



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»  
(ДГТУ)**

**09.04.02 ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ**  
(ТЕХНОЛОГИИ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА В ИНФОРМАЦИОННЫХ  
СИСТЕМАХ)

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ПОДГОТОВКИ  
К ЗАКЛЮЧИТЕЛЬНОМУ ЭТАПУ ОЛИМПИАДЫ

Составители: профессор кафедры «КИС», д. физ.-мат. н. Ляпин А. А.  
доцент кафедры «КИС», к.т.н. Шиляева О.В.  
доцент кафедры «КИС», к. физ.-мат. н. Анисимова Г. Б.

Председатель методической комиссии:  
профессор кафедры «КИС», д. физ.-мат. н. Ляпин А. А.

## **ЗАКЛЮЧИТЕЛЬНЫЙ ЭТАП**

Характер и уровень сложности олимпиадных задач направлены на достижение целей проведения Олимпиады: выявление и поддержка лиц, проявивших выдающиеся способности; стимулирование учебно-познавательной и научно-исследовательской деятельности обучающихся; развитие у обучающихся интеллектуальных и творческих способностей; создание необходимых условий для формирования качественного контингента магистрантов, ориентированных на продолжение академической карьеры; формирование системы непрерывного взаимодействия с одаренной и талантливой молодежью; распространение и популяризация научных знаний; привлечение талантливой молодежи, в том числе из зарубежных стран, к обучению в магистратуре.

Задания дифференцированы по сложности и требуют различных временных затрат на верное и полное решение. Задания направлены на выявление интеллектуального потенциала, аналитических способностей и креативности мышления участников и т.п.

Очный этап Олимпиады проводится только в письменной форме. Каждый участник Олимпиады получает бланк с заданием (если предусматриваются варианты заданий, то необходимо указать и это), содержащий 10 заданий.

При выполнении заданий требуется:

**1. Демонстрировать логическую полноту и обоснованность.** Решение каждой задачи должно быть **полным, последовательным и логически безупречным**. Все утверждения, выводы и вычисления требуют четкого обоснования. Ответы, приведенные без необходимых пояснений, доказательств или промежуточных шагов, не считаются полными и оцениваются ниже.

**2. Проявлять креативность и нестандартный подход.** Приветствуются **оригинальные и оптимальные методы решения**, выходящие за рамки шаблонных подходов. Участникам рекомендуется не только найти верный ответ, но и предложить наиболее изящное, эффективное или неочевидное решение, демонстрирующее глубину понимания проблемы.

**3. Оформлять решения четко и структурированно.** Все записи должны быть **разборчивыми** и вестись на специальных бланках, предоставленных организаторами. Решение необходимо излагать в виде связного текста с четкой структурой (постановка подхода, ход решения, итоговый ответ). Использование общепринятых научных обозначений и терминов обязательно.

**4. Следовать установленным техническим правилам.** Ответы на задачи, требующие конкретной формы (число, последовательность, формула, график), должны быть представлены **в явном и окончательном виде**, оговоренном в условии.

При подготовке к Олимпиаде следует повторить приведенные ниже темы.

## **ПЕРЕЧЕНЬ ЭЛЕМЕНТОВ СОДЕРЖАНИЯ, ВКЛЮЧЕННЫХ В ЗАДАНИЯ ОЛИМПИАДЫ ЗАКЛЮЧИТЕЛЬНОГО ЭТАПА 2025/2026 УЧЕБНОГО ГОДА**

### **Раздел 1. «Элементы программирования»**

#### Описание раздела

Этот блок представляет собой вводный тест на знание базового синтаксиса и основных конструкций языка программирования Python. Вопросы охватывают фундаментальные темы, необходимые для начала программирования.

#### Ключевые темы, проверяемые в блоке

- Базовый вывод данных.
- Работа с переменными и вводом данных.
- Арифметические операции и их приоритет.
- Логические операции и выражения.
- Условные операторы.
- Циклы, включая управление потоком.
- Работа со строками (индексация, срезы, методы).
- Работа со списками и другими структурами данных.
- Основы функций.

