

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Якушин Владимир Андреевич
Должность: ректор, д.ю.н., профессор
Дата подписания: 06.10.2025
Уникальный программный ключ:
a5427c2559e1ff4b007ed9b1994671e27053e0dc

Министерство науки и высшего образования РФ
Образовательная автономная некоммерческая организация
высшего образования
«Волжский университет имени В.Н. Татищева» (институт)

УТВЕРЖДАЮ

Ректор Якушин В.А.
от 24.04.2025г.

Рабочая программа

Программирование

Направление подготовки 09.03.02 Информационные системы и технологии

Квалификация (степень) выпускника – бакалавр

Форма обучения – очная, заочная, очно-заочная

Тольятти, 2025 г.

Рабочая программа **Программирование** составлена с требованиями ФГОС, ВО, ОПОП по направлению подготовки 09.03.02 Информационные системы и технологии (уровень высшего образования: бакалавриат) и учебного плана.

Одобрено Учебно-методическим советом вуза

протокол № 4/25 от 24.04.2025г

Председатель УМС

к.п.н. И.И. Муртаева

1. ПЕРЕЧЕНЬ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

В результате освоения дисциплины у обучающихся должны быть сформированы следующие общепрофессиональные компетенции и профессиональные компетенции:

Наименование компетенции	Код компетенции
Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий	ОПК-6

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Данная учебная дисциплина относится к обязательной части образовательной программы 09.03.02 «Информационные системы и технологии».

В таблице 1 представлен перечень компетенций с указанием перечня дисциплин, формирующих эти компетенции согласно учебному плану ОПОП

Таблица 1

Код компетенции	Наименование компетенции, формируемой в рамках освоения дисциплины	Предшествующие дисциплины, формирующие указанную компетенцию	Последующие дисциплины, формирующие указанную компетенцию
ОПК-6.	Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий		Учебная практика. Ознакомительная практика. Защита выпускной квалификационной работы, включая подготовку к процедуре защиты и процедуру защиты

* в качестве этапа формирования компетенций используются номера семестров согласно учебного плана ОПОП

Перечень планируемых результатов обучения, соотнесенных с планируемыми результатами освоения основной профессиональной образовательной программы направления подготовки, представлен в таблице:

Код и наименование компетенции	Код и наименование индикатора достижения компетенции
ОПК-6. Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий	ОПК-6.1. Составляет алгоритмы, пишет и отлаживает коды на языке программирования или СУБД. ОПК-6.2. Использует языки программирования и современные программные среды разработки информационных систем и технологии для автоматизации бизнес-процессов, решения прикладных задач различных классов. ОПК-6.3. Проводит тестирование работоспособности программы.

3. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

ОЧНАЯ ФОРМА ОБУЧЕНИЯ

Вид учебной работы	Всего	Семестр			
		1	2	3	4
Общая трудоёмкость дисциплины	576 час 16 з.е.	180 час 5 з.е.	108 час 3 з.е.	144 час 4 з.е.	144 час 4 з.е.
Контактная работа с преподавателем (всего)	256	64	64	64	64
В том числе:					
Лекции	128	32	32	32	32
Практические / семинарские занятия	-	-	-	-	-
Лабораторные занятия	128	32	32	32	32
Консультации	-	-	-	-	-
Самостоятельная работа (всего)	284	116	44	80	44
В том числе (если есть):					
Курсовой проект / работа	20				20
Расчетно-графическая работа	-	-	-	-	-
Контрольная работа	-	-	-	-	-
Реферат / эссе / доклад	-	-	-	-	-
Иное					
Вид промежуточной аттестации (зачет, экзамен)	Экзамен	зачет	зачет	зачет	экзамен
Контроль	36				(36)

ЗАОЧНАЯ ФОРМА

Вид учебной работы	Всего	Семестр			
		1	2	3	4
Общая трудоёмкость дисциплины	576 час 16 з.е.	180 час 5 з.е.	108 час 3 з.е.	144 час 4 з.е.	144 час 4 з.е.
Контактная работа с преподавателем (всего)	48	12	12	12	12
В том числе:					
Лекции	24	6	6	6	6
Практические / семинарские занятия	-	-	-	-	-
Лабораторные занятия	24	6	6	6	6
Консультации	-	-	-		
Самостоятельная работа (всего)	456	168	60	132	96
В том числе (если есть):					
Курсовой проект / работа	20				20
Расчетно-графическая работа	-	-	-	-	-
Контрольная работа	-	-	-	-	-
Реферат / эссе / доклад	-	-	-	-	-
Иное					

Вид учебной работы	Всего	Семестр			
		1	2	3	4
Общая трудоёмкость дисциплины	576 час 16 з.е.	180 час 5 з.е.	108 час 3 з.е.	144 час 4 з.е.	144 час 4 з.е.
Вид промежуточной аттестации (зачет, экзамен)	экзамен	Зачет	Зачет	Зачет	экзамен (36)

ОЧНО-ЗАОЧНАЯ ФОРМА

Вид учебной работы	Всего	Семестр			
		1	2	3	4
Общая трудоёмкость дисциплины	576 час 16 з.е.	180 час 5 з.е.	108 час 3 з.е.	144 час 4 з.е.	144 час 4 з.е.
Контактная работа с преподавателем (всего)	96	24	24	24	24
В том числе:					
Лекции	24	12	12	12	12
Практические / семинарские занятия	-	-	-	-	-
Лабораторные занятия	24	12	12	12	12
Консультации	-	-	-		
Самостоятельная работа (всего)	492	156	84	120	84
<i>В том числе (если есть):</i>					
<i>Курсовой проект / работа</i>	20				20
<i>Расчетно-графическая работа</i>	-	-	-	-	-
<i>Контрольная работа</i>	-	-	-	-	-
<i>Реферат / эссе / доклад</i>	-	-	-	-	-
<i>Иное</i>					
Вид промежуточной аттестации (зачет, экзамен)	экзамен	Зачет	Зачет	Зачет	экзамен (36)

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1. КАЛЕНДАРНО-ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

ОЧНАЯ ФОРМА ОБУЧЕНИЯ

№ п/п	Тема	Количество часов на				Форма контроля
		лекции	практические /семинарские занятия	лаборат орные занятия	самостоятел ьную работу	
Семестр 1						
1	Классификация языков программировани	2			13	тест АСТ

	я. Основное назначение языка программирования Python.					
2	Введение в программирование на Python. Символы языка, структура программы. Переменные, константы. Объявление, инициализация.	4			15	тест АСТ
3	Операции языка. Программирование Python	4		4	15	тест АСТ отчет по лабораторной работе
4	Операторы языка. Условные операторы (if, switch)	6		6	19	тест АСТ отчет по лабораторной работе
5	Операторы цикла (for, while, do while).	4		6	19	тест АСТ, отчет по лабораторной работе
6	Одномерные массивы	6		8	17	тест АСТ, отчет по лабораторной работе
7	Двумерные массивы	6		8	18	тест АСТ, отчет по лабораторной работе
Итого по 1 семестру		32		32	116	зачет
Семестр 2						
8	Сортировка массивов	4		4	4	тест АСТ, отчет по лабораторной работе
9	Указатели. Указатели на простые переменные	2		4	5	тест АСТ, отчет по лабораторной работе
10	Указатели на массивы	2			5	тест АСТ отчет по лабораторной работе
11	Структуры. Объединения.	2		4	5	тест АСТ отчет по лабораторной работе

12	Указатели на структуры	2		2	5	тест АСТ отчет по лабораторной работе
13	Функции работы со строками.	4		2	5	тест АСТ отчет по лабораторной работе
14	Функции. Объявление, определение, вызов функции. Передача параметров функции main.	6			5	тест АСТ
15	Работа с файлами. Текстовые файлы	6		8	5	тест АСТ отчет по лабораторной работе
16	Работа с файлами. Бинарные файлы	6		8	5	тест АСТ отчет по лабораторной работе
Итого по 2 семестру		32		32	44	зачет
Семестр 3						
Раздел Динамические структуры данных						
17	Машинное обучение. Группировка. Сортировка. Корреляция. Визуализация данных	4		4	14	тест АСТ отчет по лабораторной работе
18	Машинное обучение. Пропуски данных. Выбросы и аномалии	4			14	тест АСТ
19	Машинное обучение. Кластеризация	5		7	13	тест АСТ отчет по лабораторной работе
20	Машинное обучение. Задача предсказания и регрессия	5		7	13	тест АСТ отчет по лабораторной работе
21	Машинное обучение. Алгоритмы	5		7	13	тест АСТ отчет по лабораторной работе

	классификации. Задача классификации. Линейный классификатор					ой работе
22	Машинное обучение. Вероятностные алгоритмы	5		7	13	тест АСТ отчет по лабораторн ой работе
23	Машинное обучение. Селекция признаков.	4			13	тест АСТ
Итого по 3 семестру		32		32	80	зачет
Семестр 4						
Раздел Объектно-ориентированное программирование						
24	Реализация объектно- ориентированного программировани я на языке Python	14		14	4	тест АСТ отчет по лабораторн ой работе
25	Наследование	4		4	4	тест АСТ отчет по лабораторн ой работе
26	Виртуальные функции и абстрактные классы	6		6	4	тест АСТ отчет по лабораторн ой работе
27	Стандартная библиотека шаблонов языка Python	4		4	5	тест АСТ отчет по лабораторн ой работе
28	Обработка исключительных ситуаций в Python	4		4	5	тест АСТ отчет по лабораторн ой работе
	Курсовая работа				20	
Итого по 4 семестру		32		32	44	Экзамен (36)

ЗАОЧНАЯ ФОРМА ОБУЧЕНИЯ

№ п/п	Тема	Количество часов на				Форма контроля
		лекции	практические /семинарские занятия	лаборат орные занятия	самостоятел ьную работу	
Семестр 1						
1	Классификация языков программировани я. Основное	0,5			24	тест АСТ

	назначение языка программирования Python.					
2	Введение в программирование на Python. Символы языка, структура программы. Переменные, константы. Объявление, инициализация.	0,5			24	тест АСТ
3	Операции языка. Программирование Python	1			24	тест АСТ
4	Операторы языка. Условные операторы (if, switch)	1		1	24	тест АСТ
5	Операторы цикла (for, while, do while).	1		1	24	тест АСТ
6	Одномерные массивы	1		2	24	тест АСТ, отчет по лабораторной работе
7	Двумерные массивы	1		2	24	тест АСТ, отчет по лабораторной работе
Итого по 1 семестру		6		6	168	зачет
Семестр 2						
8	Сортировка массивов	0,5		2	6	тест АСТ, отчет по лабораторной работе
9	Указатели. Указатели на простые переменные	0,5		2	6	тест АСТ,
10	Указатели на массивы	0,5			7	тест АСТ
11	Структуры. Объединения.	0,5			7	тест АСТ
12	Указатели на структуры	1			7	тест АСТ
13	Функции работы со строками.	1		1	7	тест АСТ
14	Функции. Объявление,	0,5			7	тест АСТ

