

УТВЕРЖДАЮ
 Декан факультета

«___» _____ 20__

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ ТЕХНОЛОГИИ ПОСТРОЕНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ ПРОГРАММНЫХ СИСТЕМ

Направление/специальность подготовки	24.05.06 Системы управления летательными аппаратами
Специализация/профиль/программа подготовки	Системы управления беспилотными летательными аппаратами
Уровень высшего образования	Специалитет
Форма обучения	Очная
Факультет	А Ракетно-космическая техника
Выпускающая кафедра	А5 ДИНАМИКА И УПРАВЛЕНИЕ ПОЛЁТОМ ЛЕТАТЕЛЬНЫХ АППАРАТОВ
Кафедра-разработчик рабочей программы	Н2 Программная инженерия и интеллектуальные системы

КУРС	СЕМЕСТР	ОБЩАЯ ТРУДОЁМКОСТЬ (ЗАЧЕТНЫХ ЕДИНИЦ)	ЧАСЫ (по наличию видов занятий)									ВИД ПРОМЕЖУТОЧНОГО КОНТРОЛЯ
			ОБЩАЯ ТРУДОЁМКОСТЬ	АУДИТОРНЫЕ ЗАНЯТИЯ				САМОСТОЯТЕЛЬНАЯ РАБОТА				
				ВСЕГО	ЛЕКЦИИ	ЛАБОРАТОРНЫЙ ПРАКТИКУМ	ПРАКТИЧЕСКИЕ ЗАНЯТИЯ	ВСЕГО	КУРСОВОЙ ПРОЕКТ	КУРСОВАЯ РАБОТА	ДРУГИЕ ВИДЫ САМОСТ. РАБОТЫ	
1	2	4	144	68	34	0	34	76	0	0	76	ЭКЗ.

ЛИСТ СОГЛАСОВАНИЯ

**РАБОЧАЯ ПРОГРАММА СОСТАВЛЕНА В СООТВЕТСТВИИ С ТРЕБОВАНИЯМИ ФЕДЕРАЛЬНОГО
ГОСУДАРСТВЕННОГО ОБРАЗОВАТЕЛЬНОГО СТАНДАРТА ВЫСШЕГО ОБРАЗОВАНИЯ (ФГОС ВО)**

24.05.06 Системы управления летательными аппаратами

год набора группы: 2026

Программу составил:

Кафедра Н2 Программная инженерия и интеллектуальные системы
Лестенко Никита Александрович, старший преподаватель

Программа рассмотрена
на заседании кафедры-разработчика
рабочей программы **Н2 Программная инженерия и интеллектуальные системы**

Заведующий кафедрой Семенова Е.Г., д.т.н., проф.

Программа рассмотрена
на заседании выпускающей кафедры

А5 ДИНАМИКА И УПРАВЛЕНИЕ ПОЛЕТОМ ЛЕТАТЕЛЬНЫХ АППАРАТОВ

Заведующий кафедрой Петрова И.Л., к.т.н., доц.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ТЕХНОЛОГИИ ПОСТРОЕНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ ПРОГРАММНЫХ СИСТЕМ

Разделы рабочей программы

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ
2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВО
3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ
4. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ
5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ
6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Приложения к рабочей программе дисциплины

- Приложение 1. Аннотация рабочей программы
- Приложение 2. Технологии и формы обучения
- Приложение 3. Фонды оценочных средств

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины является формирование следующих компетенций:

ПК-93 — Способен генерировать новые идеи для решения задач цифровой экономики, абстрагироваться от стандартных моделей, перестраивать сложившиеся способы решения задач, выдвигать альтернативные варианты действий с целью выработки новых оптимальных алгоритмов

ОПК-2 — Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности

ОПК-9 — Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения

Формированию компетенций служит достижение следующих результатов образования:

ПК-93

знания:

знать алгоритмы и структуры данных для эффективного решения задач;;

умения:

уметь применять языки программирования для реализации программных решений;;

навыки:

обладать навыком отладки и оптимизации кода для повышения производительности и надежности программ;.

ОПК-2

знания:

знать основы алгоритмизации и структуры данных для эффективного решения задач.;;

умения:

умение применять языки программирования для реализации программных решений.;;

навыки:

обладать навыком чтения и анализа чужого кода, документации к библиотекам, работы с официальной документацией Python.;.

ОПК-9

знания:

знать этапы жизненного цикла разработки программного обеспечения для интеллектуальных систем;

умения:

уметь разрабатывать алгоритмы для решения практических задач в области обработки данных и интеллектуального анализа;

навыки:

обладать навыком создания прикладных программных модулей, которые могут быть встроены в реальные информационно-коммуникационные системы.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВО

Дисциплина **ТЕХНОЛОГИИ ПОСТРОЕНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ ПРОГРАММНЫХ СИСТЕМ** является дисциплиной **обязательной части блока 1** программы подготовки по направлению *24.05.06 Системы управления летательными аппаратами*.