#### Характер заданий

Задания сформулированы как вопросы, где требуется предсказать результат выполнения кода, написать корректный синтаксис или определить поведение языковой конструкции.

#### Пример вопросов (заданий)

1. Дан фрагмент программного кода на языке программирования Python. Что будет выведено?

```
my_list = [10, 20, 30, 40, 50]
print(my_list[1:-1:2])
```

Ответ: [20, 40]

В этом задании используется срез (slice). В Python – это операция, которая позволяет получить часть (подмножество) последовательности (списка, строки, кортежа) по определённым правилам.

Часть среза	Значение	Объяснение
1	<b>Начальный индекс</b>	Начинаем с элемента с индексом 1 (второй элемент, 20)
-1	<b>Конечный индекс</b>	Заканчиваем на элементе с индексом -1 (последний элемент, 50), НЕ включая его
2	<b>Шаг</b>	Берём каждый второй элемент

Действие	Текущий индекс	Элемент	Входит в результат?	Почему
Начинаем с индекса 1	1	20	Да (первый элемент среза)	Индекс $1 \geq 1$ и $< 4$ , шаг начинается
Прибавляем шаг 2	3	40	Да	Индекс $3 \geq 1$ и $< 4$
Прибавляем шаг 2	5	-	Нет	Индекс 5 выходит за границу 4

2. Дан фрагмент программного кода на языке программирования Python. Дополните функцию, которая считает сумму только положительных чисел в списке.

```
def sum_positive(numbers):
    """
        Возвращает сумму всех положительных чисел в списке.
    """

    total = 0
    for num in numbers:
        if _____: # Пропущено условие
            total += _____ # Пропущено
    return total

# Тесты
print(sum_positive([1, -2, 3, -4, 5])) # Должно вывести: 9 (1+3+5)
print(sum_positive([-1, -2, -3])) # Должно вывести: 0
print(sum_positive([10, 20, 30])) # Должно вывести: 60
```

Ответ: 1)  $num > 0$   
2)  $num$

<pre>def sum_positive(numbers):     """         Возвращает сумму всех положительных чисел в списке.     """      total = 0     for num in numbers:         if num &gt; 0: # Пропущено условие             total += num # Пропущено     return total  # Тесты print(sum_positive([1, -2, 3, -4, 5])) # Должно вывести: 9 (1+3+5) print(sum_positive([-1, -2, -3])) # Должно вывести: 0 print(sum_positive([10, 20, 30])) # Должно вывести: 60 print(sum_positive([10, 120, 30])) # Должно вывести: 60</pre> <p>9 0 60 160</p>	<p>В этом задании – ключевое слово: положительных – значит, нам нужно отбирать только числа больше нуля. Мы перебираем все числа в списке <code>numbers</code>. Для каждого числа <code>num</code> нужно:</p> <ol style="list-style-type: none"> <li>1) Проверить – является ли оно положительным?</li> <li>2) Если да – добавить его к общей сумме <code>total</code>.</li> </ol> <p>Как записать:</p> <p><code>total += num</code> – добавляем значение переменной <code>num</code>  <code>total = total + num</code> – более длинная запись того же.</p>
--	---

3. В данном задании представлена программа (на языке программирования Python) с ошибкой. Необходимо исправить одну строку в коде и вписать ее полностью в ответ.

```

# Программа должна определять, чётное ли число
def is_even(n):
    if n % 2 == 0:
        return True
    else:
        return False

print("4 - чётное?", is_even(4))

```

Ответ: `if n % 2 == 0:`

В условии `if` нужно сравнивать, а не присваивать.

= – это присваивание (положить значение в переменную);

== – это сравнение (проверить, равны ли значения).

<code>if n % 2 == 0:</code>	<code># Программа должна определять, чётное ли число</code> <code>def is_even(n):</code> <code>    if n % 2 == 0:</code> <code>        return True</code> <code>    else:</code> <code>        return False</code>  <code>print("4 - чётное?", is_even(4))</code>
-----------------------------	--

## Раздел 2. «Элементы искусственного интеллекта»

### Описание раздела

Этот блок представляет собой комплексный тест по основам искусственного интеллекта и машинного обучения. Он охватывает как теоретические концепции, так и их практическую реализацию с использованием библиотек Python.