	определение, вызов функции. Передача параметров функции main.					
15	Работа с файлами. Текстовые файлы	1		1	7	тест АСТ, отчет по лабораторн ой работе
16	Работа с файлами. Бинарные файлы	0,5			7	тест АСТ, отчет по лабораторн ой работе
Итого по 2 семестру		6		6	60	зачет
Семестр 3						
Раздел Динамические структуры данных						
17	Машинное обучение. Группировка. Сортировка. Корреляция. Визуализация данных	2		2	19	тест АСТ отчет по лабораторн ой работе
18	Машинное обучение. Пропуски данных. Выбросы и аномалии	1		2	19	тест АСТ
19	Машинное обучение. Кластеризация	1		2	19	тест АСТ отчет по лабораторн ой работе
20	Машинное обучение. Задача предсказания и регрессия	0,5			19	тест АСТ отчет по лабораторн ой работе
21	Машинное обучение. Алгоритмы классификации. Задача классификации. Линейный классификатор	0,5			19	тест АСТ отчет по лабораторн ой работе
22	Машинное обучение. Вероятностные алгоритмы	0,5			20	тест АСТ отчет по лабораторн ой работе

23	Машинное обучение. Селекция признаков.	0,5			20	тест АСТ
Итого по 3 семестру		6		6	132	зачет
Семестр 4						
Раздел Объектно-ориентированное программирование						
24	Реализация объектно-ориентированного программирования на языке Python	2		2	15	тест АСТ отчет по лабораторной работе
25	Наследование	2		2	15	тест АСТ отчет по лабораторной работе
26	Виртуальные функции и абстрактные классы	1			15	тест АСТ отчет по лабораторной работе
27	Стандартная библиотека шаблонов языка Python	0,5			15	тест АСТ отчет по лабораторной работе
28	Обработка исключительных ситуаций в Python	0,5		2	16	тест АСТ отчет по лабораторной работе
	Курсовая работа				20	
Итого по 4 семестру		6		6	96	Экзамен (36)

ОЧНО-ЗАОЧНАЯ ФОРМА ОБУЧЕНИЯ

№ п/п	Тема	Количество часов на				Форма контроля
		лекции	практические /семинарские занятия	лаборат орные занятия	самостоятел ьную работу	
Семестр 1						
1	Классификация языков программировани я. Основное назначение языка программировани я Pyhton.	1			22	тест АСТ
2	Введение в программировани е на Pyhton. Символы языка, структура	1			22	тест АСТ

	программы. Переменные, константы. Объявление, инициализация.					
3	Операции языка. Программирование на Python	2		2	22	тест АСТ
4	Операторы языка. Условные операторы (if, switch)	2		2	22	тест АСТ
5	Операторы цикла (for, while, do while).	2		2	22	тест АСТ
6	Одномерные массивы	2		4	23	тест АСТ, отчет по лабораторн ой работе
7	Двумерные массивы	2		4	23	тест АСТ, отчет по лабораторн ой работе
Итого по 1 семестру		12		12	156	зачет
Семестр 2						
8	Сортировка массивов	1		2	9	тест АСТ, отчет по лабораторн ой работе
9	Указатели. Указатели на простые переменные	1		4	9	тест АСТ,
10	Указатели на массивы	1			9	тест АСТ
11	Структуры. Объединения.	1			9	тест АСТ
12	Указатели на структуры	1			9	тест АСТ
13	Функции работы со строками.	2		2	9	тест АСТ
14	Функции. Объявление, определение, вызов функции. Передача параметров функции main.	2			10	тест АСТ
15	Работа с файлами.	2		4	10	тест АСТ, отчет по

	Текстовые файлы					лабораторной работе
16	Работа с файлами. Бинарные файлы	1			10	тест АСТ, отчет по лабораторной работе
Итого по 2 семестру		12		12	84	зачет
Семестр 3						
Раздел Динамические структуры данных						
17	Машинное обучение. Группировка. Сортировка. Корреляция. Визуализация данных	3		2	17	тест АСТ отчет по лабораторной работе
18	Машинное обучение. Пропуски данных. Выбросы и аномалии	1		2	17	тест АСТ
19	Машинное обучение. Кластеризация	2		2	17	тест АСТ отчет по лабораторной работе
20	Машинное обучение. Задача предсказания и регрессия	1			17	тест АСТ отчет по лабораторной работе
21	Машинное обучение. Алгоритмы классификации. Задача классификации. Линейный классификатор	2			17	тест АСТ отчет по лабораторной работе
22	Машинное обучение. Вероятностные алгоритмы	1			20	тест АСТ отчет по лабораторной работе
23	Машинное обучение. Селекция признаков.	1			18	тест АСТ
Итого по 3 семестру		12		12	120	зачет
Семестр 4						
Раздел Объектно-ориентированное программирование						

24	Реализация объектно-ориентированного программирования на языке Python	4		2	12	тест АСТ отчет по лабораторной работе
25	Наследование	4		2	13	тест АСТ отчет по лабораторной работе
26	Виртуальные функции и абстрактные классы	2		2	13	тест АСТ отчет по лабораторной работе
27	Стандартная библиотека шаблонов языка Python	1		2	13	тест АСТ отчет по лабораторной работе
28	Обработка исключительных ситуаций в Python	1		4	13	тест АСТ отчет по лабораторной работе
	Реализация объектно-ориентированного программирования на языке Python			2	20	
Итого по 4 семестру		12		12	84	Экзамен (36)

4.2. КРАТКОЕ СОДЕРЖАНИЕ ЛЕКЦИОННОГО КУРСА

1 семестр

Тема 1. КЛАССИФИКАЦИЯ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ. ОСНОВНОЕ НАЗНАЧЕНИЕ ЯЗЫКА ПРОГРАММИРОВАНИЯ Python.

Введение в программирование. Основы алгоритмизации. Алгоритмизация и требования к алгоритму. Способы записи алгоритмов. Критерии качества программы.

Тема 2. ВВЕДЕНИЕ В ПРОГРАММИРОВАНИЕ НА Python. СИМВОЛЫ ЯЗЫКА, СТРУКТУРА ПРОГРАММЫ. ПЕРЕМЕННЫЕ, КОНСТАНТЫ. ОБЪЯВЛЕНИЕ, ИНИЦИАЛИЗАЦИЯ.

Программирование линейных алгоритмов. Алфавит языка. Классификация типов. Стандартные типы: порядковые и вещественные. Переменные и идентификаторы. Целые типы. Вещественные типы. Стандартные функции. Оператор присваивания. Структура программы.

Тема 3. ОПЕРАЦИИ ЯЗЫКА ПРОГРАММИРОВАНИЯ Python.

Унарные, бинарные операции, тернарная операция языка программирования Python.

Тема 4. ОПЕРАТОРЫ ЯЗЫКА. УСЛОВНЫЕ ОПЕРАТОРЫ (IF, SWITCH)

Разработка, исполнение, отладка, программирование разветвляющихся алгоритмов. Условный оператор if. Сокращенный оператор if. Составной оператор if. Оператор варианта case. Оператор безусловного перехода goto.

Тема 5. ОПЕРАТОРЫ ЦИКЛА (FOR, WHILE, DO WHILE)

Разработка, исполнение, отладка алгоритмов с одним циклом. Проектирование программ циклической структуры. Оператор цикла с параметром. Оператор цикла с предусловием. Оператор цикла с постусловием. Итерационные циклы. Проектирование алгоритмов и программ со структурой вложенных циклов.

Тема 6. ОДНОМЕРНЫЕ МАССИВЫ

Понятие массива. Разработка, исполнение, отладка алгоритмов с массивами. Обработка массивов данных. Описание массива. Одномерные массивы.

Тема 7. ДВУМЕРНЫЕ МАССИВЫ

Двумерные массивы. Ввод-вывод массивов. Примеры программирования задач с использованием массивов.

2 семестр

Тема 8. СОРТИРОВКА МАССИВОВ

Алгоритмы сортировки массивов. Алгоритм сортировки «пузырьком», быстрая сортировка, метод Шелла. Примеры программирования задач на сортировку массивов данных.

Тема 9. УКАЗАТЕЛИ. УКАЗАТЕЛИ НА ПРОСТЫЕ ПЕРЕМЕННЫЕ

Понятия «указатель». Примеры программирования задач с использованием указателей на простые переменные.

Тема 10. УКАЗАТЕЛИ НА МАССИВЫ

Работа с указателями на массивы. Массивы указателей. Примеры программирования задач с использованием указателей на массивы.

Тема 11. СТРУКТУРЫ. ОБЪЕДИНЕНИЯ

Понятие «Структуры» в языке программирования Pyhton. Определение, описание структуры. Массивы структур. Понятие «Объединения» в языке программирования Pyhton. Определение, описание объединения. Примеры программирования задач с использованием Структур и Объединений.

Тема 12. УКАЗАТЕЛИ НА СТРУКТУРЫ

Структуры и указатели. Работа с памятью. Массивы указателей. Примеры программирования задач с указателей и массивов указателей на Структуры.

Тема 13. ФУНКЦИИ РАБОТЫ СО СТРОКАМИ

Библиотечные файлы. Функции работы со строками и функции работы с экраном. Примеры программирования задач с функциями работы со строками.

Тема 14. ФУНКЦИИ. ОБЪЯВЛЕНИЕ, ОПРЕДЕЛЕНИЕ, ВЫЗОВ ФУНКЦИИ. ПЕРЕДАЧА ПАРАМЕТРОВ ФУНКЦИИ.

Функции. Объявление, определение, вызов функции. Формальные и фактические параметры. Передача в функцию массивов, структур. Прототипы функций. Параметры и передача параметров функции main. Работа с командной строкой. Рекурсии. Понятие рекурсии. Техника построения рекурсивных алгоритмов. Формы рекурсий. Простая линейная рекурсия. Параллельная и взаимная рекурсии. Функции с переменным числом параметров.

Тема 15. РАБОТА С ФАЙЛАМИ. ТЕКСТОВЫЕ ФАЙЛЫ.

Создание и преобразование файлов. Файлы прямого доступа. Определение файлового типа. Процедуры и функции обработки файлов. Текстовые файлы. Примеры программирования задач с текстовыми файлами

Тема 16. РАБОТА С ФАЙЛАМИ. БИНАРНЫЕ ФАЙЛЫ.

Функции для создания и работы с бинарными файлами. Примеры программирования задач с бинарными файлами.

3 семестр

Тема 17. МАШИННОЕ ОБУЧЕНИЕ. ГРУППИРОВКА. СОРТИРОВКА. КОРРЕЛЯЦИЯ. ВИЗУАЛИЗАЦИЯ ДАННЫХ

Тема 18. МАШИННОЕ ОБУЧЕНИЕ. ПРОПУСКИ ДАННЫХ. ВЫБРОСЫ И АНОМАЛИИ

Тема 19. МАШИННОЕ ОБУЧЕНИЕ. КЛАСТЕРИЗАЦИЯ
Тема 20. МАШИННОЕ ОБУЧЕНИЕ. ЗАДАЧА ПРЕДСКАЗАНИЯ И РЕГРЕССИЯ
Тема 21. МАШИННОЕ ОБУЧЕНИЕ. АЛГОРИТМЫ КЛАССИФИКАЦИИ. ЗАДАЧА
КЛАССИФИКАЦИИ. ЛИНЕЙНЫЙ КЛАССИФИКАТОР
Тема 22. МАШИННОЕ ОБУЧЕНИЕ. ВЕРОЯТНОСТНЫЕ АЛГОРИТМЫ
Тема 23. МАШИННОЕ ОБУЧЕНИЕ. СЕЛЕКЦИЯ ПРИЗНАКОВ

4 семестр

Тема 24. РЕАЛИЗАЦИЯ ОБЪЕКТНО-ОРИЕНТИРОВАННОГО
ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ Python

Объекты и классы. Процедурное и объектно-ориентированное программирование. Понятие класса. Определение и реализация класса. Деструкторы и конструкторы классов. Члены-данные и члены-методы классов. Уровни доступа класса.

