданий

3 ПАСПОРТ ОЦЕНОЧНЫХ МАТЕРИАЛОВ ПО ДИСЦИПЛИНЕ

Контролируемые разделы (темы) дисциплины	**Код контролируемой компетенции (или её части)**	**Наименование оценочного средства**
1 семестр (осенний)		
Тема 1. Инженерия знаний		
Данные и знания, их отличие, особенности извлечения и представления знаний о языковой деятельности человека	ПК-1.3 ПК-3.2 ПК-5.1	Экзамен
Анализ предметной области, выявление ее языковых особенностей, составление словаря основ и морфологических таблиц.	ПК-1.3 ПК-3.2 ПК-5.1	Экзамен
Проектирование морфологического анализа.	ПК-1.3 ПК-3.2 ПК-5.1	Экзамен
Проектирование модели языка, системы составляющих, деревья зависимостей, структурные формулы словосочетаний, синтаксическая фильтрация.	ПК-1.3 ПК-3.2 ПК-5.1	Экзамен
Семантический анализ предложений и текстов русского языка. Языки СЕМП, представление зна-	ПК-1.3 ПК-3.2	Экзамен

ний в форме фреймов, RX-кодов, OWL-моделях, динамической логике, логических базах знаний.	ПК-5.1	
Синтез естественно-языковых предложений, системы машинного перевода.	ПК-1.3 ПК-3.2 ПК-5.1	Экзамен
Использование средств языка Python для анализа естественного языка.	ПК-1.3 ПК-3.2 ПК-5.1	Экзамен, КР
Проектирование архитектуры системы общения, составление базовых примеров запросов.	ПК-1.3 ПК-3.2 ПК-5.1	Экзамен, КР
Проектирование модели языка, системы составляющих, деревья зависимостей, структурные формулы словосочетаний, синтаксическая фильтрация.	ПК-1.3 ПК-3.2 ПК-5.1	Экзамен, КР
Тема 2. Интел туальные решатели задач		
Интел туальные решатели. Деревья вариантов.	ПК-1.3 ПК-3.2 ПК-5.1	Экзамен
Методы решения перебором в ширину и в глубину.	ПК-1.3 ПК-3.2 ПК-5.1	Экзамен
Оценочные функции. Метод ветвей и границ.	ПК-1.3 ПК-3.2 ПК-5.1	Экзамен
Метод равных цен.	ПК-1.3 ПК-3.2 ПК-5.1	Экзамен
Игровые методы . Минимакс.	ПК-1.3 ПК-3.2 ПК-5.1	Экзамен

4 ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ ИЛИ ИНЫЕ МАТЕРИАЛЫ

4.1 Промежуточная аттестация (КР и экзамен)

ПК-1: Способен разрабатывать требования, проектировать и выполнять программную реализацию программного обеспечения
ПК-1.3. Проектирует программное обеспечение и выполняет его программную реализацию

а) типовые тестовые вопросы закрытого типа

1. Раздел искусственного интеллекта о естественно-языковых системах общения в основе содержит:
   - Лингвистический процессор
   - *Модель знаний о языковой деятельности человека*
   - Нейронную сеть для анализа фраз естественного языка
   - Теорию проектирования интеллектуальных информационных систем
2. При разработке систем общения с интеллектуальными информационными системами актуальны следующие свойства языка:
   - Лаконичность
   - Строгая формализация грамматики
   - Ориентация на предметную область
   - *Сложность и многозначность высказываний*
3. Архитектура лингвистического процессора обязательно включает:
   - *Синтаксический анализатор*
   - Модель поведения
   - Блок целей общения
   - Сценарий диалога
4. Морфологический анализатор определяет следующие характеристики словоформ:
   - Синтаксические характеристики
   - *Лексические характеристики*
   - Семантические характеристики
   - Прагматические свойства
5. Алгоритмической основой морфологического анализа является:
   - *Флективный анализ словоформ*
   - Словарь основ
   - Морфологическая таблица
   - Выделение составляющих словоформы
6. Парсер может представлять предложение естественного языка в форме:
   - Семантической сети
   - *Дерева зависимостей*
   - Набора фреймов
   - OWL-формулы
7. Семантический анализатор может представлять предложение естественного языка в форме:
   - Системы составляющих
   - Дерева зависимостей
   - Структурных форму словосочетаний
   - *OWL-формулы*
8. Интеллектуальные решатели задач в своей основе реализуют:
   - Переборные алгоритмы
   - *Моделирование мышления человека при решении интеллектуальных задач*
   - Стратегии эвристического программирования
   - Оценочные функции экспертных знаний
9. Следующие стратегии решения задач не используют предварительных знаний эксперта:
   - Равных цен