### Ключевые темы, проверяемые в блоке

- Основы искусственного интеллекта.
- Машинное обучение.
- Предобработка данных.
- Специализированные области машинного обучения.

### Характер заданий

Задания требуют как теоретического понимания концепций ИИ, так и практических знаний о том, какие алгоритмы и библиотеки применяются для решения конкретных задач.

### Пример вопросов (заданий)

1. Вычислите значение матрицы выходов MaxPooling слоя свёрточной нейронной сети (CNN) при использовании окна размерности 2x2 для матрицы входного кодированного изображения:

25	34	2	43
78	134	7	50
132	114	11	21
157	16	41	17

Ответ:

134	50
157	41

**Разбор задания:**

Если есть входная матрица размером  $4 \times 4$  после операции свёртки, операция MaxPooling с окном размером  $2 \times 2$  и шагом 2 скользит окном по входной матрице и вычисляет максимальное значение пикселей в каждом окне. Результатом будет новая матрица с уменьшенными размерами, в которой каждый элемент представляет собой максимальное значение соответствующего окна.

25	34	2	43
78	134	7	50
132	114	11	21
157	16	41	17



134	50
157	41

2. Опишите структуру алгоритма кластеризации K-Means. В каких еще задачах можно использовать данный алгоритм?

Ответ:

**Суть алгоритма** – минимизировать вариацию внутри каждого кластера, с одновременной максимизацией расстояние между кластерами. То есть, объекты внутри одного кластера должны быть максимально похожи друг на друга, а объекты из разных кластеров – максимально различаться.

Пошаговый алгоритм K-means:

1. Инициализация – выбираются  $K$  начальных центроидов (центров кластеров). Это можно сделать случайно (выбрав  $K$  точек из набора данных) или с использованием более продвинутых методов, например K-means++.
2. Назначение кластеров – каждая точка данных назначается ближайшему центроиду на основе выбранной метрики расстояния (обычно евклидова).
3. Обновление центроидов – после того, как все точки назначены кластерам, положение каждого центроида пересчитывается как среднее арифметическое всех точек, принадлежащих соответствующему кластеру.
4. Повторение – шаги 2-3 повторяются до тех пор, пока не будет достигнута сходимость – когда центроиды перестают существенно изменяться или достигнуто максимальное число итераций.
5. Финализация – алгоритм возвращает окончательные кластеры и их центроиды.

### **Применение:**

- Сегментация
- Сжатие изображений
- Обнаружение аномалий
- Кластеризация текстов
- Организация больших наборов данных

## **Раздел 3. «Элементы проектирования информационных систем»**

### Описание раздела

Этот блок проверяет знание основ системного анализа, проектирования и жизненного цикла программного обеспечения и информационных систем. Основное внимание уделяется методологиям, процессам и стандартам.

### Ключевые темы, проверяемые в блоке

1. Основы Software Engineering (Инженерии ПО).
2. Жизненный цикл программного обеспечения (Software/IS Life Cycle).
3. Стандарты.
4. Процессы разработки требований.
5. Тестирование.
6. Каноническое проектирование ИС.
7. Эксплуатация и сопровождение ИС.
8. Обеспечение надежности данных в БД.
9. Модели данных.

### Характер заданий

Вопросы направлены на проверку понимания процессов, методологий и стандартов в области разработки программного обеспечения. Они требуют четкого разграничения понятий, знаний терминологии и понимания последовательности и взаимосвязи этапов в жизненном цикле информационных систем.