Создание и использование объектов класса. Работа с указателем `this`. Создание массивов объектов. Абстрактные типы данных.

Работа с классами. Перегрузка операций. Использование дружественных классов. Автоматическое преобразование и приведение типов для классов.

Классы и динамическое распределение памяти. Конструкторы копирования и перегруженные операторы присваивания. Конструирование объекта в динамической памяти. Статические члены класса. Использование указателей на объекты.

Тема 25. НАСЛЕДОВАНИЕ

Наследование классов. Защита доступа при наследовании. Преобразование типов вверх и вниз.

Тема 26. ВИРТУАЛЬНЫЕ ФУНКЦИИ И АБСТРАКТНЫЕ КЛАССЫ

Виртуальные функции и виртуальный деструктор. Статическое и динамическое связывание. Реализация виртуальных функций. Абстрактные базовые классы.

Повторное использование программного кода в Python. Классы, содержащие объекты абстрактных типов в качестве элементов. Виртуальные базовые классы.

Тема 27. СТАНДАРТНАЯ БИБЛИОТЕКА ШАБЛОНОВ ЯЗЫКА Python

Использование шаблонов классов.

Тема 28. ОБРАБОТКА ИСКЛЮЧИТЕЛЬНЫХ СИТУАЦИЙ В Python

Обработка исключительных ситуаций.

4.3. ТЕМАТИКА ЛАБОРАТОРНЫХ ЗАНЯТИЙ

1 семестр

Лабораторная работа №1 Операции языка программирования Си.

Лабораторная работа №2 Операторы языка. Условные операторы (`if`, `switch`)

Лабораторная работа №3 Операторы цикла (`for`, `while`, `do while`)

Лабораторная работа №4 Одномерные массивы

Лабораторная работа №5 Двумерные массивы

2 семестр

Лабораторная работа №6 Сортировка массивов

Лабораторная работа №7 Указатели. Указатели на простые переменные.

Лабораторная работа №8 Указатели на массивы.

Лабораторная работа №9 Структуры. Объединения.

Лабораторная работа №10 Указатели на структуры

Лабораторная работа №11 Функции работы со строками

Лабораторная работа №12 Работа с файлами. Текстовые файлы

Лабораторная работа №13 Работа с файлами. Бинарные файлы

3 семестр

Лабораторная работа № 1 Машинное обучение. Группировка. Сортировка. Корреляция. Визуализация данных
 Лабораторная работа № 2 Машинное обучение. Пропуски данных. Выбросы и аномалии
 Лабораторная работа №3 Кластеризация
 Лабораторная работа №4 Классификация

4 семестр

Лабораторная работа № 5 КЛАССЫ И ОБЪЕКТЫ В C++
 Лабораторная работа № 6 НАСЛЕДОВАНИЕ И ВИРТУАЛЬНЫЕ ФУНКЦИИ
 Лабораторная работа № 7 ИЕРАРХИЯ ОБЪЕКТОВ И ГРУППА. ИТЕРАТОРЫ
 Лабораторная работа № 8 ОБРАБОТКА СОБЫТИЙ
 Лабораторная работа № 9 ПЕРЕГРУЗКА ОПЕРАЦИЙ

5. ПЕРЕЧЕНЬ УЧЕБНО-МЕТОДИЧЕСКОГО ОБЕСПЕЧЕНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

5.1 Основная литература

Зыков, С. В. Программирование : учебник и практикум для вузов / С. В. Зыков. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 285 с. — (Высшее образование). — ISBN 978-5-534-16031-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/560815>

Зыков, С. В. Программирование. Функциональный подход : учебник и практикум для вузов / С. В. Зыков. — 2-е изд. — Москва : Издательство Юрайт, 2025. — 150 с. — (Высшее образование). — ISBN 978-5-534-16942-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/561744>.

Трофимов, В. В. Алгоритмизация и программирование : учебник для вузов / В. В. Трофимов, Т. А. Павловская. — 4-е изд. — Москва : Издательство Юрайт, 2025. — 108 с. — (Высшее образование). — ISBN 978-5-534-20430-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/562040>

Якимов, С. П. Структурное программирование : учебник для вузов / С. П. Якимов. — Москва : Издательство Юрайт, 2025. — 342 с. — (Высшее образование). — ISBN 978-5-534-14885-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/567948>

5.2 Дополнительная литература

Огнева, М. В. Программирование на языке C++: практический курс : учебник для вузов / М. В. Огнева, Е. В. Кудрина, А. А. Казачкова. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 342 с. — (Высшее образование). — ISBN 978-5-534-18949-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/563618>

Федоров, Д. Ю. Программирование на python : учебное пособие для вузов / Д. Ю. Федоров. — 6-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 187 с. — (Высшее образование). — ISBN 978-5-534-19666-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/556864>

5.3. Ресурсы информационно-коммуникационной сети «Интернет»

Адрес Интернет ресурса	Название Интернет ресурса	Режим доступа
http://intuit.ru/	Интернет-университет информационных технологий	Свободный

http://vkit.ru/	Сайт журнала «Вестник компьютерных и информационных технологий»	Свободный
http://ru.wikipedia.org/	Свободная общедоступная мультязычная универсальная интернет-энциклопедия	Свободный

6. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ

Дисциплина «**Программирование**» изучается в течение четырех семестров. При планировании и организации времени, необходимого на изучение обучающимся дисциплины, необходимо придерживаться следующих рекомендаций.

В период между сессиями студенты должны вести конспект лекций, изучать теоретический материал в соответствии с программой курса, выполнять предложенные преподавателем задания для самостоятельной работы, готовиться к сдаче зачета и экзамена, прорабатывая необходимый материал согласно перечню вопросов для подготовки к зачету и экзамену и списку рекомендованной литературы.

Выполнение лабораторных работ относится к числу обязательных видов работ. Перед выполнением работы необходимо внимательно ознакомиться с теоретическим материалом, представленным в методических указаниях к соответствующей лабораторной работе. При необходимости можно воспользоваться рекомендуемой литературой. В ходе выполнения работы необходимо руководствоваться порядком выполнения лабораторной работы и указаниями преподавателя, при этом должны соблюдаться правила техники безопасности. Результатом выполнения работы является отчет, который должен быть аккуратно оформлен и выполнен в соответствии с требованиями, приведенными в методических указаниях.

В указанное преподавателем время обучающиеся защищают отчеты. Защита проводится в виде собеседования по контрольным вопросам, приведенным в методических указаниях. Кроме того, преподаватель может задавать дополнительные вопросы, касающиеся результатов эксперимента, выводов по результатам опытов и т.п. К промежуточной аттестации допускаются обучающиеся, выполнившие все лабораторные работы и защитившие отчеты по ним. При наличии задолженности по лабораторным работам, по согласованию с преподавателем, возможна замена работы по выполнению отчета на реферат по теме соответствующего лабораторного занятия с последующей его защитой.

В последнем семестре изучения дисциплины учебным планом предусмотрен курсовой проект. При получении задания, необходимо внимательно с ним ознакомиться и, в случае возникновения вопросов, задать их преподавателю. Регулярное посещение консультаций, внимательное изучение методических указаний к выполнению курсового проекта, а так же строгое соблюдение графика выполнения проекта позволит избежать ненужных проблем. Оценка за курсовой проект выставляется по результатам его защиты.

В течение семестра и во время сессии основным видом подготовки являются самостоятельные занятия. Они включают в себя изучение вопросов, вынесенных на самостоятельное изучение, оформление отчетов по лабораторным работам, курсовое проектирование, а так же подготовку к промежуточной аттестации.

Систематическая работа в соответствии с программой дисциплины – условие успешного освоения материала.

Методические рекомендации по обучению лиц с ограниченными возможностями здоровья и инвалидов

Освоение дисциплины обучающимися с ограниченными возможностями здоровья может быть организовано как совместно с другими обучающимися, так и в отдельных группах. Предполагаются специальные условия для получения образования обучающимися с ограниченными возможностями здоровья.

Профессорско-педагогический состав знакомится с психолого-физиологическими особенностями обучающихся инвалидов и лиц с ограниченными возможностями здоровья, индивидуальными программами реабилитации инвалидов (при наличии). При необходимости осуществляется дополнительная поддержка преподавания тьюторами, психологами, социальными работниками, прошедшими подготовку ассистентами.

В соответствии с методическими рекомендациями Минобрнауки РФ (утв. 8 апреля 2014 г. N АК-44/05вн) в курсе предполагается использовать социально-активные и рефлексивные методы обучения, технологии социокультурной реабилитации с целью оказания помощи в установлении полноценных межличностных отношений с другими студентами, создании комфортного психологического климата в студенческой группе. Подбор и разработка учебных материалов производятся с учетом предоставления материала в различных формах: аудиальной, визуальной, с использованием специальных технических средств и информационных систем.

Медиа материалы также следует использовать и адаптировать с учетом индивидуальных особенностей обучения лиц с ОВЗ.

Освоение дисциплины лицами с ОВЗ осуществляется с использованием средств обучения общего и специального назначения (персонального и коллективного использования). Материально-техническое обеспечение предусматривает приспособление аудиторий к нуждам лиц с ОВЗ.

Форма проведения аттестации для студентов-инвалидов устанавливается с учетом индивидуальных психофизических особенностей. Для студентов с ОВЗ предусматривается доступная форма предоставления заданий оценочных средств, а именно:

- в печатной или электронной форме (для лиц с нарушениями опорно-двигательного аппарата);
- в печатной форме или электронной форме с увеличенным шрифтом и контрастностью (для лиц с нарушениями слуха, речи, зрения);
- методом чтения ассистентом задания вслух (для лиц с нарушениями зрения).

Студентам с инвалидностью увеличивается время на подготовку ответов на контрольные вопросы. Для таких студентов предусматривается доступная форма предоставления ответов на задания, а именно:

- письменно на бумаге или набором ответов на компьютере (для лиц с нарушениями слуха, речи);
- выбором ответа из возможных вариантов с использованием услуг ассистента (для лиц с нарушениями опорно-двигательного аппарата);
- устно (для лиц с нарушениями зрения, опорно-двигательного аппарата).

При необходимости для обучающихся с инвалидностью процедура оценивания результатов обучения может проводиться в несколько этапов.

7. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ИНФОРМАЦИОННЫХ СПРАВОЧНЫХ СИСТЕМ

При проведении занятий по дисциплине используются следующие программные продукты:

1. Проектор;
2. Windows (для академических организациях, лицензия Microsoft Imagine (ранее MSDN AA, DreamSpark);
3. Anaconda — дистрибутив языков программирования Python (свободное ПО);

4. Доступ к электронным изданиям ЭБС ЮРАЙТ (www.biblio-online.ru).