Содержание дисциплины является логическим продолжением дисциплин: **ВВЕДЕНИЕ В КИБЕРФИЗИЧЕСКИЕ СИСТЕМЫ**.

Содержание дисциплины является основой для освоения дисциплин: **ИНТЕЛЛЕКТУАЛЬНЫЕ АВТОМАТИЗИРОВАННЫЕ ПРОИЗВОДСТВА, МЕТОДЫ ОБРАБОТКИ БОЛЬШИХ ДАННЫХ В СЛОЖНЫХ ТЕХНИЧЕСКИХ СИСТЕМАХ, МОДЕЛИРОВАНИЕ ПРОЦЕССОВ УПРАВЛЕНИЯ ИНФОРМАЦИОННО-ИЗМЕРИТЕЛЬНЫМИ И УПРАВЛЯЮЩИМИ СИСТЕМАМИ, ЦИФРОВАЯ ТРАНСФОРМАЦИЯ ПРОИЗВОДСТВЕННЫХ СИСТЕМ: МЕТОДЫ И ТЕХНОЛОГИИ, МЕТОДЫ ПРИНЯТИЯ ОПТИМИЗАЦИОННЫХ РЕШЕНИЙ**.

Предварительные компетенции, сформированные у обучающегося до начала изучения дисциплины:

- ОПК-2 — Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности
- ПК-94 — Способен к управлению информацией и данными, поиску источников информации и данных, восприятию, анализу, запоминанию и передаче информации с использованием цифровых средств, а также с помощью алгоритмов при работе с полученными из различных источников данными с целью эффективного использования полученной информации для решения задач

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины составляет 4 з.е., 144 ч.

3.1. Содержание (дидактика) дисциплины

КУРС	СЕМЕСТР	Наименование разделов и дидактических единиц	ВСЕГО	Аудиторные занятия в контактной форме			Самостоятельная работа студентов	Формируемая компетенция, %		
				ВСЕГО	Лекции	Практические занятия		ПК-93	ОПК-2	ОПК-9
1	2	Раздел 1. Введение в Python. Основы синтаксиса, типы данных, ввод/вывод, линейные алгоритмы. Понятия информация и информационные технологии, программирование, язык и среда программирования. Этапы написания программы. Алгоритмы и блок-схемы. Общий обзор языка Python. Основы работы в IDLE. Базовый синтаксис и основные команды языка Python. Понятие переменной, требования к их наименованию. Целочисленные данные, вещественные данные, строковые данные, логические данные.	22	12	6	6	10	20	20	20
1	2	Раздел 2. Ветвления и циклы. Условные операторы, циклы for, while, рекуррентные вычисления, оператор break.	34	16	8	8	18	20	20	20
1	2	Раздел 3. Структуры данных. Списки, кортежи, словари, множества. Базовые алгоритмы обработки. Функции и методы работы со списками, кортежами, словарями, множествами. Срезы.	16	8	4	4	8	20	20	20
1	2	Раздел 4. Функции и модули. Определение функций, аргументы, область видимости, lambda-функции, импорт библиотек.	38	20	10	10	18	20	20	20
1	2	Раздел 5. Библиотеки для интеллектуального анализа данных. Изучение библиотек math, pandas, numpy, mathprolib, random. Работа с данными с использованием библиотек math, pandas, numpy, mathprolib, random.	34	12	6	6	22	20	20	20
Всего за 2 семестр			144	68	34	34	76	100	100	100
Всего по дисциплине			144	68	34	34	76	100	100	100

3.2. Аудиторный практикум

№ п/п	Номер и наименование раздела дисциплины	Тема практического занятия	Объем, ауд. часов
1	Раздел 1. Введение в Python.	Установка среды (VS Code, PyCharm, IDLE Python, Jupiter). Первая программа: ввод/вывод, типы данных, вычисления.	6
2	Раздел 2. Ветвления и циклы.	Решение задач на ветвления и циклы(числовые ряды, табуляция функций)	8
3	Раздел 3. Структуры данных.	Обработка списков: поиск, сортировка, срезы. Работа со словарями.	4
4	Раздел 4. Функции и модули.	Создание пользовательских функций. Передача параметров. Функции высшего порядка.	10
5	Раздел 5. Библиотеки для интеллектуального анализа данных.	Визуализация данных (mathprolib). Вычисления с numpy и pandas. Генерация случайных данных.	6
Всего за 2 семестр			34

3.3. Самостоятельная работа студента (СРС)