Поиск в ширину
Встречный поиск
Ветвей и границ

10. Оценка интеллектуальных методов решения содержит:
Вычислительную сложность алгоритма
Количество затраченных на проектирование человеко-дней
Направленность поиска
Масштабирование стратегии решения

б) типовые тестовые вопросы открытого типа

1. Основные этапы проектирования естественно-языкового общения с информационными системами [Проектирование модели языка, системы составляющих, деревья зависимостей, структурные формулы словосочетаний, синтаксическая фильтрация.]
2. С какой целью проектируется семантический анализ предложений и текстов русского языка. [Построение формальной структуры, отражающей смысл входного предложения]
3. В какой форме может быть представлена семантика естественно-языкового предложения? [Языки СЕМП, представление знаний в форме фреймов, RX-кодов, OWL-моделях, динамической логике, логических базах знаний.]
4. В каких программах может быть опущен модуль семантического анализа? [Синтез естественно-языковых предложений, системы машинного перевода.]
5. Какие современные зыки программирования содержат наиболее развитые средства анализа естественного языка? [Python, ATNL, Refal].
6. С чего начинается проектирование архитектуры интеллектуальной информационной системы? [Выбор и анализ предметной области для проектирования лингвистического процессора.]
7. Какой из этапов разработки лингвистических процессоров является концептуальным? [Проектирование архитектуры системы общения, составление базовых примеров запросов.]
8. С чего начинается проектирование лексического анализатора? [Проектирование и программная реализация словаря основ и морфологических таблиц.]
9. Как называется модуль, реализующий токенизацию? [Морфологический анализ.]
10. Какие работы выполняются по завершении разработки морфологического анализатора? [Программная реализация парсинга для выбранной модели языка.]
11. С чего начинается проектирование семантического анализаторы? [Выбор типа и проектирование модели предметной области.]
12. Какой из этапов проектирования естественно-языковых систем является заключительным и может быть опущен в простых реализациях? [Программная реализация синтеза естественно-языкового ответа.]
13. Как называются интеллектуальные информационные системы, реализующие построение последовательности действий решения сложных задач? [Интеллектуальные решатели задач]
14. Какой механизм теории графов используется в интеллектуальных решателях задач? [Дерево вариантов]
15. Что реализуют методов перебора в глубину и в ширину? [Эвристические методы проектирования без начальных знаний]
16. Какая стартегия реализует перебор в глубину с использованием оценочной функции? [Алгоритм поиска по градиенту]

ПК-3: Способен выполнять работы и управление работами по созданию и сопровождению информационных систем

ПК-3.2. Выполняет проектирование и реализацию информационной системы

а) типовые тестовые вопросы закрытого типа

1. С помощью каких таблиц реализуется морфологический анализ?:
   Словарь основ и морфологическая таблица
   Таблица окончаний и грамматическая информация
   Используются оба предыдущих варианта
2. Какие из структур синтаксического представления позволяет выделить словосочетания, но не учитывает зависимости между ними?
   Системы составляющих
   Структурные формулы словосочетаний
   Деревья зависимостей
3. Какие структуры синтаксической формы могут быть размеченными?:
   Системы составляющих
   Структурные формулы словосочетаний
   Деревья зависимостей
   Обобщенная синтаксическая структура
4. Как задать формулу словосочетания «дерево зависимостей для интеллектуального синтаксического анализа»?:
   С→С→Р→ПС
   Ср→Ср→П
   ПСрСр
   ПСр&Ср
5. Для чего в структурных формулах словосочетаний используется символ & ?
   Системы составляющих
   Структурные формулы словосочетаний
   Деревья зависимостей
   Обобщенная синтаксическая структура
6. Какие структуры синтаксической формы могут быть размеченными?:
   Он обозначает логическое ИЛИ
   Обозначает логическое И
   Обозначает начало следующего словосочетания
   Обозначает сочинительный союз
7. Какой из способов поиска решения наиболее эффективен?
   От начальной вершины к целевой
   От целевой вершины к начальной
   Встречный поиск

б) типовые тестовые вопросы открытого типа

1. Составьте простейшую программу перебора в глубину.

```
1.  DFS(G, u)
2.      u.visited = true
3.      for each v ∈ G.Adj[u]
4.          if v.visited == false
5.              DFS(G,v)
6.
7.  init() {
8.      For each u ∈ G
9.          u.visited = false
10.      For each u ∈ G
11.        DFS(G, u)
12. }
```