### Пример вопросов (заданий)

1. Создать ERD, описывающую модель базы данных (БД) логического уровня для хранения данных по теме «Отдел кадров предприятия»

### Ответ:

ERD- диаграмма, описывающая модель базы данных, может быть схематично представлена ручным способом, например, в виде

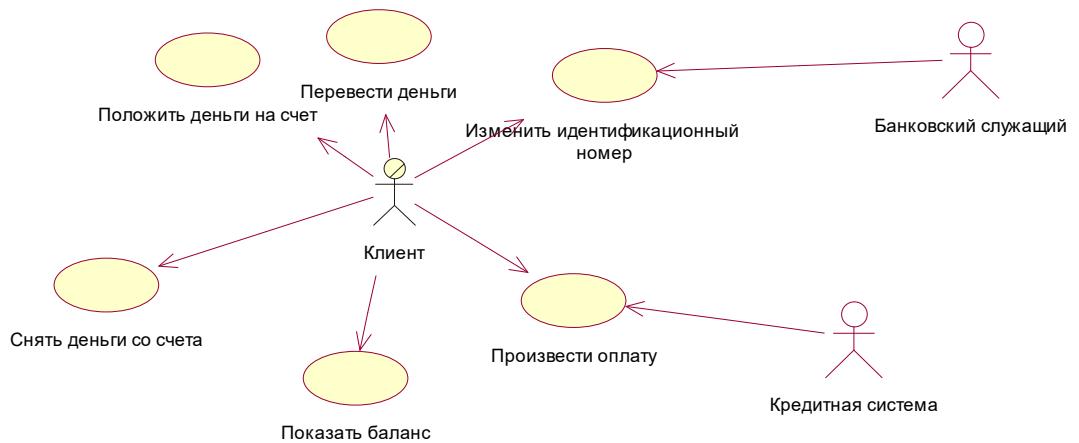


Далее следует пояснить типы связей между сущностями, например между «Отделом» и «Сотрудником». А также пояснить выбор ключей. Например, почему «Код должности» является ключом.

## 2. 2. Создать Use Case диаграмму для описания функциональных требований по теме «Клиент банка в банкомате».

Ответ:

Use Case - диаграмма, описывающая функциональные требования по теме «Клиент банка в банкомате», может быть схематично представлена ручным способом, например, в виде:



Конкретные задачи могут быть описаны в табличной форме:

A	Снять деньги со счета
B	Положить деньги на счет
C	Перевести деньги
D	Показать баланс
E	Произвести оплату
F	Изменить идентификационный номер

Необходимо заполнить таблицу, описывающую задачи и подзадачи для использования ее в системе управления проектом. Например:

Задачи	Предыдущие операции	Время, мин	Количество людей, необходимых для проведения операций
A	-	3	1
B	-	4	1
C	-	6	1
D	A	1	1
E	D	6	1
F	-	10	2

### Литература для подготовки

- 1 Северанс Ч. Р. Python для всех. – СПб.: Питер, 2023. – 272 с.
- 2 Бэрри П. Изучаем программирование на Python. – СПб.: Питер, 2023. – 624 с.
- 3 Рассел С., Норвиг П. Искусственный интеллект: современный подход. – М.: Вильямс, 2021. – 1408 с.
- 4 Жерон О. Hands-on Machine Learning with Scikit-Learn, Keras and TensorFlow. – М.: Питер, 2022. – 1008 с.
- 5 Купер А., Рейман Р., Кронин Д. Анализ требований и проектирование систем. Разработка, основанная на пользовательском опыте. – М.: Вильямс, 2020. – 432 с.
- 6 Соммервилл И. Инженерия программного обеспечения. – М.: Вильямс, 2019. – 624 с.

**Заключительный этап олимпиады «Я – МАГИСТР» для поступающих в  
МАГИСТРАТУРУ**

**09.04.02 ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ**  
(ТЕХНОЛОГИИ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА В ИНФОРМАЦИОННЫХ  
СИСТЕМАХ)

**МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ  
ПО ФОРМИРОВАНИЮ КРИТЕРИЕВ ПРОВЕРКИ (ОЦЕНИВАНИЯ)  
ОЛИМПИАДНЫХ ЗАДАНИЙ ЗАКЛЮЧИТЕЛЬНОГО ЭТАПА**

Составители: профессор кафедры «КИС», д. физ.-мат. н. Ляпин А. А.  
доцент кафедры «КИС», к.т.н. Шиляева О.В.  
доцент кафедры «КИС», к. физ.-мат. н. Анисимова Г. Б.