№ п/п	Номер и наименование раздела дисциплины	Содержание учебного задания	Объем, часов
1	Раздел 1. Введение в Python.	Изучение предусмотренной программой дисциплины рекомендуемой литературы	4
2		Оформление отчета	2
3		Подготовка к практическим занятиям	4
4	Раздел 2. Ветвления и циклы.	Изучение предусмотренной программой дисциплины рекомендуемой литературы	4
5		Подготовка к практическим занятиям	12
6		Оформление отчета	2
7	Раздел 3. Структуры данных.	Подготовка к практическим занятиям	4

8		Оформление отчета	2
9		Изучение предусмотренной программой дисциплины рекомендуемой литературы	2
10	Раздел 4. Функции и модули.	Подготовка к практическим занятиям	14
11		Оформление отчета	2
12		Изучение предусмотренной программой дисциплины рекомендуемой литературы	2
13	Раздел 5. Библиотеки для интеллектуального анализа данных.	Подготовка к практическим занятиям	18
14		Оформление отчета	2
15		Изучение предусмотренной программой дисциплины рекомендуемой литературы	2
Всего за 2 семестр			76

4. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

СЕМЕСТР	НЕДЕЛИ СЕМЕСТРА																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
2			ИПЗ			ДР	ИПЗ		ИПЗ	ДР			ИПЗ			ДР	ИПЗ

Условные обозначения:

- ДР – диагностическая работа;
- ИПЗ – индивидуальное практическое задание.

Текущий контроль успеваемости студентов проводится в дискретные временные интервалы в следующих формах:

- диагностическая работа;
- индивидуальное практическое задание.

Промежуточная аттестация проводится в формах:

- экзамен.

5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

5.1. Основная литература по дисциплине:

1. Д. Бейдер. . Чистый Python. Тонкости программирования для профи. Санкт-Петербург: Питер, 2021, эл. рес.
2. Д. Ю. Фёдоров. . Программирование на языке высокого уровня Python. Москва: Юрайт, 2023, эл. рес.
3. Н. А. Лестенко, И. Д. Мамаев. Технология обработки данных для предметно-ориентированных задач на языке Python. БГТУ "ВОЕНМЕХ" им. Д.Ф. Устинова, 2024, 12 экз.
4. Н. А. Лестенко, И. Д. Мамаев. Технология обработки данных для предметно-ориентированных задач на языке Python. БГТУ "ВОЕНМЕХ" им. Д.Ф. Устинова, 2024, эл. рес.
5. Ф. Шолле. . Глубокое обучение на Python. Санкт-Петербург: Питер, 2021, эл. рес.

5.2. Дополнительная литература по дисциплине:

не требуется.

5.3. Периодические издания:

1. Прикладная информатика.

5.4. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины, электронные библиотечные системы:

1. <https://e.lanbook.com/> — ЭБС Лань;;
2. <https://urait.ru/> — Образовательная платформа «Юрайт». Для вузов и ссузов.;

Современные профессиональные базы данных:

1. <https://rusneb.ru> – Национальная электронная библиотека (НЭБ);
2. <https://cyberleninka.ru/> - Научная электронная библиотека «Киберленинка»;
<http://www.rfbr.ru/rffi/ru/library> - Полнотекстовая электронная библиотека Российского фонда фундаментальных исследований.

Информационные справочные системы:

1. Техэксперт – Информационный портал технического регулирования: Нормы, правила, стандарты РФ;
2. http://library.voenmeh.ru/jirbis2/index.php?option=com_irbis&view=irbis&Itemid=457 - БД ГОСТов собственной генерации БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова;
3. <http://www.consultant.ru/>- КонсультантПлюс- информационный портал правовой информации.

5.5. Программное обеспечение:

1. Набор средств трансляции, компоновки, отладки и выполнения Python 3.x с интегрированной средой разработки IDLE.

5.6. Информационные технологии:

взаимодействие с обучающимися посредством ЭИОС Moodle БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова.

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1. Лекционные занятия:

специализированные требования по оборудованию отсутствуют; аудитория с посадочными местами по количеству студентов; доска.

6.2. Практические занятия:

1. Проектор;
2. Набор средств трансляции, компоновки, отладки и выполнения Python 3.x с интегрированной средой разработки IDLE.

6.3. Прочее:

1. рабочее место преподавателя, оснащенное компьютером с доступом в Интернет;
2. рабочие места студентов, оснащенные компьютерами с доступом в Интернет, предназначенные для работы в электронной образовательной среде.