2. Приведите пример программы на языке Python, реализующий перебор в ширину.

```
import collections
def bfs(graph, root):
    visited, queue = set(), collections.deque([root])
    visited.add(root)
    while queue:
        vertex = queue.popleft()
        for neighbour in graph[vertex]:
            if neighbour not in visited:
                visited.add(neighbour)
                queue.append(neighbour)
if __name__ == '__main__':
    graph = {0: [1, 2], 1: [2], 2: [3], 3: [1,2]}
    breadth_first_search(graph, 0)
```

3. Запишите схему рекурсивного алгоритма поиска по градиенту

Recursive Program Gradient_Heuristic (Ситуация S)

1. if (Goal (S)), return (O'key); {Если цель, вернуть O'key. }
2. if Deadend (S),return (FAIL);
   {Если тупик, вернуть "Неудача".}
3. <Rules> ← Apprules (S); { Apprules (Appreciate Rules) - }
   { процедура упорядочивающая продукции }
   { с помощью оценочной функции и }
   { выбирающая лучшую применимую к S }
   { продукцию. }
4. LOOP: { Метка начала цикла. }
   if (NULL <Rules>) return (FAIL);
   { NULL - процедура проверки на пустоту. }
5. <R> ← First <Rules>;
6. <Rules> ← Tail <Rules>; {Tail - процедура удаления }
   {отработавшей продукции. }
7. <RS> ← R (<S>); {<RS> - новая область ситуаций. }
8. <Result> ← Gradient_Heuristic (RS);
   {<Result> - результат рекурсивного вызова программы.}
9. if <Result> = FAIL, go LOOP; { Переход на метку. }
10. Return Build(R, Result); {Здесь программа оказывается}
    {только при обнаружении цели. Build - процедура }
    {составления решения из продукций, находящихся }
    {на пути решения.

4. Составьте схему алгоритма поиска решения с оценочной функцией

Recursive Program Intelligent_Heuristic(Ситуации SList)

1. <S> ← First (SList){Выбор первой ситуации списка. }
2. if Member(S, Tail(SList)) return (FAIL);
   {Member - процедура проверки на вхождение }
   { новой ситуации S в ранее рассмотренные. }
3. if (Goal (S)), return (O'key); {Если цель, вернуть O'key. }
4. if Deadend (S),return (FAIL); {Если тупик, вернуть "Неудача".}
5. if Length (SList) > BOUND, return(FAIL);
   {Length - вычисление длины списка.}
6. <Rules> ← Apprules (S); { Apprules (Appreciate Rules) - }
   { процедура упорядочивающая продукции }
   { с помощью оценочной функции и }
   { выбирающая лучшую применимую к S }
   { продукцию. }
7. LOOP: { Метка начала цикла. }
   if (NULL <Rules>) return (FAIL);
   { NULL - процедура проверки на пустоту. }
8. <R> ← First <Rules>;
9. <Rules> ← Tail <Rules>; {Tail - процедура удаления }

{отработавшей продукции. }

10.<RS>← R(<S>); {<RS>-новая область ситуации.}

11. <RSList> ← Build(RS, SList);

{Построение нового списка, в котором }

{первой будет новая полученная ситуация.}

12. <Result> ← Intelligent_Heuristic (RSList);

{<Result> - результат рекурсивного вызова программы.}

13. if <Result> = FAIL, go LOOP; { Переход на метку. }

14.Return Build(R, Result); {Здесь программа оказывается}

{только при обнаружении цели. Build - процедура }

{составления решения из продукций, находящихся }

{на пути решения. }

5.Приведите схему алгоритма равных цен.

1) Поместить начальную вершину s в список, называемый ОТКРЫТ. Положить g(s)=0.

2) Если список ОТКРЫТ пуст, то на выход подается сигнал о неудаче поиска, в противном случае переходить к следующему шагу.

3) Взять из списка ОТКРЫТ ту вершину, для которой величина g имеет наименьшее значение, и поместить ее в список ЗАКРЫТ. Дать этой вершине название n. (В случае совпадения значений выбирать вершину с минимальными g произвольно, но всегда отдавая предпочтение целевой вершине.)

4) Если n есть целевая вершина, то на выход выдать решающий путь, получаемый путем просмотра назад в соответствии с указателями; в противном случае переходить к следующему шагу.