8. НЕОБХОДИМАЯ МАТЕРИАЛЬНО-ТЕХНИЧЕСКАЯ БАЗА

Оборудование лекционных аудиторий 504, 509, 604, 609: офисная мебель, экран – 1 шт.; проектор – 1 шт.; ПК – 1 шт.

Оборудование аудиторий для лабораторных занятий: ауд. 508, 510: офисная мебель, 7 ПК с доступом в Интернет; ауд. 509, 511: офисная мебель, 8 ПК с доступом в Интернет; ауд. 504, 609: офисная мебель, 10 ПК с доступом в Интернет; ауд. 604: офисная мебель, 9 ПК с доступом в Интернет.

Оборудование аудиторий для самостоятельной работы: читальный зал НТБ: 5 ПК с доступом в Интернет; ауд. 609: 10 ПК с доступом в Интернет.

Разработчик:

Кафедра ИиСУ

(место работы)

**ст. преподаватель
кафедры ИиСУ**

(занимаемая должность)

Е.В. Плюснина

(инициалы, фамилия)

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
ОБРАЗОВАТЕЛЬНАЯ АВТОНОМНАЯ НЕКОММЕРЧЕСКАЯ
ОРГАНИЗАЦИЯ ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОЛЖСКИЙ УНИВЕРСИТЕТ имени В.Н. ТАТИЩЕВА» (институт)**

Фонд оценочных средств

«Программирование»

для направления подготовки

09.03.02 «Информационные системы и технологии»

Квалификация (степень) выпускника – бакалавриат

1 Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

Оценочные средства разработаны для оценки профессиональных компетенций: ОПК-6.

1. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Планируемые результаты освоения основной профессиональной образовательной программы (ОПОП) – компетенции обучающихся определяются требованиями стандарта по направлению подготовки (специальности) и формируются в соответствии с матрицей компетенций ОПОП (Таблица 2)

Планируемые результаты обучения по дисциплине – знания, умения, навыки и (или) опыт деятельности, характеризующие этапы формирования компетенций и обеспечивающие достижение планируемых результатов освоения образовательной программы, формируются в соответствии с картами компетенций ОПОП.

Таблица 1

Планируемые результаты обучения по дисциплине

Код и наименование компетенции	Код и наименование индикатора достижения компетенции
ОПК-6. Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий	ОПК-6.1. Составляет алгоритмы, пишет и отлаживает коды на языке программирования или СУБД. ОПК-6.2. Использует языки программирования и современные программные среды разработки информационных систем и технологии для автоматизации бизнес-процессов, решения прикладных задач различных классов. ОПК-6.3. Проводит тестирование работоспособности программы.

2 Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Результаты обучения по дисциплине «Программирование» направления подготовки 09.03.02 «Информационные системы и технологии» определяются показателями и критериями оценивания сформированности компетенций на этапах их формирования представлены в табл. 2.

Таблица 2

Матрица соответствия оценочных средств запланированным результатам обучения

Компетенции	Оценочные средства			
	Текущий контроль		Промежуточный контроль (зачет)	
	Оценочное средство 1 (лабораторные задания)	Оценочное средство 2	Зачет (вопросы к зачету)	Экзамен (вопросы к экзамену)
ОПК-6	ОПК-6.1. ОПК -6.2. ОПК -6.3.		ОПК-6.1. ОПК -6.2. ОПК -6.3.	ОПК-6.1. ОПК -6.2. ОПК -6.3.

Показатели и критерии оценивания сформированности компетенций (промежуточного контроля)

На этапе промежуточной аттестации используется система оценки успеваемости обучающихся, которая позволяет преподавателю оценить уровень освоения материала обучающимися. Критерии оценивания сформированности планируемых результатов обучения (дескрипторов) представлены в карте компетенции ОПОП.

Форма оценки знаний: оценка - 5 «отлично»; 4 «хорошо»; 3 «удовлетворительно»; 2 «неудовлетворительно». Лабораторные работы, практические занятия, практика оцениваются: «зачет», «незачет». Возможно использование балльно-рейтинговой оценки.

Шкала оценивания:

«Зачет» – выставляется, если сформированность заявленных дескрипторов компетенций на 51 % и более оценивается не ниже «удовлетворительно» при условии отсутствия критерия «неудовлетворительно». Выставляется, когда обучающийся показывает хорошие знания изученного учебного материала; самостоятельно, логично и последовательно излагает и интерпретирует материалы учебного курса; полностью раскрывает смысл предлагаемого вопроса; владеет основными терминами и понятиями изученного курса; показывает умение переложить теоретические знания на предполагаемый практический опыт.

«Отлично» – выставляется, если сформированность заявленных дескрипторов компетенций 85 % более (в соответствии с картами компетенций ОПОП) оценивается критериями «хорошо» и «отлично», при условии отсутствия оценки «неудовлетворительно»: студент показал прочные знания основных положений фактического материала, умение самостоятельно решать конкретные практические задачи повышенной сложности, свободно использовать справочную литературу, делать обоснованные выводы из результатов анализа конкретных ситуаций;

«Хорошо» – выставляется, если сформированность заявленных дескрипторов компетенций на 61 % и более (в соответствии с картами компетенций ОПОП) оценивается критериями «хорошо» и «отлично», при условии отсутствия оценки «неудовлетворительно», допускается оценка «удовлетворительно»: обучающийся показал прочные знания основных положений фактического материала, умение самостоятельно решать конкретные практические задачи, предусмотренные рабочей программой, ориентироваться в рекомендованной справочной литературе, умеет правильно оценить полученные результаты анализа конкретных ситуаций;

«Удовлетворительно» – выставляется, если сформированность заявленных дескрипторов компетенций 51 % и более (в соответствии с картами компетенций ОПОП) оценивается критериями «удовлетворительно», «хорошо» и «отлично»: обучающийся показал знание основных положений фактического материала, умение получить с помощью преподавателя правильное решение конкретной практической задачи из числа предусмотренных рабочей программой, знакомство с рекомендованной справочной литературой;

«Неудовлетворительно» «Незачет» – выставляется, если сформированность заявленных дескрипторов компетенций менее чем 51 % (в соответствии с картами компетенций ОПОП): при ответе обучающегося выявились существенные пробелы в знаниях основных положений фактического материала, неумение с помощью преподавателя получить правильное решение конкретной практической задачи из числа предусмотренных рабочей программой учебной дисциплины.

Ответы и решения обучающихся оцениваются по следующим общим критериям: распознавание проблем; определение значимой информации; анализ проблем; аргументированность; использование стратегий; творческий подход; выводы; общая грамотность.

Соответствие критериев оценивания сформированности планируемых результатов обучения (дескрипторов) системам оценок представлено в табл.

Критерии	Традиционная оценка	Балльно-рейтинговая оценка
5	5	86 - 100
4	4	61-85
3	3	51-60
2 и 1	2, Незачет	0-50
5, 4, 3	Зачет	51-100

Обучающиеся обязаны сдавать все задания в сроки, установленные преподавателем. Оценка «Удовлетворительно» по дисциплине, может выставляться и при неполной сформированности компетенций в ходе освоения отдельной учебной дисциплины, если их формирование предполагается продолжить на более поздних этапах обучения, в ходе изучения других учебных дисциплин.

Показатели и критерии оценки достижений студентом запланированных результатов освоения дисциплины в ходе текущего контроля и промежуточной аттестации

Оценка, уровень	Критерии
«отлично», повышенный уровень	Студент показал прочные знания основных положений фактического материала, умение самостоятельно решать конкретные практические задачи повышенной сложности, свободно использовать справочную литературу, делать обоснованные выводы из результатов анализа конкретных ситуаций
«хорошо», пороговый уровень	Студент показал прочные знания основных положений фактического материала, умение самостоятельно решать конкретные практические задачи, предусмотренные рабочей программой, ориентироваться в рекомендованной справочной литературе, умеет правильно оценить полученные результаты анализа конкретных ситуаций
«удовлетворительно», пороговый уровень	Студент показал знание основных положений фактического материала, умение получить с помощью преподавателя правильное решение конкретной практической задачи из числа предусмотренных рабочей программой, знакомство с рекомендованной справочной литературой
«неудовлетворительно», уровень не сформирован	При ответе студента выявились существенные пробелы в знаниях основных положений фактического материала, неумение с помощью преподавателя получить правильное решение конкретной практической задачи из числа предусмотренных рабочей программой учебной дисциплины

3 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Номер задания	Содержание вопроса	Правильный ответ на задание
1.	Выберите правильный вариант ответа. Текстовые языки, приближенные по словарю и синтаксису к человеческому языку, позволяющие записывать программные конструкции в форме, удобной для человеческого мышления и	А

Номер задания	Содержание вопроса	Правильный ответ на задание
	подобные обычному тексту — конспекту, стенограмме, являются <u>A) языками высокого уровня</u> B) языками визуального программирования C) интеллектуальными языками программирования D) логическими языками	
2.	Выберите правильный вариант ответа. Программирование, когда программа отделена от данных и состоит из последовательности команд, обрабатывающих данные. Данные как правило хранятся в виде переменных. Весь процесс вычисления сводится к изменению их содержимого, относят к A) объектно ориентированному программированию <u>B) процедурному программированию</u> C) визуальному программированию D) логическому программированию	B
3.	Выберите правильный вариант ответа. _____ — модель процесса разработки программного обеспечения, жизненный цикл которой выглядит как поток, последовательно проходящий фазы анализа требований, проектирования, реализации, тестирования, интеграции и поддержки. <u>A) каскадная модель</u> B) инкрементная модель C) спиральная модель D) эволюционная модель	A
4.	Выберите правильный вариант ответа. На рисунке _____ жизненного цикла программного продукта  A) каскадная модель <u>B) инкрементная модель</u> C) спиральная модель D) эволюционная модель	C
5.	Выберите правильный вариант ответа. Характеристику качества программного продукта означающая независимость ПП	B

Номер задания	Содержание вопроса	Правильный ответ на задание
	от технического комплекса системы обработки данных, операционной среды, сетевой технологии обработки данных, специфики предметной области и т.п. относят к характеристикам А) надежности <u>В) мобильности</u> С) эффективности D) эволюционная модель	
6.	Выберите правильный вариант ответа. Для контроля этапов алгоритмизации и программирования применяют тестирование направлено на проверку правильности логики работы алгоритма и выполнения арифметических действий <u>А) Алгоритмическое тестирование</u> В) Аналитическое тестирование С) Содержательное тестирование D) Логическое тестирование	А
7.	Выберите правильный вариант ответа. Проверку совместимости программы с аппаратным обеспечением и прочими software-элементами (различными версиями OS и процессоров) проводят на этапе А) тестирования пользователями В) нагрузочного тестирования <u>С) конфигурационного тестирования</u> D) логическое тестирование	С
8.	Выберите правильный вариант ответа. Тестирование _____ подразумевает, что у разработчика теста есть доступ к исходному коду приложения и он имеет возможность писать код, связанный с библиотеками тестируемого ПО А) чёрного ящика <u>В) белого/прозрачного ящика</u> С) нагрузочное тестирование D) стрессовое тестирование	В
9.	Выберите правильный вариант ответа. Определить результат работы программы Python <pre>A = {1, 2, 3} B = {3, 2, 3, 1} Print (A == B)</pre> А) 2 В) 43 <u>С) True</u> D) False	С