Аннотация рабочей программы

Дисциплина **ТЕХНОЛОГИИ ПОСТРОЕНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ ПРОГРАММНЫХ СИСТЕМ** является дисциплиной **обязательной части блока 1** программы подготовки по направлению *24.05.06 Системы управления летательными аппаратами*. Дисциплина реализуется на факультете *Н Робототехника и инновационная инженерия* БГТУ "ВОЕНМЕХ" им. Д.Ф. Устинова кафедрой *Н2 Программная инженерия и интеллектуальные системы*.

Дисциплина нацелена на формирование *компетенций*:

ПК-93 Способен генерировать новые идеи для решения задач цифровой экономики, абстрагироваться от стандартных моделей, перестраивать сложившиеся способы решения задач, выдвигать альтернативные варианты действий с целью выработки новых оптимальных алгоритмов;

ОПК-2 Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности;

ОПК-9 Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения.

Содержание дисциплины охватывает круг вопросов, связанных с разработкой алгоритмов, созданием программного обеспечения и автоматизацией процессов для решения практических задач с помощью вычислительных систем.

Программой дисциплины предусмотрены следующие **виды контроля**:

Текущий контроль успеваемости студентов проводится в дискретные временные интервалы в следующих формах:

- диагностическая работа;
- индивидуальное практическое задание.

Промежуточная аттестация проводится в формах:

- экзамен.

Общая трудоемкость освоения дисциплины составляет **4 з.е., 144 ч.** Программой дисциплины предусмотрены лекционные занятия (**34 ч.**), практические занятия (**34 ч.**), самостоятельная работа студента (**76 ч.**).

ТЕХНОЛОГИИ И ФОРМЫ ОБУЧЕНИЯ

Рекомендации по освоению дисциплины для студента

Трудоемкость освоения дисциплины составляет 144 ч., из них 68 ч. аудиторных занятий, и 76 ч., отведенных на самостоятельную работу студента.

Рекомендации по распределению учебного времени по видам самостоятельной работы и разделам дисциплины приведены в таблице.

Контроль освоения дисциплины производится в соответствии с Положением о текущем, рубежном контроле успеваемости и промежуточной аттестации обучающихся.

Формы контроля и критерии оценивания приведены в приложении 3 к Рабочей программе.

Наименование работы	Рекомендуемая литература	Трудоемкость, час.
Раздел 1. Введение в Python.		
Изучение предусмотренной программой дисциплины рекомендуемой литературы	Н. А. Лестенко, И. Д. Мамаев. Технология обработки данных для предметно-ориентированных задач на языке Python: БГТУ "ВОЕНМЕХ" им. Д.Ф. Устинова, 2024 (1-6)	4
Оформление отчета		2
Подготовка к практическим занятиям		4
Итого по разделу 1		10
Раздел 2. Ветвления и циклы.		
Изучение предусмотренной программой дисциплины рекомендуемой литературы	Ф. Шолле. . Глубокое обучение на Python: Санкт-Петербург: Питер, 2021 (1-2)	4
Подготовка к практическим занятиям		12
Оформление отчета		2
Итого по разделу 2		18
Раздел 3. Структуры данных.		
Подготовка к практическим занятиям	Д. Бейдер. . Чистый Python. Тонкости программирования для профи: Санкт-Петербург: Питер, 2021 (2-4)	4
Оформление отчета		2
Изучение предусмотренной программой дисциплины рекомендуемой литературы		2
Итого по разделу 3		8
Раздел 4. Функции и модули.		
Подготовка к практическим занятиям	Д. Ю. Фёдоров. . Программирование на языке высокого уровня Python: Москва: Юрайт, 2023 (3-4)	14
Оформление отчета		2
Изучение предусмотренной программой дисциплины рекомендуемой литературы		2
Итого по разделу 4		18
Раздел 5. Библиотеки для интеллектуального анализа данных.		
Подготовка к практическим занятиям	Н. А. Лестенко, И. Д. Мамаев. Технология обработки данных для предметно-ориентированных задач на языке Python: БГТУ "ВОЕНМЕХ" им. Д.Ф. Устинова, 2024 (4)	18
Оформление отчета		2
Изучение предусмотренной программой дисциплины		2

рекомендуемой литературы		
Итого по разделу 5		22

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Фонд оценочных средств, позволяющие оценить результаты обучения по данной дисциплине, включают в себя:

- диагностическая работа
- индивидуальное практическое задание;
- экзамен.

Критерии оценивания

Диагностическая работа

Диагностическая работа проводится в форме теста в ЭИОС Moodle:

- при правильном ответе менее чем на 60% вопросов - не аттестация;
- при правильном ответе на 60% вопросов и более - аттестация.

Индивидуальное практическое задание

Перечень индивидуальных заданий:

1. Введение в Python.
2. Ветвления и циклы.
3. Структуры данных.
4. Функции и модули.
5. Библиотеки для интеллектуального анализа данных.