5) Раскрыть вершину n, построив все непосредственно следующие за ней вершины. (Если таковых нет переходить к шагу (2).) Для каждой из такой непосредственно следующей (дочерней) вершины n_i вычислить стоимость g(n), положив $g(n_i)=g(n)+c(n,n_i)$. Поместить эти вершины вместе с соответствующими им только что найденными значениями g в список ОТКРЫТ и построить указатели, идущие назад к n.

6) Перейти к шагу (2).

6.Приведите формулы вычисление базовых характеристик поиска.

1. *Глубина поиска* D: максимальное число вершин, порожденных до целевой или тупиковой вершины в дереве вариантов.

2. *Длина пути решения* L: число вершин, лежащих на минимальном из найденных путей решения.

3. *Общее число порожденных вершин* N.

Производные характеристики поиска

4) *Разветвленность поиска* $R = N / L$.

5) *Направленность поиска* $W = N^{1/L}$.

ПК-5: Способен проводить научно-исследовательские работы и экспериментальные исследования по отдельным разделам темы в области прикладной информатики
ПК-5.1. Осуществляет проведение работ по обработке и анализу научно-технической информации и результатов исследований

а) типовые тестовые вопросы закрытого типа

1. С помощью каких характеристик можно сравнить стратегии поиска?

Базовых характеристик поиска

Производных характеристик поиска

Эффективных характеристик поиска

Всех перечисленных характеристик

2. Что изучает семантика?

Структурные аспекты сочетаний знаков какой-либо системы, правила их образования и преобразования безотносительно к их значениям и функциям

Знаковые системы как средство выражения смысла, занимается установлением зависимости между структурой и выразительными возможностями, изучает интерпретации знаков, знаковых систем, сочетаний, образующих осмысленные тексты

Отношение воспринимающего знаковую систему к самой знаковой системе

3. Что изучает синтактика?

Структурные аспекты сочетаний знаков какой-либо системы, правила их образования и преобразования безотносительно к их значениям и функциям

Знаковые системы как средство выражения смысла, занимается установлением зависимости между структурой и выразительными возможностями, изучает *интерпретации знаков, знаковых систем, сочетаний, образующих осмысленные тексты*

Отношение воспринимающего знаковую систему к самой знаковой системе

4. Что изучает прагматика?

Структурные аспекты сочетаний знаков какой-либо системы, правила их образования и преобразования безотносительно к их значениям и функциям

Знаковые системы как средство выражения смысла, занимается установлением зависимости между структурой и выразительными возможностями, изучает интерпретации знаков, знаковых систем, сочетаний, образующих осмысленные тексты

Отношение воспринимающего знаковую систему к самой знаковой системе

5. Проективное дерево зависимостей это:

Дерево зависимостей, составленной с помощью формальной грамматики

Система составляющих, размеченная метками

Дерево зависимостей, корень которого не лежит ни под одной из дуг отношения зависимостей

б) типовые тестовые вопросы открытого типа

Составьте структурные формулы словосочетаний для следующих фрагментов текста:

Фрагмент	Решение
использование управляющих ЦВМ	С → ПППСр
автоматический поиск информации	ПС → Ср
автоматизированная система поиска информации	ПС → Ср → Ср
международная автоматическая система телефонной связи	ППС → ПСр
автоматизация процессов управления	С → Ср → Ср
проектирование систем обработки информации	С → Ср → Ср → Ср
автоматизированная система управления воздушным движением	ПС → Ср → ПСт
информационная система для административного руководителя	ПС → р → ПСр
символические языки для поиска информации	ПС → р → Ср → Ср
сопротивление в месте повреждения	С → р → Сп → Ср
электрическая сеть с возвратом тока через	ПС→р→Ст→ Ср→ р→Св