Председатель методической комиссии:  
профессор кафедры «КИС», д. физ.-мат. н. Ляпин А. А.

## **Критерии проверки**

Вариант заключительного этапа Олимпиады по 09.04.02 Информационные системы и технологии (Технологии искусственного интеллекта в информационных системах) включает в себя 10 заданий разного типа. Каждое задание оценивается от 0 до 10 баллов. Наибольшая итоговая сумма баллов, которой могут быть оценены ответы на все вопросы олимпиадного варианта при условии отсутствия в них ошибок, неправильных, неполных или неточных ответов, равна 100. Неверные ответы оцениваются в 0 баллов. Возможен частичный зачёт баллов за неполный ответ на задание. Под неполным понимается ответ, содержащий правильные ответы не на все вопросы задания. В таком случае присуждается только часть баллов за правильные ответы задания, соответствующая доле от максимально возможного балла. Подсчёт итоговой оценки за задание осуществляется путём суммирования баллов, выставленных за каждый из вопросов.

## ВАРИАНТ 1

### Задание 1.

Всего баллов: 10.

Дан фрагмент программного кода на языке программирования Python:

```
data = [x**2 for x in range(5) if x % 2 == 0]
```

Напишите, что будет содержаться в **data** = ...?

Ответ: data = [0, 4, 16]

### Критерии оценивания

Критерий	Балл
Дан неверный ответ/ответ отсутствует	0
Дан неполный ответ, например, вместо списка в ответе использован кортеж	5
Дан верный ответ	10

### Задание 2.

Всего баллов: 10.

Дан фрагмент программного кода на языке программирования Python:

```
def modify(lst, value):
    lst.append(value)
    value *= 2

numbers = [1]; x = 5
modify(numbers, x)
print(numbers, x)
```

Что будет выведено?

Ответ: [1, 5] 5

### Критерии оценивания

Критерий	Балл
Дан неверный ответ/ответ отсутствует	0
Дан неполный ответ, например, вместо списка в ответе использован кортеж	5
Дан верный ответ	10

### **Задание 3.**

Всего баллов: 10.

Дан фрагмент программного кода на языке программирования Python.  
Дополните функцию, которая находит минимальное число в списке.

```
def find_minimum(numbers):
    """
    Находит минимальное число в списке.
    Если список пустой, возвращает None.
    """
    if len(numbers) == 0:
        return None

    min_value = _____ # Пропущено начальное значение
    for num in numbers:
        if _____: # Пропущено условие сравнения
            min_value = num
    return min_value

# Тесты
print(find_minimum([5, 3, 8, 1, 9])) # Должно вывести: 1
print(find_minimum([10, 20, 30])) # Должно вывести: 10
print(find_minimum([])) # Должно вывести: None
```

Ответ: 1) numbers[0]

2) num < min\_value

Критерии оценивания

Критерий	Балл
Дан неверный ответ/ответ отсутствует	0
Дан верный ответ	10

### **Задание 4.**

Всего баллов: 10.

В данном задании представлена программа (на языке программирования Python) с ошибкой. Необходимо исправить одну строку в коде и вписать ее полностью в ответ.

```

# Программа должна разворачивать строку
def reverse_string(s):
    result = ""

    for i in range(1, len(s) + 1):
        result = s[i] + result

    return result

print("Разворот 'hello' выводит:", reverse_string("hello"))

```

Ответ:

<pre> for i in range(len(s)): </pre>	<pre> # Программа должна разворачивать строку def reverse_string(s):     result = ""      for i in range(len(s)): # Исправленный диапазон         result = s[i] + result      return result  print("Разворот 'hello' выводит:", reverse_string("hello")) </pre>
<b>ИЛИ</b>	
<pre> for i in range(len(s) - 1, -1, -1): </pre>	<pre> # Программа должна разворачивать строку def reverse_string(s):     result = ""      for i in range(len(s) - 1, -1, -1): # Исправленный диапазон         result = result + s[i]      return result  print("Разворот 'hello' выводит:", reverse_string("hello")) </pre>

#### Критерии оценивания

Критерий	Балл
Дан неверный ответ/ответ отсутствует	0
Дан верный ответ	10

#### **Задание 5.**

Всего баллов: 10.