Индивидуальное задание считается выполненным и защищенным успешно при условии:

- наличия программного кода, реализующего поставленную задачу;
- наличия отчета;
- защиты индивидуального задания по комплекту тестовых вопросов для защиты индивидуальных заданий, размещенного в УМК дисциплины.

Экзамен

График контрольных мероприятий предусматривает выполнение студентом пяти индивидуальных заданий, каждое из которых может быть оценено согласно технологической карте дисциплины. Также в рамках курса проводятся три диагностические работы, каждая из которых оценивается на 10 баллов при условии успешного прохождения.

Оценка за экзамен выставляется по сумме результатов контрольных мероприятий, проводимых в течение семестра. Максимальная сумма баллов за семестр – 100 баллов.

Набранная итоговая сумма баллов пересчитывается в оценку по следующей схеме:

- 60 – 74 баллов – удовлетворительно;
- 75 – 84 балла - хорошо;
- 85– 100 баллов – отлично.

В случае несогласия студента с оценкой, выставляемой согласно БРС, может быть проведён экзамен, вопросы к которому располагаются в УМК дисциплины. В этом случае экзамен проходит по билетам. Каждый билет содержит два вопроса и задачу. Корректный ответ на один вопрос и частично решенную задачу - "удовлетворительно", Полные ответы на два вопроса и частично решенная задача - "хорошо", либо неполные ответы на два вопроса и полностью корректно решенная задача - "хорошо" . Развернутый ответ на два вопроса и полностью корректно решенная задача- "отлично"

Паспорт фонда оценочных средств

КУРС	СЕМЕСТР	Наименование разделов и дидактических единиц	ВСЕГО	Аудиторные занятия в контактной форме			Самостоятельная работа студентов	Формируемая компетенция, %			НАИМЕНОВАНИЕ ОЦЕНОЧНОГО СРЕДСТВА
				ВСЕГО	Лекции	Практические занятия		ПК-93	ОПК-2	ОПК-9	
1	2	Раздел 1. Введение в Python.	22	12	6	6	10	20	20	20	Индивидуальное практическое задание
1	2	Раздел 2. Ветвления и циклы.	34	16	8	8	18	20	20	20	Индивидуальное практическое задание
1	2	Раздел 3. Структуры данных.	16	8	4	4	8	20	20	20	Индивидуальное практическое задание
1	2	Раздел 4. Функции и модули.	38	20	10	10	18	20	20	20	Индивидуальное практическое задание
1	2	Раздел 5. Библиотеки для интеллектуального анализа данных.	34	12	6	6	22	20	20	20	Индивидуальное практическое задание
Всего за 2 семестр			144	68	34	34	76	100	100	100	
Всего по дисциплине			144	68	34	34	76	100	100	100	

Оценочные материалы по дисциплине ТЕХНОЛОГИИ ПОСТРОЕНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ ПРОГРАММНЫХ СИСТЕМ

ПК-93 - Способен генерировать новые идеи для решения задач цифровой экономики, абстрагироваться от стандартных моделей, перестраивать сложившиеся способы решения задач, выдвигать альтернативные варианты действий с целью выработки новых оптимальных алгоритмов

- № 1 Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа
Дан список `nums = [1, 2, 3, 4, 5]`. Каким будет результат выполнения `[x for x in nums if x % 2 == 0 else 0]`?
- Варианты ответа:
- [0, 2, 0, 4, 0]
 - [2, 4]
 - [0, 0, 0, 0, 0]
 - Синтаксическая ошибка
- № 2 Прочитайте текст и запишите развернутый обоснованный ответ
Чем отличается кортеж (tuple) от списка (list) при передаче в функцию, которая должна гарантировать неизменность переданных данных для последующего интеллектуального анализа?
- № 3 Прочитайте текст и запишите развернутый обоснованный ответ
Допустим, вы используете библиотеку `sklearn.pipeline.Pipeline` для построения конвейера обработки данных. В одном из шагов кастомного трансформера вы обращаетесь к глобальной переменной, которая изменяется в другом модуле во время выполнения `fit()`. При повторном вызове `predict()` с теми же данными результаты оказываются разными, хотя трансформер не содержит случайных компонент. Какая фундаментальная особенность Python (связанная с изменяемыми объектами и областью видимости) приводит к такому эффекту, и как переписать трансформер, чтобы обеспечить детерминированность?
- № 4 Прочитайте текст и установите соответствие
Сопоставьте фрагмент кода слева с его результатом или поведением справа (при условии, что `import pandas as pd` выполнен).