землю	
комбинированный трансформатор тока и напряжения	 ПС → Ср

4.2 Типовые вопросы к экзамену по дисциплине

1. Системы общения с информационными системами на естественном языке
2. Токенизация
3. Определение частей речи
4. Лемматизация интеллектуальных информационных систем.
5. Удаление стоп-слов
6. Парсинг зависимостей интеллектуальных информационных систем.
7. Выделение именных групп
8. Распознавание именованных сущностей
9. Понятие эвристического программирования для интеллектуальных информационных систем.
10. Деревья вариантов. Виды вершин.
11. И/ИЛИ деревья.
12. Поиск решения в пространстве задач и в пространстве состояний программирования для интеллектуальных информационных систем.
13. Семиотика как наука о знаковых системах.
14. Синтактика.
15. Семантика.
16. Прагматика.
17. Треугольник Фреге.
18. Синонимия и омонимия слов и предложений.
19. Пресуппозиция в интеллектуальных информационных системах.
20. Вербальный и невербальный уровни общения в интеллектуальных информационных системах.
21. Проектирование морфологического анализа.
22. Основные таблицы для морфологического анализа.
23. Синтаксический анализатор интеллектуальных информационных систем.
24. Понятие цепочки в языке.
25. Цепочки в языке. Основные формулы математической лингвистики.
26. Системы составляющих.
27. Размеченные системы составляющих.
28. Деревья зависимостей.
29. Размеченные деревья зависимостей.
30. Обобщенная синтаксическая структура предложения.
31. Структурные формулы словосочетаний интеллектуальных информационных систем.
32. Метод фильтров в синтаксическом анализе.
33. История развития систем синтаксического анализа.

34. Основные архитектуры лингвистических процессоров интеллектуальных информационных систем.
35. Архитектура интеллектуальных информационных систем с машинным переводом.
36. Архитектура интеллектуальных информационных систем общения с базами данных.
37. Архитектура интеллектуальных информационных систем автоматического индексирования.
38. Архитектура интеллектуальных информационных систем с интеллектуальными решателями задач.
39. Обобщенная архитектура систем общения на естественном языке.
40. Семантический анализ.
41. Языки семантического представления. Требования к ним.
42. Интенсионал и экстенсионал.
43. Сценарии диалога в интеллектуальных информационных системах.
44. Оценка эффективности в интеллектуальных информационных систем. Релевантность.
45. Оценка эффективности в интеллектуальных информационных систем. Пертинентность.
46. Понятие модели поведения в интеллектуальных системах.
47. Язык программирования ATNL. Структура программы.
48. Язык программирования ATNL. Расширенные сети переходов.
49. Язык программирования ATNL. Формы.
50. Язык программирования ATNL. Условия.
51. Язык программирования ATNL. Действия.
52. Язык программирования ATNL. Дуга POP.
53. Язык программирования ATNL. Дуга PUSH.
54. Язык программирования ATNL. Дуга CAT.
55. Язык программирования ATNL. Дуга TST.
56. Генерация ответов в интеллектуальных информационных системах.

4.3 Типовые задачи на экзамен по дисциплине

1. Составить программу токенизации для предложения «Китай с Саудовской Аравиейпровел транзакцию в юанях.»

2. Составить программу токенизации для предложения `'Закат доллара США произойдет в следующем десятилетии'.`

3. Пояснить, что делает следующая программа

```
!pip install natasha spacy==3.3
import spacy
!python -m spacy download ru_core_news_sm
nlp = spacy.load('ru_core_news_sm')
text1 = 'Закат доллара США произойдет в следующем десятилетии'
text2 = 'Китай с Саудовской Аравией
                        провел транзакцию в юанях.'
Doc1 = nlp(text1)
```

```
Doc2 = nlp(text2)
for token in Doc1:
  print (token.text1, token.pos_, token.dep_)
```

4. Пояснить, что делает следующая программа

```
   from natasha import Segmenter, NewsEmbedding,
                                                NewsSyntax-
Parser, Doc
   emb = NewsEmbedding()
   segmenter = Segmenter()
   textS = 'Закат доллара США произойдет в следующем дес
ятилетии –
Н.Рубини, Financial Times.'
   docS = Doc(textS)
   docS.segment(segmenter)
   for token in docS.tokens:
     print(token)
   docS.parse_syntax(syntax_parser)
   docS.sents[0].syntax.print()
```

5. Привести программу перебора в глубину
6. Привести программу перебора в ширину
7. Привести программу градиентного спуска
8. Привести программу поиска от наилучшего частичного пути
9. Привести программу морфологического анализа с использованием морфологических таблиц
10. Привести программу для алгоритма равных цен (задача о комивояжере)
11. Привести программу перебора для стратегии ветвей и границ
12. Привести программу вычисления базовых характеристик поиска
13. Привести программу вычисления производных характеристик поиска
14. Привести программу вычисления эффективных характеристик поиска
15. Записать фрагмент программы полного перебора вариантов в продукционной нотации
16. Привести программу градиентного спуска в продукционной нотации
17. Привести программу поиска от наилучшего частичного пути в продукционной нотации

Оператор ЭДО ООО "Компания "Тензор"
ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ
СОГЛАСОВАНО **ФГБОУ ВО "РГРТУ", РГРТУ,** Овечкин Геннадий Владимирович, Заведующий кафедрой ВПМ **03.11.26** 22:31 (MSK) Простая подпись