Опишите основные отличия между машинным обучением с учителем и без учителя. Перечислите, какие алгоритмы реализации этих типов машинного обучения Вы знаете.

Ответ:

Основные отличия между машинным обучением с учителем и без учителя заключаются в том, что в обучении с учителем алгоритм обучается на маркированных данных, где каждому примеру соответствует «правильный ответ» (целевая переменная). В обучении без учителя алгоритмы работают с немаркированными

данными, где отсутствуют «правильные ответы». Задача алгоритма – найти скрытую структуру или закономерности в данных самостоятельно.

Выбор между этими подходами определяется не только наличием размеченных данных, но и типом задачи. Иногда оптимальным решением становится комбинирование этих подходов: например, сначала применить обучение без учителя для выявления структуры и предобработки данных, а затем использовать обучение с учителем для решения конкретной задачи прогнозирования.

### **С учителем**

Задачи:

- Классификация – определение категории, к которой относится образец. Например, распознавание спама, диагностика заболеваний, распознавание изображений.
- Регрессия – предсказание непрерывного числового значения. Например, прогнозирование цен, предсказание погоды, оценка кредитного риска.

### **Некоторые алгоритмы:**

- Линейная и логистическая регрессия – для предсказания числовых значений и вероятностей класса.
- Деревья решений и ансамблевые методы – для решения сложных задач классификации и регрессии.
- Метод опорных векторов (SVM) – эффективен в пространствах с большим количеством признаков.
- Байесовские методы – когда важно учитывать неопределенность в данных.

### **Без учителя**

Задача – выявить структуру, которая не была очевидна изначально.

Некоторые алгоритмы:

- Кластеризация – группировка схожих объектов (K-means, DBSCAN, иерархическая кластеризация).
- Снижение размерности – для визуализации и сжатия данных (PCA, t-SNE).
- Обнаружение аномалий – выявление нетипичных паттернов (изолирующий лес, One-Class SVM).
- Ассоциативные правила – поиск зависимостей между элементами (алгоритм Apriori).

### **Критерии оценивания**

Критерий	Балл
Дан неверный ответ/ответ отсутствует	0
Дан неполный ответ	10
Дан верный ответ	20

### **Задание 6.**

Всего баллов: 10.

По заданной матрице ошибок для задачи классификации вычислите следующие метрики оценки общей эффективности модели:

- Accuracy (Общая точность)
- Precision (Точность)

	Predicted: No	Predicted: Yes
Actual: No	TN= 25	FP=24
Actual: Yes	FN=10	TP=76

Ответ: Accuracy = 0.75  
Precision = 0.76

#### Критерии оценивания

Критерий	Балл
Дан неверный ответ/ответ отсутствует	0
Дан верный ответ	10

#### Задание 7.

Всего баллов: 10.

Вычислите значение матрицы выходов AveragePooling слоя свёрточной нейронной сети (CNN) при использовании окна размерности 2x2 для матрицы входного кодированного изображения:

35	47	28	44
43	35	47	53
13	14	1	2
15	16	4	7

Ответ:

40	43
14.5	3.5

#### Критерии оценивания

Критерий	Балл
Дан неверный ответ/ответ отсутствует	0
Дан верный ответ	10

#### Задание 8.

Всего баллов: 10.

1. Создать ERD, описывающую модель БД логического уровня для хранения данных по теме «Библиотека»

Пояснить типы связей между сущностями и выбор ключей.

Ответ:

ERD- диаграмма, описывающая модель базы данных.

Пояснение к типам связей между сущностями.

Пояснение к выбору ключей.

### Критерии оценивания

Критерий	Балл
Дан неверный ответ/ответ отсутствует	0
Дан неполный ответ	5
Дан верный ответ	10

### Задание 9.

Всего баллов: 10.