Левый столбец

1. `df[df['x'] > 0]`
2. `df['x'].apply(lambda v: v if v>0 else 0)`
3. `for i in range(len(df)): if df.loc[i, 'x'] < 0: df.loc[i, 'x'] = 0`

Правый столбец

- А. Замена всех значений меньше 0 на 0 с помощью цикла
- Б. Фильтрация строк, где значение в столбце 'x' положительное
- В. Векторизованная замена отрицательных значений на 0
- Г. Поэлементное преобразование с помощью лямбда-функции

- № 5 Прочитайте текст и установите соответствие
Сопоставьте понятие или механизм слева с его описанием справа.

Левый столбец

Правый столбец

Левый столбец**Правый столбец**

1. Область
видимости
(scope)

А. Механизм, позволяющий функции «запоминать» переменные из внешней функции даже после её завершения

2. Замыкание
(closure)

Б. Словарь, который хранит все глобальные переменные модуля

3. globals()

В. Правила, определяющие, где переменная доступна (локальная, объемлющая, глобальная, встроенная)

Г. Источник трудноуловимых ошибок, так как объект сохраняет своё состояние между вызовами функции

№ 6 Прочитайте текст и установите последовательность

Дан фрагмент кода с ошибкой в порядке операций. Расположите строки в правильном порядке, чтобы программа вычисляла факториал числа 5 рекурсивно и корректно обрабатывала случай нуля:

Перемешанные строки:

A. else:

B. def factorial(n):

C. return 1

D. if n == 0:

E. print(factorial(5))

F. return n * factorial(n-1)

№ 7 Прочитайте текст и установите последовательность

Выполняется следующий код. Расположите промежуточные состояния DataFrame в том порядке, в котором они возникают в процессе выполнения цепочки операций:

```
import pandas as pd
```

```
df = pd.DataFrame({'A': [1, 2, None, 4], 'B': [5, None, 7, 8]})
```

```
df = df.dropna().assign(C=lambda x: x['A'] + x['B']).query('C > 10').reset_index(drop=True)
```

Варианты состояний (условные описания):

DataFrame с колонками A, B и новой колонкой C, содержащей сумму A+B, только для строк без пропусков

Исходный DataFrame с пропусками (None) в A и B

DataFrame только с теми строками, где C > 10, сброшенным индексом

DataFrame после удаления всех строк, содержащих хотя бы один None

№ 8 Прочитайте текст, выберите правильные ответы и запишите аргументы, обосновывающие выбор ответов

Какие из следующих утверждений о сравнении pandas.Series и numpy.ndarray являются верными?

Варианты ответа:

Оба поддерживают векторные операции (+, -, *, /)

Series может иметь индекс произвольного типа (не только целочисленный)

ndarray автоматически выравнивает данные по индексу при арифметических операциях

В Series можно хранить разнородные типы данных в одном объекте

№ 9 Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа

Что произойдёт при выполнении кода?

```
def outer():
```



```
x = 10

def inner():

    print(x)

    x = 20

inner()

outer()
```

Варианты ответа:

Выведет 10

Выведет 20

Выведет 10 и затем 20

Ошибка UnboundLocalError

№ 10 Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа

Нужно объединить два DataFrame: df1 (сотрудники) и df2 (отделы) по ключу 'dept_id'. В df1 есть строки с dept_id, отсутствующими в df2, и вы хотите сохранить ВСЕ строки из df1, даже если нет соответствия в df2. Какой параметр how в pd.merge() нужно использовать?

Варианты ответа:

how='inner'

how='outer'

how='left'

how='right'

№ 11 Прочитайте текст, выберите правильные ответы и запишите аргументы, обосновывающие выбор ответов

Какие из следующих выражений корректно создают список [0, 2, 4, 6, 8] в Python?

Варианты ответа:

list(range(0, 10, 2))

[i for i in range(10) if i % 2 == 0]

[0, 2, 4, 6, 8]

list(filter(lambda x: x % 2 == 0, range(10)))

№ 12 Прочитайте текст, выберите правильные ответы и запишите аргументы, обосновывающие выбор ответов

Какие из следующих утверждений об области видимости (scope) в Python являются верными?

Варианты ответа:

Переменная, определённая внутри функции, доступна глобально после её вызова

Ключевое слово global позволяет изменять глобальную переменную внутри функции

Переменные, определённые в цикле for, доступны после цикла на том же уровне вложенности

nonlocal используется для изменения переменной из объемлющей (не глобальной) области видимости

ОПК-2 - Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности

№ 1 Прочитайте текст и запишите развернутый обоснованный ответ

Как с помощью цикла for и условного оператора if можно преобразовать исходный список чисел, заменив все отрицательные элементы на их абсолютные значения, не создавая при этом новый список?