Номер задания	Содержание вопроса	Правильный ответ на задание
	E) Type	
10.	Выберите правильный вариант ответа. Определить результат работы программы Python со строками <pre>>>> S1 = 'spam' >>> S2 = 'eggs' >>> print(S1 + S2)</pre> A) spamspamspam B) spameggs C) ameg D) aeg E) sggemaps	B
11.	Выберите правильный вариант ответа. Определить результат работы программы Python <pre>>>> for i in 'hello world': ... print(i * 2, end='')</pre> A) hello world B) hheellll C) hheellll wwrrlldd D) hheelllloo wwoorrlldd E) world	D
12.	Выберите правильный вариант ответа. Определить количество выводов на экран при работе программы Python <pre>a = 1 while a < 10: print('Цикл выполнен', a, 'раз(a) ') a = a+1 print('Цикл окончен')</pre> A) 12 B) 11 C) 10 D) 9 E) 8	C
13.	Выберите правильный вариант ответа. Определить как изменился список после выполнения программы Python <pre>items = ['foo', 'bar', 'baz'] for (index, elem) in enumerate(items): items[index] = elem + '!'</pre> A) ['foo*', 'bar*', 'baz*']	D

Номер задания	Содержание вопроса	Правильный ответ на задание
	B) ['foo!', 'bar!'] C) ['foo', 'bar', 'baz'] D) ['foo!', 'bar!', 'baz!'] E) ['bar!', 'baz!']	
14.	Выберите правильный вариант ответа. Определить значение переменной pos после работы программы Python <pre> L = ['abc', 'bcd', 'def', 'abd', 'bcd', 'hef', 'inn'] s = str(input("Input string:")) # s - искомая строка pos=-1 # искомая позиция k=0 # текущая позиция в цикле for t in L: # обход цикла if s==t: pos=k break k=k+1 print("pos = ",pos) </pre> A) 8 B) 9 C) 10 D) 11 E) 12	С
15.	Выберите правильный вариант ответа. Определить сколько раз сработает внешний цикл при работе программы Python <pre> i = 1 j = 1 while i < 10: while j < 10: print(i * j, end="\t") j += 1 print("\n") j = 1 i += 1 </pre> A) 8 B) 9 C) 10 D) 11 E) 12	В
16.	Использование логического программирования (Язык Пролог)	Язык Пролог и логическое программирование и широко используются для создания баз знаний и экспертных систем и исследований в сфере искусственного

Номер задания	Содержание вопроса	Правильный ответ на задание
		интеллекта на основе логических моделей баз знаний и логических процедур вывода и принятия решений.
17.	Дайте краткое описание принципу процедурного программирования	Процедурное программирование — программирование на императивном языке, при котором последовательно выполняемые операторы можно собрать в подпрограммы, то есть более крупные целостные единицы кода, с помощью механизмов самого языка.
18.	Опишите понятие линейный алгоритм в современном программировании	Линейный алгоритм - описание действий, которые выполняются однократно в заданном порядке
19.	Опишите понятие циклический алгоритм в современном программировании	Циклический алгоритм - описание действий, которые должны повторяться указанное число раз или пока не выполнено задание
20.	Опишите понятие разветвляющий алгоритм в современном программировании	Разветвляющий алгоритм - алгоритм, в котором в зависимости от условия выполняется либо одна, либо другая последовательность действий
21.	Перечислите виды жизненных циклов программного обеспечения.	Жизненный цикл можно представить в виде моделей. В настоящее время наиболее распространенными являются: каскадная, инкрементная (поэтапная модель с промежуточным контролем) и спиральная модели жизненного цикла.
22.	Перечислите классы программных продуктов по характеру использования и категориям пользователей	Все программы по характеру использования и категориям пользователей можно разделить на два класса: 1) утилитарные программы 2) программные продукты (изделия)
23.	Что такое программный продукт?	Программный продукт — комплекс взаимосвязанных программ для решения определенной проблемы (задачи) массового спроса, подготовленный к реализации как любой другой вид промышленной продукции.
24.	Определите характеристику качества программных продуктов (ПП) - эффективность	Эффективность ПП оценивается как с позиций прямого его назначения — требований пользователя, так и с точки зрения расхода вычислительных ресурсов, необходимых для его эксплуатации.
25.	Перечислите основные этапы решения	1. Постановка задачи

Номер задания	Содержание вопроса	Правильный ответ на задание
	задач с помощью компьютерной техники	2. Формализация задачи. 3. Построение алгоритма. 4. Составление программы на языке программирования. 5. Отладка и тестирование программы. 6. Проведение расчётов и анализ полученных результатов.
26.	Перечислите этапы процесса отладки программного продукта	Отладка проводится в два этапа: - определяется природа и местонахождение подозреваемой ошибки в программе; - фиксируется и исправляется ошибка.
27.	Дайте описание понятию тестирования	Тестирование — это процесс исследования объекта тестирования с целью определить, что программа работает в соответствии с заявленными требованиями с помощью заранее подготовленных проверок.
28.	Перечислите не менее трех видов тестирования в зависимости от объекта тестирования	Три примера можно выбрать из данного списка видов тестирования в зависимости от объекта тестирования: - Функциональное тестирование - Тестирование производительности - Нагрузочное тестирование - Стресс-тестирование - Тестирование стабильности - Конфигурационное тестирование - Юзабилити-тестирование - Тестирование безопасности - Тестирование локализации - Тестирование совместимости.
29.	Определите результат работы программы Python <pre>collection = [1,2,3,4,5,6,7,8,9,10] n = 10 for i in collection: l = n*i print(l)</pre>	Этот код выведет результаты умножения заданного числа n на элементы из переменной collection по очереди: «10, 20, 30, 40, 50, 60, 70, 80, 90, 100».
30.	Определите количество повторов цикла и значение n программного кода Python <pre>n = 1 while n < 6: print('Цикл выполнен', n, 'раз(a)') n = n+1</pre>	Цикл повторится 5 раз, n будет равна 6, условие станет ложным и цикл завершится.
31.	Определите результат работы программного кода Python <pre>def my_function(name): print(f'Привет, {name}') my_function('Вася')</pre>	Результат вызова функции с параметром: Привет, Вася
32.	Определите результат работы	Результат работы программного кода

Номер задания	Содержание вопроса	Правильный ответ на задание
	программного кода Python, если при запросе <i>Введите число: введено число 45</i> <pre>number = int(input("Введите число: ")) if number > 10: print("первая строка") print("вторая строка") print("третья строка") print("Выполняется каждый раз, когда вы запускаете программу") print("Конец")</pre>	Python: первая строка вторая строка третья строка Выполняется каждый раз, когда вы запускаете программу Конец
33.	Дана программа Python для расчета площади и длины окружности круга. Приведите пример работы программы при вводе радиуса -12 <pre>radius = int(input("Введите радиус: ")) if radius >= 0: print("Длина окружности = ", 2 * 3.14 * radius) print("Площадь = ", 3.14 * radius ** 2) else: print("Пожалуйста, введите положительное число")</pre>	Результат работы программного кода Python: Пожалуйста, введите положительное число
34.	Дана программа Python для проверки пароля, введенного пользователем. Приведите пример работы программы при вводе пароля ABS <pre>password = input("Введите пароль: ") if password == "ssh": print("Добро пожаловать") else: print("Доступ запрещен")</pre>	Результат работы программного кода Python: Доступ запрещен
35.	Дана программа двухмерного массива в Python. Определите результат работы программного кода <pre>example_array = [[-1, 0, 0, 1], [2, 3, 5, 8]] print(example_array[0]) print(example_array[1]) print(example_array[0][3])</pre>	Результат работы программного кода Python: <i>[-1, 0, 0, 1]</i> <i>[2, 3, 5, 8]</i> <i>1</i>
36.	Дана программа прохода (итерации) по списку в Python. Определите результат работы программного кода. <pre>my_list = [1, 2, 3, 4, 5] for i in range(len(my_list)): my_list[i] += 5 print(my_list)</pre>	Результат работы программного кода будет следующим: <i>[6, 7, 8, 9, 10]</i>
37.	Перечислите, что включает в себя процесс тестирования программного продукта	Процесс тестирования включает в себя: - тестирование модулей; - тестирование связей между модулями; - системное тестирование - тестирование системы в целом.
38.	Запишите результат работы программы Python <pre>shop = ['яблоки', 'манго',</pre>	бананы

Номер задания	Содержание вопроса	Правильный ответ на задание
	<pre>'морковь ', 'бананы'] print(shop[-1])</pre>	
39.	Запишите какая структура представлена на языке Python <pre>shop = ['яблоки', 'манго', 'морковь ', 'бананы']</pre>	В данном примере на языке .Python описан список
40.	Приведите не менее трех типов данных языка Python относят к неизменяемым	Три примера можно выбрать из данного списка неизменяемых типов данных языка Python относятся: - целые числа, - числа с плавающей точкой, - комплексные числа, - логические переменные, - кортежи, - строки - неизменяемые множества
41.	Какие типы данных языка Python относят к изменяемым?	К изменяемым типам языка Python относятся: 1) списки 2) множества 3) словари
42.	Приведите не менее трех структурированных типов данных языка Python	Три примера можно выбрать из данного списка структурированных типов данных языка Python: - строка - список - кортеж - словарь - множество - файл
43.	Перечислите операторы сравнения языка Python	Всего в Python таких операторов 6: • == — равно; • != — не равно; • > — больше; • < — меньше; • >= — больше или равно; • <= — меньше или равно.
44.	Перечислите арифметические операторы языка Python	Арифметические операторы в Python производят арифметические операции над числами: • + — сложение; • - — вычитание; • * — умножение; • / — деление; • // — целочисленное деление (возвращает только целую часть от деления, значение после запятой отбрасывается); • % — деление по модулю

Номер задания	Содержание вопроса	Правильный ответ на задание
		(возвращает остаток от деления); ** — возведение в степень.
45.	Приведите не менее четырех операторов присваивания языка Python	Четыре примера можно выбрать из данного списка операторов присваивания языка Python: • = — значение правого операнда присвоится левому операнду; • += — сумма левого и правого операнда присвоится левому операнду; • -= — разность левого и правого операнда присвоится левому операнду; • *= — произведение левого и правого операнда присвоится левому операнду; • /= — разделит левый операнд на правый и результат присвоится левому операнду; • //= — результат целочисленного деления левого операнда на правый операнд присвоится левому операнду; • %= — разделит левый операнд на правый по модулю и результат присвоится левому операнду; • **= — возведет левый операнд в степень правого и результат присвоится левому операнду.
46.	Запишите структуру условного оператора языка Python	Структура условного оператора языка Python: <pre> if <условие>: <оператор_1> <оператор_2> <оператор_n> else: <оператор_1> <оператор_2> <оператор_k> </pre>
47.	Запишите структуру оператора выбора языка Python	Структура оператора выбора языка Python: <pre> if <условие>: <операторы> elif <условие>: <операторы> else: <операторы> </pre>
48.	Опишите чем отличается массив от списка	Массив - это структура данных, в