Создать Use Case диаграмму для описания функциональных требований по теме «Библиотека». Для 2 ролей Библиотекарь и Читатель

Ответ:

Use Case - диаграмма, описывающая функциональные требования по теме.

Конкретные задачи: регистрация читателя; выдача книг; прием книг; направление уведомление о задолжности.

Таблица, описывающая задачи и подзадачи для использования ее в системе управления проектом.

### Критерии оценивания

Критерий	Балл
Дан неверный ответ/ответ отсутствует	0
Дан неполный ответ	5
Дан верный ответ	10

## ВАРИАНТ 2

### Задание 1.

Всего баллов: 10.

Дан фрагмент программного кода на языке программирования Python. Что будет выведено?

```
text = "Питон - мощный язык"
result = text.split()[:-1]
print(" ".join(result))
```

Ответ: язык мощный – Питон

### Критерии оценивания

Критерий	Балл
Дан неверный ответ/ответ отсутствует	0
Дан верный ответ	10

### Задание 2.

Всего баллов: 10.

Дан фрагмент программного кода на языке программирования Python. Что будет выведено?

```
a = [1, 2, 3]
b = a
c = a[:]
b[0] = 99
c[1] = 88
print(a, b, c)
```

Ответ: [99, 2, 3] [99, 2, 3] [1, 88, 3]

### Критерии оценивания

Критерий	Балл
Дан неверный ответ/ответ отсутствует	0
Дан верный ответ	10

### Задание 3.

Всего баллов: 10.

Дан фрагмент программного кода на языке программирования Python. Дополните функцию `special_sum`, которая должна возвращать сумму всех чисел в списке, которые **либо чётные, либо больше 10**.

```
def special_sum(numbers):
    """
    Возвращает сумму чисел, которые являются чётными ИЛИ больше 10.
    """
    total = 0
    for num in numbers:
        if _____: # Пропущено условие
            total += num
    return total

# Тесты
print(special_sum([1, 2, 3, 14, 5, 6])) # Должно вывести: 22 (2 + 14 + 6)
print(special_sum([10, 11, 12, 13]))    # Должно вывести: 46 (10 + 11 + 12 + 13)
```

Ответ: num % 2 == 0 or num > 10

или

```
num > 10 or num % 2 == 0
```

#### Критерии оценивания

Критерий	Балл
Дан неверный ответ/ответ отсутствует	0
Дан верный ответ	10

#### Задание 4.

Всего баллов: 10.

В данном задании представлена программа (на языке программирования Python) с ошибкой. Необходимо исправить одну строку в коде и вписать ее полностью в ответ.

```
# Программа должна вычислять сумму всех чётных чисел от 1 до N
def sum_even_numbers(n):
    total = 0
    for i in range(1, n):
        if i % 2 == 0:
            total = i
    return total

# Тест
print("Сумма чётных чисел от 1 до 10:", sum_even_numbers(10))
```

Ответ: total += i или total = i + 1

Ответ:

total += i	# Программа должна вычислять сумму всех чётных чисел от 1 до N def sum_even_numbers_correct(n): total = 0 for i in range(1, n + 1): if i % 2 == 0: total += i return total  print("Сумма чётных от 1 до 10:", sum_even_numbers_correct(10)) # 30
ИЛИ	total = total + i  # Программа должна вычислять сумму всех чётных чисел от 1 до N def sum_even_numbers_correct(n): total = 0 for i in range(1, n + 1): if i % 2 == 0: total = total + i return total  print("Сумма чётных от 1 до 10:", sum_even_numbers_correct(10)) # 30

#### Критерии оценивания

Критерий	Балл
Дан неверный ответ/ответ отсутствует	0
Дан верный ответ	10

### **Задание 5.**

Всего баллов: 20.

Опишите структуру алгоритма классификации KNN (K-Nearest Neighbors). В каких задачах можно использовать данный алгоритм?

Ответ:

K-Nearest Neighbors (К-ближайших соседей) – это один из самых простых и интуитивно понятных алгоритмов классификации в машинном обучении.