№ 2 Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа

Сколько раз выполнится тело цикла?

```
i = 5
```

```
while i > 0:
```

```
    if i == 3:
```

```
        break
```

```
    i -= 1
```

1. 2

2. 3

3. 4

4. Бесконечное количество

№ 3 Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа

Какая операция на словаре (dict) имеет в среднем временную сложность O(1)?

Варианты ответа:

1. list(d.keys())

2. d[ключ] (доступ по ключу)

3. d.items() (получение представления)

4. Сортировка словаря по ключам

№ 4 Прочитайте текст, выберите правильные ответы и запишите аргументы, обосновывающие выбор ответов

Какие из следующих утверждений о типах данных в Python являются верными?

Варианты ответа:

1. tuple — изменяемый тип данных

2. list поддерживает добавление элементов с помощью метода .append()

3. str является неизменяемым (immutable) типом

4. set может содержать в качестве элементов другие set

№ 5 Прочитайте текст, выберите правильные ответы и запишите аргументы, обосновывающие выбор ответов

Какие из следующих конструкций в Python могут привести к бесконечному циклу при определённых условиях?

Варианты ответа:

1. for i in range(10): pass

2. while True: break

3. while x < 100: x -= 1 (при начальном x = 50)

4. for i in iter(int, 1): pass

№ 6 Прочитайте текст, выберите правильные ответы и запишите аргументы, обосновывающие выбор ответов

Какие операции над списком (list) имеют среднюю временную сложность $O(1)$?

Варианты ответа:

1. Доступ к элементу по индексу (lst[i])
2. Поиск элемента по значению (x in lst)
3. Добавление элемента в конец (.append())
4. Вставка элемента в начало (.insert(0, x))

№ 7 Прочитайте текст и запишите развернутый обоснованный ответ

Представьте, что вы обрабатываете большой DataFrame (pandas) с миллионами строк. Вам нужно заменить все значения в столбце «temp» по правилу: если значение > 100, оставить как есть; если значение < -50, заменить на -50; иначе заменить на среднее арифметическое соседних 5 строк (с отступом в 2 строки в каждую сторону). Почему использование явного цикла for i in range(len(df)) с ветвлениями будет крайне неэффективным, и какой более продвинутый подход (на основе векторизации и оконных функций) вы предложите?

№ 8 Прочитайте текст и установите соответствие

Сопоставьте тип данных слева с его характеристикой справа.

Тип данных	Характеристика
1. list	А. Неизменяемый, хешируемый, часто используется как ключ словаря
2. tuple	Б. Изменяемый, позволяет быстро добавлять/удалять элементы с конца, но медленный поиск по значению
3. dict	В. Неупорядоченная коллекция уникальных элементов, поддерживает операции теории множеств Г. Хранит пары «ключ-значение», поиск по ключу за $O(1)$ в среднем

№ 9 Прочитайте текст и установите соответствие

Сопоставьте конструкцию или подход слева с его типичным применением справа.

Левый столбец	Правый столбец
1. if-elif-else	А. Повторение блока кода фиксированное количество раз
2. while	Б. Выход из цикла досрочно при выполнении условия
3. for	В. Разветвление логики по нескольким взаимоисключающим условиям Г. Повторение, пока условие истинно (когда число итераций неизвестно заранее)

№ 10 Прочитайте текст и установите последовательность

Расположите этапы выполнения следующего фрагмента кода в правильном порядке от первого шага к последнему:

```
def is_even(n):
```

```
    return n % 2 == 0
```

```
numbers = [1, 2, 3, 4]
```

```
result = []
```

```
for x in numbers:
```

```
    if is_even(x):
```

```
result.append(x)

print(result)
```

Варианты последовательности:

1. Вызов функции `is_even(1)`
2. Создание пустого списка `result`
3. Вывод `[2, 4]` на экран
4. Определение функции `is_even`
5. Присваивание переменной `numbers` списка `[1, 2, 3, 4]`
6. Вызов функции `is_even(2)` и добавление 2 в `result`
7. Проверка условия `if` для `x = 3`
8. Завершение цикла

№ 11 Прочитайте текст и установите последовательность

Расположите действия в том порядке, в котором они обычно выполняются при загрузке и первичной обработке CSV-файла с использованием `pandas`:

Варианты последовательности:

1. `3 → 2 → 4 → 5 → 1 → 6 → 7`

№ 12 Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа

Что произойдёт при выполнении следующего кода?

```
a = [1, 2, 3]
```

```
b = a
```

```
b.append(4)
```

```
print(a)
```

Варианты ответа:

`[1, 2, 3]`

`[1, 2, 3, 4]`

`None`

Ошибка: `b` не определён

ОПК-9 - Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения

№ 1 Прочитайте текст и запишите развернутый обоснованный ответ

Вы работаете с `DataFrame`, содержащим столбцы `price` и `quantity`. Вам нужно добавить новый столбец `total` как произведение `price * quantity`, но только для строк, где `quantity > 0`. Если `quantity <= 0`, то `total` должен быть равен `NaN`. Как наиболее эффективно и читаемо сделать это в `pandas`, избегая явного цикла? Приведите пример кода и поясните, почему такой подход быстрее.