Номер задания	Содержание вопроса	Правильный ответ на задание
	в языке Python?	которой хранятся значения одного типа. В Python это основное различие между массивами и списками.
49.	Какой метод используется на языке Python для добавления элементы в массив?	Для добавления элементов в массив можно использовать метод <code>insert</code> .
50.	Какой метод используется на языке Python для удаления элемента массива?	для удаления элемента массива можно использовать метод <code>pop</code> .
51.	Приведите пример функции в Python	Обычно функция определяется с помощью инструкции <code>def</code> . Определим простейшую функцию: <pre>def add(x, y): return x + y</pre>
52.	Запишите структуру оператора цикла <code>for</code> языка Python	Структура цикла <code>for</code> языка Python: <pre>for [элемент] in [последовательность]: [сделать указанное]</pre>
53.	Приведите пример цикла <code>for</code> в Python, содержащий различные данные разных типов в последовательности.	For может содержать данные разных типов: цифры, слова и пр. Например: <pre>for i in 10, 14, 'первый', 'второй': print(i)</pre>
54.	Запишите структуру оператора цикла <code>while</code> языка Python	Структура цикла <code>while</code> языка Python: <pre>while [условие истинно]: [сделать указанное]</pre>
55.	Приведите пример цикла <code>while</code> в Python, начиная с 0 и меньше 6 по условию. В каждом цикле значение должно увеличиваться на 2.	Программа с использованием цикла <code>while</code> по примеру: <pre>count = 0 while count < 6: print(count) count += 2</pre>
56.	Приведите 4 функции для работы со строками встроенх в интерпретатор Python	Python предоставляет множество функций, которые встроены в интерпретатор. Вот несколько, которые работают со строками: <code>chr()</code> -Преобразует целое число в символ <code>ord()</code> -Преобразует символ в целое число <code>len()</code> -Возвращает длину строки <code>str()</code> -Изменяет тип объекта на <code>string</code>
57.	Приведите структуру операторов цикла <code>for</code> на языке Python.	<pre>for <переменная> in <последовательность>:</pre>

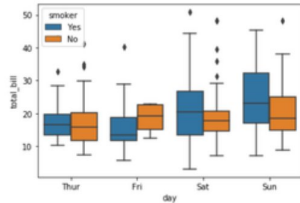
Номер задания	Содержание вопроса	Правильный ответ на задание
		<действие> else: <действие>
58.	Перечислите операторы тождественности используемые в Python.	В Python существует два оператора тождественности: is и is not.
59.	Перечислите логические операторы используемые в Python.	В Python имеются следующие логические операторы: - оператор and (логическое умножение); - оператор or (логическое сложение); - оператор not (логическое отрицание).
60.	Перечислите изменяемые и неизменяемые типы данных в Python.	К изменяемым типам данных в Python относят списки, словари и множества. К неизменяемым типам данных в Python относят числа, строки и кортежи.
61.	Перечислите простые типы данных в Python.	К простым типам данных в Python относят; - числовые значения (например, 4); - строковые значения (например, 'p').
62.	Перечислите составные типы данных в Python.	К составным типам данных в Python относят; - строки (например, 'Привет'); - списки (например, [4, 5, 6]); - кортежи (например, (0, 4, 6, 8)).
63.	Дана программа на Python, на рисунке представлена ее часть. <pre>>>> a = 10 >>> print(a)</pre> Опишите данную часть программы	Переменной, а присвоено значение 10. Значение переменной а выведено на экран
64.	Дана программа на Python и результат ее выполнения. Записать тип данных представленных в этой программе. s = 'Hello, friend. You are my world' print(type(s)) > <class 'str'>	В программе представлена неизменяемая последовательность СТРОКА
65.	Дана программа на Python. Кратко опишите ее работу <pre>def capitalize(String): return String.title() capitalize("shop") # [Shop] capitalize("python programming") # [Python Programming] capitalize("how are you!") # [How Are You!]</pre>	Этот пример используется для превращения каждой первой буквы символов строки в прописную букву.
1.	Выберите правильный вариант ответа. Дан код Python. Каким видом атрибута класса является width и height. Вид используемого атрибута класса class Rectangle:	В

Номер задания	Содержание вопроса	Правильный ответ на задание
	<pre>default_color = "green"</pre> <pre>def __init__(self, width, height): self.width = width self.height = height</pre> А) статический В) динамический С) иерархический D) стратегический	
2.	Выберите правильный вариант ответа. Дан код Python. Каким видом атрибута класса является default_color. Вид используемого атрибута класса <pre>class Rectangle: default_color = "green"</pre> <pre>def __init__(self, width, height): self.width = width self.height = height</pre> А) статический В) динамический С) иерархический D) стратегический	А
3.	Выберите правильный вариант ответа. Поставлена задача. Создать класс автомобилей Python и описать методы __init__() для инициализации его объектов. А) <pre>class Car: def stop_engine(self): if self.engine_on: self.engine_on = False return "Двигатель остановлен." return "Двигатель уже был остановлен."</pre> В) <pre>class Car: pass</pre> С) <pre>class Car: def __init__(self, color, consumption, tank_volume, mileage=0): self.color = color self.consumption = consumption self.tank_volume = tank_volume self.reserve = tank_volume self.mileage = mileage self.engine_on = False</pre> Д) <pre>class Car: def stop_engine(self): if self.engine_on:</pre>	С
4.	Выберите правильный вариант ответа. По предложенному коду Python определите уровень доступа к атрибуту __how_many_times_turned_on <pre>class Phone:</pre>	С

Номер задания	Содержание вопроса	Правильный ответ на задание
	<pre>username = "Kate" _age = 1 __how_many_times_turned_on = 0</pre> A) публичный B) защищенный C) приватный D) общий	
5.	Выберите правильный вариант ответа. По предложенному коду Python определите уровень доступа к атрибуту <code>_age</code> <pre>class Phone: username = "Kate" _age = 1 __how_many_times_turned_on = 0</pre> A) публичный B) защищенный C) приватный D) общий	B
6.	Выберите правильный вариант ответа. По предложенному коду Python определите уровень доступа к атрибуту <code>username</code> <pre>class Phone: username = "Kate" _age = 1 __how_many_times_turned_on = 0</pre> A) публичный B) защищенный C) приватный D) общий	A
7.	Выберите правильный вариант ответа. По предложенному коду Python определите вид полиморфизма <pre>print(len("Programiz")) print(len(["Python", "Java", "C"])) print(len({"Name": "John", "Address": "Nepal"}))</pre> A) полиморфизм в классах B) полиморфизм на примере функции <code>len()</code> C) полиморфизм оператора D) полиморфизм общий	B
8.	Выберите правильный вариант ответа. Представлен код Python. Определите результат работы кода. <pre>class Cat: def __init__(self, name, age): self.name = name self.age = age def info(self):</pre>	B

Номер задания	Содержание вопроса	Правильный ответ на задание
	<pre> print(f"I am a cat. My name is {self.name}. I am {self.age} years old.") class Dog: def __init__(self, name, age): self.name = name self.age = age def info(self): print(f"I am a dog. My name is {self.name}. I am {self.age} years old.") cat1 = Cat("Kitty", 2.5) dog1 = Dog("Fluffy", 4) for animal in (cat1, dog1): animal.make_sound() animal.info() animal.make_sound() A) Meow I am a cat. My name is Kitty. I am 2.5 years old. Meow Bark I am a dog. My name is Fluffy. I am 4 years old. Bark B) I am a cat. My name is Kitty. I am 2.5 years old. I am a dog. My name is Fluffy. I am 4 years old. C) I am a cat. My name is Kitty. I am 2.5 years old. D) I am a dog. My name is Fluffy. I am 4 years old. </pre>	
9.	Выберите правильный вариант ответа. Результат выполнения кода Python <pre> from abc import ABC, abstractmethod class Car(ABC): def mileage(self): pass class Tesla(Car): def mileage(self): print("The mileage is 30kmph") class Suzuki(Car): def mileage(self): print("The mileage is 25kmph ") class Duster(Car): def mileage(self): print("The mileage is 24kmph ") class Renault(Car): </pre>	C

Номер задания	Содержание вопроса	Правильный ответ на задание																																																																																																																																														
	<pre>def mileage(self): print("The mileage is 27kmph ")</pre> s = Suzuki() s.mileage() A) The mileage is 30kmph B) The mileage is 27kmph C) The mileage is 25kmph D) The mileage is 24kmph																																																																																																																																															
10.	Выберите правильный вариант ответа. Результат выполнения кода Python, при работе с файлом *. csv. import pandas as pd csvframe = pd.read_csv('ch05_01.csv') pd.read_csv('ch05_02.csv', names= ['white','red','blue','green','animal']) A) <table><tr><td></td><td>white</td><td>red</td><td>blue</td><td>green</td><td>animal</td></tr><tr><td>0</td><td>1</td><td>5</td><td>2</td><td>3</td><td>cat</td></tr><tr><td>1</td><td>2</td><td>7</td><td>8</td><td>5</td><td>dog</td></tr><tr><td>2</td><td>3</td><td>3</td><td>6</td><td>7</td><td>horse</td></tr><tr><td>3</td><td>2</td><td>2</td><td>8</td><td>3</td><td>duck</td></tr><tr><td>4</td><td>4</td><td>4</td><td>2</td><td>1</td><td>mouse</td></tr></table> B) <table><tr><td></td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td></tr><tr><td>0</td><td>1</td><td>5</td><td>2</td><td>3</td><td>cat</td></tr><tr><td>1</td><td>2</td><td>7</td><td>8</td><td>5</td><td>dog</td></tr><tr><td>2</td><td>3</td><td>3</td><td>6</td><td>7</td><td>horse</td></tr><tr><td>3</td><td>2</td><td>2</td><td>8</td><td>3</td><td>duck</td></tr><tr><td>4</td><td>4</td><td>4</td><td>2</td><td>1</td><td>mouse</td></tr></table> C) <table><tr><td></td><td></td><td>item1</td><td>item2</td><td>item3</td></tr><tr><td>color</td><td>status</td><td></td><td></td><td></td></tr><tr><td>black</td><td>up</td><td>3</td><td>4</td><td>6</td></tr><tr><td></td><td>down</td><td>2</td><td>6</td><td>7</td></tr><tr><td>white</td><td>up</td><td>5</td><td>5</td><td>5</td></tr><tr><td></td><td>down</td><td>3</td><td>3</td><td>2</td></tr><tr><td></td><td>left</td><td>1</td><td>2</td><td>1</td></tr><tr><td>red</td><td>up</td><td>2</td><td>2</td><td>2</td></tr><tr><td></td><td>down</td><td>1</td><td>1</td><td>4</td></tr></table> D) <table><tr><td>white</td><td>up</td><td>5</td><td>5</td><td>5</td></tr><tr><td></td><td>down</td><td>3</td><td>3</td><td>2</td></tr><tr><td></td><td>left</td><td>1</td><td>2</td><td>1</td></tr><tr><td>red</td><td>up</td><td>2</td><td>2</td><td>2</td></tr><tr><td></td><td>down</td><td>1</td><td>1</td><td>4</td></tr></table>		white	red	blue	green	animal	0	1	5	2	3	cat	1	2	7	8	5	dog	2	3	3	6	7	horse	3	2	2	8	3	duck	4	4	4	2	1	mouse		0	1	2	3	4	0	1	5	2	3	cat	1	2	7	8	5	dog	2	3	3	6	7	horse	3	2	2	8	3	duck	4	4	4	2	1	mouse			item1	item2	item3	color	status				black	up	3	4	6		down	2	6	7	white	up	5	5	5		down	3	3	2		left	1	2	1	red	up	2	2	2		down	1	1	4	white	up	5	5	5		down	3	3	2		left	1	2	1	red	up	2	2	2		down	1	1	4	A
	white	red	blue	green	animal																																																																																																																																											
0	1	5	2	3	cat																																																																																																																																											
1	2	7	8	5	dog																																																																																																																																											
2	3	3	6	7	horse																																																																																																																																											
3	2	2	8	3	duck																																																																																																																																											
4	4	4	2	1	mouse																																																																																																																																											
	0	1	2	3	4																																																																																																																																											
0	1	5	2	3	cat																																																																																																																																											
1	2	7	8	5	dog																																																																																																																																											
2	3	3	6	7	horse																																																																																																																																											
3	2	2	8	3	duck																																																																																																																																											
4	4	4	2	1	mouse																																																																																																																																											
		item1	item2	item3																																																																																																																																												
color	status																																																																																																																																															
black	up	3	4	6																																																																																																																																												
	down	2	6	7																																																																																																																																												
white	up	5	5	5																																																																																																																																												
	down	3	3	2																																																																																																																																												
	left	1	2	1																																																																																																																																												
red	up	2	2	2																																																																																																																																												
	down	1	1	4																																																																																																																																												
white	up	5	5	5																																																																																																																																												
	down	3	3	2																																																																																																																																												
	left	1	2	1																																																																																																																																												
red	up	2	2	2																																																																																																																																												
	down	1	1	4																																																																																																																																												
11.	Выберите правильный вариант ответа. Необходимо провести запись данных, например, из объекта Dataframe в файл CSV. Какой функцией библиотеки pandas Python для этого можно воспользоваться? A) read_table()	B																																																																																																																																														