#### **Структура алгоритма**

1. Выбор параметра K – количество ближайших соседей, которые будут учитываться при классификации.
2. Вычисление расстояний между новым объектом и всеми объектами в обучающей выборке. Обычно используется евклидово расстояние.
3. Сортировка объектов по возрастанию расстояния до нового объекта.
4. Выбор K ближайших соседей из отсортированного списка.
5. Определение класса нового объекта на основе классов его ближайших соседей (чаще всего используется метод голосования).

#### **Применение алгоритма**

KNN можно использовать в следующих задачах:

- Классификация текстов: определение тональности отзывов, категоризация документов.
- Распознавание образов: классификация изображений, определение категорий товаров на фото.
- Рекомендательные системы: рекомендации фильмов, книг, товаров на основе предпочтений пользователей.
- Медицинская диагностика: анализ симптомов для постановки диагноза.
- Анализ финансовых данных: прогнозирование курсов валют, оценка кредитных рисков.
- Геоинформационные системы: определение принадлежности точек к определённым категориям (например, тип почвы, тип застройки).
- Биологическая классификация: определение видов по их характеристикам.

#### **Критерии оценивания**

Критерий	Балл
Дан неверный ответ/ответ отсутствует	0
Дан неполный ответ	10
Дан верный ответ	20

### **Задание 6.**

Всего баллов: 10.

По заданной матрице ошибок для задачи классификации вычислите следующие метрики оценки общей эффективности модели:

- Recall (Полнота)
- Specificity (Специфичность)

	Predicted: No	Predicted: Yes
Actual: No	TN= 18	FP=15
Actual: Yes	FN=5	TP=72

Ответ: Recall = 0.935

Specificity = 0.545

#### Критерии оценивания

Критерий	Балл
Дан неверный ответ/ответ отсутствует	0
Дан неполный ответ	5
Дан верный ответ	10

**Задание 7.**

Всего баллов: 10.

Опишите основные понятия в обработке естественного языка (Natural Language Processing), такие как токенизация (Tokenization), стемминг (Stemming), лемматизация (Lemmatization)

Ответ:

Токенизация – разделение текста на более мелкие единицы (токены). Это могут быть отдельные слова, фразы или предложения.

Стемминг – способ упрощения слов и получения их основы, при котором отбрасываются окончания и суффиксы. Цель стемминга – упростить и стандартизировать слова, что помогает улучшить производительность задач NLP, например поиска информации, классификации текста.

Лемматизация – процесс приведения слова к его базовой, начальной форме (лемме). Цель лемматизации – помочь компьютеру «понимать» различные грамматические формы слова как одно и то же слово.

#### Критерии оценивания

Критерий	Балл
Дан неверный ответ/ответ отсутствует	0
Дан неполный ответ	5

Дан верный ответ	10
------------------	----

### Задание 8.

Всего баллов: 10.

1. Создать ERD, описывающую модель БД логического уровня для хранения данных по теме «Магазин».

Пояснить типы связей между сущностями и выбор ключей

Ответ:

ERD- диаграмма, описывающая модель базы данных.

Пояснение к типам связей между сущностями.

Пояснение к выбору ключей.

Критерии оценивания

Критерий	Балл
Дан неверный ответ/ответ отсутствует	0
Дан неполный ответ	5
Дан верный ответ	10

### Задание 9.

Всего баллов: 10.

Создать Use Case диаграмму для описания функциональных требований по теме «Магазин». Для 2 ролей Продавец и Покупатель.

Ответ:

Use Case - диаграмма, описывающая функциональные требования по теме.

Конкретные задачи: консультирование покупателя по ценам на товары и свойства товаров; оформление корзины заказов; оплата заказа; возврат товара; оформление претензий на обслуживание; оценка качества процесса обслуживания.

Таблица, описывающая задачи и подзадачи для использования ее в системе управления проектом.

Критерии оценивания

Критерий	Балл
Дан неверный ответ/ответ отсутствует	0
Дан неполный ответ	5
Дан верный ответ	10