№ 2 Прочитайте текст и установите последовательность

Расположите строки кода в правильном порядке, чтобы программа находила максимальное значение в списке `numbers` без использования встроенной функции `max()` (алгоритм линейного поиска).

Перемешанные строки:

- A. `for num in numbers:`
- B. `numbers = [3, 1, 4, 1, 5, 9, 2]`
- C. `max_val = numbers[0]`
- D. `print(f"Максимум: {max_val}")`

E. if num > max_val:
F. max_val = num

№ 3 Прочитайте текст и установите последовательность

Расположите этапы выполнения следующего кода в порядке обращения к переменным (поиск по LEGB-правилу).

```
x = 'global'

def outer():

    x = 'outer'

    def inner():

        x = 'inner'

        print(x)

    inner()

outer()
```

Перемешанные события (обращения к x):

1. Обращение к x внутри print в inner()
2. Поиск в локальной области inner — находит 'inner'
3. Вывод 'inner'
4. После завершения inner — возврат в outer
5. Создание локальной переменной x в outer со значением 'outer'
6. Создание локальной переменной x в inner со значением 'inner'
7. Определение глобального x = 'global'

№ 4 Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа

Вопрос:

Что будет выведено?

```
a = [1, 2]

b = a

a = a + [3]

print(b)
```

Варианты:

[1, 2, 3]
[1, 2]
[1, 2, 3, 3]

Ошибка

№ 5 Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа

Какое значение примет s после цикла?

```
s = 0

for i in range(1, 4):

    s += i
```

Варианты: 1) 3, 2) 4, 3) 6, 4) 10

№ 6 Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа

Какая функция из `matplotlib.pyplot` создаёт круговую диаграмму?

Варианты:

1. `plot()`
2. `pie()`
3. `bar()`
4. `hist()`

№ 7 Прочитайте текст, выберите правильные ответы и запишите аргументы, обосновывающие выбор ответов

Какие из следующих операций над множеством `set` в Python являются допустимыми? (Выберите все.)

Варианты:

1. Добавление элемента (`add`)
2. Удаление элемента (`remove`)
3. Доступ по индексу (`s[0]`)
4. Проверка на вхождение (`elem in s`)

№ 8 Прочитайте текст, выберите правильные ответы и запишите аргументы, обосновывающие выбор ответов

Какие из следующих вызовов функции `def func(a, b=2, *args): pass` являются корректными? (Выберите все.)

Варианты:

1. `func(1)`
2. `func(1, 3)`
3. `func(1, b=4, 5)`
4. `func(1, 2, 3, 4)`

№ 9 Прочитайте текст, выберите правильные ответы и запишите аргументы, обосновывающие выбор ответов

Какие из следующих вызовов функции `def func(a, b=2, *args): pass` являются корректными? (Выберите все.)

Варианты:

1. `func(1)`
2. `func(1, 3)`
3. `func(1, b=4, 5)`
4. `func(1, 2, 3, 4)`

№ 10 Прочитайте текст и запишите развернутый обоснованный ответ

Представьте, что у вас есть большой текстовый файл (несколько ГБ), каждая строка которого содержит целое число. Напишите **описание генератора** `read_integers(file_path)`, который читает файл построчно, преобразует каждую строку в целое число и возвращает его, а также пропускает строки, которые не являются целыми числами (логируя ошибку в отдельный файл). Объясните, почему генератор лучше, чем чтение всего файла в список, с точки зрения потребления памяти.

№ 11 Прочитайте текст и установите соответствие

Сопоставьте выражение (левый столбец) с его результатом (правый столбец).

Левый столбец

1. `3 * 'a' + 'b'`
2. `'a' in 'abc'`

Правый столбец

- A. `'aaab'`
- B. `True`

Левый столбец**Правый столбец**

- | | |
|-----------------|---------|
| 3. 'abc'[1:3] | В. 'bc' |
| 4. len('hello') | Г. 5 |

№ 12 Прочитайте текст и установите соответствие

Сопоставьте имя модели из scikit-learn (левый столбец) с типовой задачей (правый столбец).

Левый столбец**Правый столбец**

- | | |
|---------------------------|-----------------------------------|
| 1. LinearRegression | А. Классификация |
| 2. LogisticRegression | Б. Регрессия (предсказание числа) |
| 3. KMeans | В. Кластеризация |
| 4. RandomForestClassifier | Г. Классификация (ансамбль) |