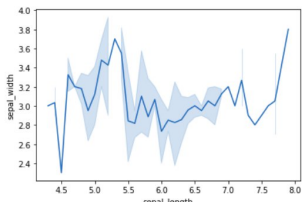
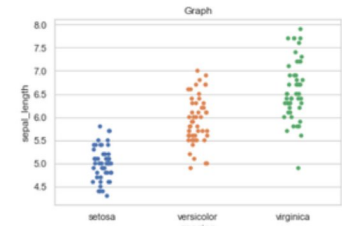
Номер задания	Содержание вопроса	Правильный ответ на задание																																																															
	B) to_csv() C) na_filter() D) df.drop()																																																																
12.	Выберите правильный вариант ответа. Какую библиотеку Python необходимо подключить кроме pandas для визуализации данных (построение простых графиков). A) Seaborn B) Scrubadub C) Datacleaner D) TensorFlow E) Dora F) NumPy	A																																																															
13.	Выберите правильный вариант ответа. Представлен пример диаграммы. По внешнему виду графика определите функцию построения и ее синтаксис.  A) barplot([x, y, hue, data, order, hue_order, ...]) B) countplot([x, y, hue, data, order, ...]) C) boxplot([x, y, hue, data, order, hue_order, ...]) D) violinplot([x, y, hue, data, order, ...])	C																																																															
14.	Выберите правильный вариант ответа. Необходимо провести кластеризацию данных Python pandas. С помощью какого модуля мы можем это сделать? A) sklearn.cluster B) scikit.learn C) sklearn.tree D) X-Small	A																																																															
15.	Выберите правильный вариант ответа. В Python при работе с категориальными признаками преобразовали значения поля Credit_score в числовые. Определите результат работы кода. <table border="1"><thead><tr><th></th><th>Name</th><th>Age</th><th>City</th><th>Region</th><th>Experience</th><th>Salary</th><th>Credit_score</th><th>Outcome</th></tr></thead><tbody><tr><td>0</td><td>Иван</td><td>35</td><td>Москва</td><td>1</td><td>7</td><td>95</td><td>Good</td><td>Yes</td></tr><tr><td>1</td><td>Николай</td><td>43</td><td>Нижний Новгород</td><td>1</td><td>13</td><td>135</td><td>Good</td><td>Yes</td></tr><tr><td>2</td><td>Алексей</td><td>21</td><td>Санкт-Петербург</td><td>1</td><td>2</td><td>73</td><td>Bad</td><td>No</td></tr><tr><td>3</td><td>Александра</td><td>34</td><td>Владивосток</td><td>0</td><td>8</td><td>100</td><td>Medium</td><td>Yes</td></tr><tr><td>4</td><td>Евгений</td><td>24</td><td>Москва</td><td>1</td><td>4</td><td>78</td><td>Medium</td><td>No</td></tr><tr><td>5</td><td>Елена</td><td>27</td><td>Екатеринбург</td><td>0</td><td>12</td><td>110</td><td>Good</td><td>Yes</td></tr></tbody></table> df_cat.Credit_score = df_cat.Credit_score.astype('category').cat.codes		Name	Age	City	Region	Experience	Salary	Credit_score	Outcome	0	Иван	35	Москва	1	7	95	Good	Yes	1	Николай	43	Нижний Новгород	1	13	135	Good	Yes	2	Алексей	21	Санкт-Петербург	1	2	73	Bad	No	3	Александра	34	Владивосток	0	8	100	Medium	Yes	4	Евгений	24	Москва	1	4	78	Medium	No	5	Елена	27	Екатеринбург	0	12	110	Good	Yes	C
	Name	Age	City	Region	Experience	Salary	Credit_score	Outcome																																																									
0	Иван	35	Москва	1	7	95	Good	Yes																																																									
1	Николай	43	Нижний Новгород	1	13	135	Good	Yes																																																									
2	Алексей	21	Санкт-Петербург	1	2	73	Bad	No																																																									
3	Александра	34	Владивосток	0	8	100	Medium	Yes																																																									
4	Евгений	24	Москва	1	4	78	Medium	No																																																									
5	Елена	27	Екатеринбург	0	12	110	Good	Yes																																																									

Номер задания	Содержание вопроса	Правильный ответ на задание																																																																																																																																																																																						
	df_cat A) <table><thead><tr><th></th><th>Name</th><th>Age</th><th>City</th><th>Region</th><th>Experience</th><th>Salary</th><th>Credit_score</th><th>Outcome</th></tr></thead><tbody><tr><td>0</td><td>Иван</td><td>35</td><td>Москва</td><td>1</td><td>7</td><td>95</td><td>Good</td><td>Yes</td></tr><tr><td>1</td><td>Николай</td><td>43</td><td>Нижний Новгород</td><td>1</td><td>13</td><td>135</td><td>Good</td><td>Yes</td></tr><tr><td>2</td><td>Алексей</td><td>21</td><td>Санкт-Петербург</td><td>1</td><td>2</td><td>73</td><td>Bad</td><td>No</td></tr><tr><td>3</td><td>Александра</td><td>34</td><td>Владивосток</td><td>0</td><td>8</td><td>100</td><td>Medium</td><td>Yes</td></tr><tr><td>4</td><td>Евгений</td><td>24</td><td>Москва</td><td>1</td><td>4</td><td>78</td><td>Medium</td><td>No</td></tr><tr><td>5</td><td>Елена</td><td>27</td><td>Екатеринбург</td><td>0</td><td>12</td><td>110</td><td>Good</td><td>Yes</td></tr></tbody></table> B) <table><thead><tr><th></th><th>Name</th><th>Age</th><th>City</th><th>Region</th><th>Experience</th><th>Salary</th><th>Outcome</th></tr></thead><tbody><tr><td>0</td><td>Иван</td><td>35</td><td>Москва</td><td>1</td><td>7</td><td>95</td><td>Yes</td></tr><tr><td>1</td><td>Николай</td><td>43</td><td>Нижний Новгород</td><td>1</td><td>13</td><td>135</td><td>Yes</td></tr><tr><td>2</td><td>Алексей</td><td>21</td><td>Санкт-Петербург</td><td>1</td><td>2</td><td>73</td><td>No</td></tr><tr><td>3</td><td>Александра</td><td>34</td><td>Владивосток</td><td>0</td><td>8</td><td>100</td><td>Yes</td></tr><tr><td>4</td><td>Евгений</td><td>24</td><td>Москва</td><td>1</td><td>4</td><td>78</td><td>No</td></tr><tr><td>5</td><td>Елена</td><td>27</td><td>Екатеринбург</td><td>0</td><td>12</td><td>110</td><td>Yes</td></tr></tbody></table> C) <table><thead><tr><th></th><th>Name</th><th>Age</th><th>City</th><th>Region</th><th>Experience</th><th>Salary</th><th>Credit_score</th><th>Outcome</th></tr></thead><tbody><tr><td>0</td><td>Иван</td><td>35</td><td>Москва</td><td>1</td><td>7</td><td>95</td><td>2</td><td>Yes</td></tr><tr><td>1</td><td>Николай</td><td>43</td><td>Нижний Новгород</td><td>1</td><td>13</td><td>135</td><td>2</td><td>Yes</td></tr><tr><td>2</td><td>Алексей</td><td>21</td><td>Санкт-Петербург</td><td>1</td><td>2</td><td>73</td><td>0</td><td>No</td></tr><tr><td>3</td><td>Александра</td><td>34</td><td>Владивосток</td><td>0</td><td>8</td><td>100</td><td>1</td><td>Yes</td></tr><tr><td>4</td><td>Евгений</td><td>24</td><td>Москва</td><td>1</td><td>4</td><td>78</td><td>1</td><td>No</td></tr><tr><td>5</td><td>Елена</td><td>27</td><td>Екатеринбург</td><td>0</td><td>12</td><td>110</td><td>2</td><td>Yes</td></tr></tbody></table>		Name	Age	City	Region	Experience	Salary	Credit_score	Outcome	0	Иван	35	Москва	1	7	95	Good	Yes	1	Николай	43	Нижний Новгород	1	13	135	Good	Yes	2	Алексей	21	Санкт-Петербург	1	2	73	Bad	No	3	Александра	34	Владивосток	0	8	100	Medium	Yes	4	Евгений	24	Москва	1	4	78	Medium	No	5	Елена	27	Екатеринбург	0	12	110	Good	Yes		Name	Age	City	Region	Experience	Salary	Outcome	0	Иван	35	Москва	1	7	95	Yes	1	Николай	43	Нижний Новгород	1	13	135	Yes	2	Алексей	21	Санкт-Петербург	1	2	73	No	3	Александра	34	Владивосток	0	8	100	Yes	4	Евгений	24	Москва	1	4	78	No	5	Елена	27	Екатеринбург	0	12	110	Yes		Name	Age	City	Region	Experience	Salary	Credit_score	Outcome	0	Иван	35	Москва	1	7	95	2	Yes	1	Николай	43	Нижний Новгород	1	13	135	2	Yes	2	Алексей	21	Санкт-Петербург	1	2	73	0	No	3	Александра	34	Владивосток	0	8	100	1	Yes	4	Евгений	24	Москва	1	4	78	1	No	5	Елена	27	Екатеринбург	0	12	110	2	Yes	
	Name	Age	City	Region	Experience	Salary	Credit_score	Outcome																																																																																																																																																																																
0	Иван	35	Москва	1	7	95	Good	Yes																																																																																																																																																																																
1	Николай	43	Нижний Новгород	1	13	135	Good	Yes																																																																																																																																																																																
2	Алексей	21	Санкт-Петербург	1	2	73	Bad	No																																																																																																																																																																																
3	Александра	34	Владивосток	0	8	100	Medium	Yes																																																																																																																																																																																
4	Евгений	24	Москва	1	4	78	Medium	No																																																																																																																																																																																
5	Елена	27	Екатеринбург	0	12	110	Good	Yes																																																																																																																																																																																
	Name	Age	City	Region	Experience	Salary	Outcome																																																																																																																																																																																	
0	Иван	35	Москва	1	7	95	Yes																																																																																																																																																																																	
1	Николай	43	Нижний Новгород	1	13	135	Yes																																																																																																																																																																																	
2	Алексей	21	Санкт-Петербург	1	2	73	No																																																																																																																																																																																	
3	Александра	34	Владивосток	0	8	100	Yes																																																																																																																																																																																	
4	Евгений	24	Москва	1	4	78	No																																																																																																																																																																																	
5	Елена	27	Екатеринбург	0	12	110	Yes																																																																																																																																																																																	
	Name	Age	City	Region	Experience	Salary	Credit_score	Outcome																																																																																																																																																																																
0	Иван	35	Москва	1	7	95	2	Yes																																																																																																																																																																																
1	Николай	43	Нижний Новгород	1	13	135	2	Yes																																																																																																																																																																																
2	Алексей	21	Санкт-Петербург	1	2	73	0	No																																																																																																																																																																																
3	Александра	34	Владивосток	0	8	100	1	Yes																																																																																																																																																																																
4	Евгений	24	Москва	1	4	78	1	No																																																																																																																																																																																
5	Елена	27	Екатеринбург	0	12	110	2	Yes																																																																																																																																																																																
16.	Дан код Python. Перечислите свойства класса Point. <pre>class Point: MAX_COORD = 100 MIN_COORD = 0 def __init__(self, x, y): self.x = x self.y = y def set_coord(self, x, y): self.x = x self.y = y</pre>	В классе Point определены атрибуты: два свойства MAX_COORD и MIN_COORD.																																																																																																																																																																																						
17.	Дан код Python. Перечислите методы класса Point. <pre>class Point: MAX_COORD = 100 MIN_COORD = 0 def __init__(self, x, y): self.x = x self.y = y def set_coord(self, x, y): self.x = x self.y = y</pre>	В классе Point определены два метода __init__ и set_coord.																																																																																																																																																																																						
18.	Приведен пример кода Python. Дайте краткое описание используемому методу	object.__getattr__() вызывается при обращении, к несуществующему																																																																																																																																																																																						

Номер задания	Содержание вопроса	Правильный ответ на задание
	object.__getattr__(). object.__getattr__(self, name)	атрибуту
19.	Приведен пример кода Python. Дайте краткое описание используемому методу object.__getattribute__(). object.__getattribute__(self, name)	object.__getattribute__() вызывается всегда, при обращении к любому атрибуту
20.	Приведен пример кода Python. Дайте краткое описание используемому методу object.__setattr__(). object.__setattr__(self, name, value)	object.__setattr__() вызывается при попытке присвоения значения любому атрибуту
21.	Приведен пример кода Python. Дайте краткое описание используемому методу object.__delattr__(). object.__delattr__(self, name)	object.__delattr__() вызывается при попытке удаления любого атрибута
22.	Приведен пример кода Python. Дайте краткое описание используемому методу object.__dir__(). object.__dir__(self)	object.__dir__() вызывается функцией dir()
23.	Особенности использования переменных с одним подчеркиванием (например _name) в Python.	Имя с одним подчеркиванием (_name) - по общему соглашению используется для указания на то, что объект является внутренним и предназначен только для использования внутри класса, модуля или функции.
24.	Особенности использования переменных с двумя подчеркиваниями (например __name) в Python.	Имя с двумя подчеркиваниями __name служит для создания «приватных» атрибутов класса. Это означает, что доступ к таким атрибутам извне класса будет затруднен.
25.	Пропишите синтаксис создания объекта в Python на основе существующего класса.	имя_объекта = имя_класса()
26.	Пропишите синтаксис создания класса в Python.	class <ИмяКласса>: <описание класса>
27.	Пропишите синтаксис свойств объектов (атрибутов) класса в Python.	<имя_объекта>.<имя_атрибута> = <значение>
28.	Пропишите синтаксис методов класса в Python.	def <имя_метода>(self, <аргументы>): <тело метода>
29.	Дайте описание методу __del__() в Python.	Деструктор __del__() в Python – это специальный метод, который вызывается при уничтожении объекта.
30.	Какие особенности сокрытия атрибутов класса в Python.	В языке программирования Python скрыть атрибуты класса можно сделав их приватными или закрытыми и ограничив доступ к ним через специальные методы, которые еще называются свойствами.

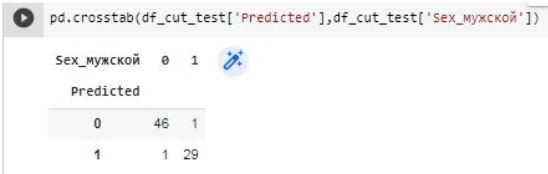
Номер задания	Содержание вопроса	Правильный ответ на задание
31.	Представлен код на Python. Какой возраст (age) будет у объекта tom <pre> class Person: def __init__(self, name): self.__name = name # устанавливаем имя self.__age = 1 # устанавливаем возраст def set_age(self, age): if age in range(1, 100): self.__age = age else: print("Недопустимый возраст") def get_age(self): return self.__age def get_name(self): return self.__name def display_info(self): print("Имя:", self.__name, "\tВозраст:", self.__age) tom = Person("Tom") tom.__age = 43 tom.display_info()</pre>	т.к. у свойства __age две подчеркивания оно является приватным, следовательно может быть изменено только через родительский класс (Person), но не как через дочерний (tom).
32.	Представлен код на Python. Опишите полученные пользователем данные. <pre> class Person: def __init__(self, name): self.__name = name # устанавливаем имя self.__age = 1 # устанавливаем возраст def set_age(self, age): if age in range(1, 100): self.__age = age else: print("Недопустимый возраст") def get_age(self): return self.__age def get_name(self): return self.__name def display_info(self): print("Имя:", self.__name, "\tВозраст:",</pre>	Пользователь увидит информацию «Недопустимый возраст», т.к. в родительском классе (Person) в атрибуте set_age указаны ограничения присвоения значения.

Номер задания	Содержание вопроса	Правильный ответ на задание
	<pre>self.__age) tom = Person("Tom") tom.set_age(-3486)</pre>	
33.	Какой метод в Python называют абстрактным?	Абстрактным называется объявленный, но не реализованный метод. Абстрактные классы не могут быть инстанцированы, от них нужно унаследовать, реализовать все их абстрактные методы и только тогда можно создать экземпляр такого класса.
34.	Представлен код Python. Результат выполнения кода. <pre>from abc import ABC class Polygon(ABC): # abstract method def sides(self): pass class Triangle(Polygon): def sides(self): print("Triangle has 3 sides") class Pentagon(Polygon): def sides(self): print("Pentagon has 5 sides") class Hexagon(Polygon): def sides(self): print("Hexagon has 6 sides") class square(Polygon): def sides(self): print("I have 4 sides") t = Triangle() t.sides()</pre>	Triangle has 3 sides
35.	Опишите понятие датафрейм (DataFrame) в Pandas Python.	Python Pandas DataFrame — это широко используемая структура данных, которая работает с двумерным массивом с

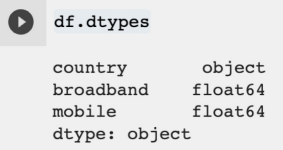
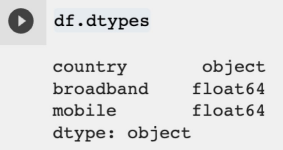
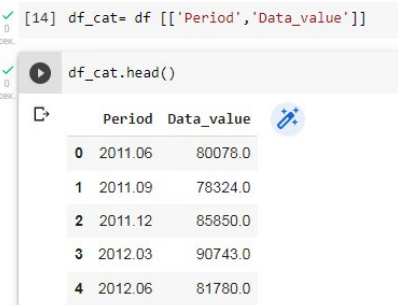
Номер задания	Содержание вопроса	Правильный ответ на задание
		маркированными осями (строками и столбцами). DataFrame определяется как стандартный способ хранения данных с двумя разными индексами, т. е. индексом строки и индексом столбца.
36.	Представлен код Python. Дайте краткое описание коду. <pre>import seaborn as sns df = sns.load_dataset("penguins") sns.pairplot(df, hue="species")</pre>	Первой строкой подключили библиотеку seaborn для создания статистических графиков на Python. Второй строкой – формирование датасета. Третьей строкой формируем набор графиков зависимостей различных характеристик от значений "species".
37.	Представлен код Python с результатом работы кода. Определите вид графика и метод создания графика. <pre># importing packages import seaborn as sns # loading dataset data = sns.load_dataset("iris") # draw lineplot sns.lineplot(x="sepal_length", y="sepal_width", data=data)</pre> 	В приведенном выше примере простой линейный график создается с использованием метода lineplot().
38.	Представлен код Python с результатом работы кода. Почему на графике разделение данных по цвету? <pre>import matplotlib.pyplot as plt import seaborn as sns sns.set(style="whitegrid") iris = sns.load_dataset('iris') ax = sns.stripplot(x='species', y='sepal_length', data=iris) plt.title('Graph') plt.show()</pre> 	В столбце таблицы три вида данных (видов цветов) которые представлены на графике тремя цветами точек.
39.	По программному коду Python определите	Представлен код

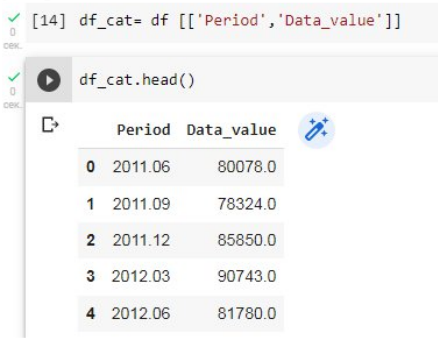
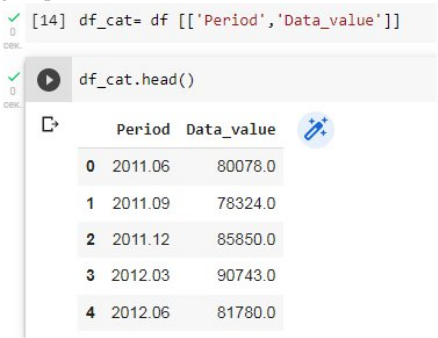
Номер задания	Содержание вопроса	Правильный ответ на задание																																								
	вид иерархической кластеризации <pre># Hierarchical clustering using Complete Linkage hc1 <- hclust(d, method = "complete") # Plot the obtained dendrogram plot(hc1, cex = 0.6, hang = -1)</pre>	(Параметр method hclust) реализующий кластерный анализ типа агломеративной иерархической кластеризации																																								
40.	Представлена часть кода Python. Кратко опишите действия кода. <pre>from sklearn.linear_model import LogisticRegression from sklearn.discriminant_analysis import LinearDiscriminantAnalysis from sklearn.neighbors import KNeighborsClassifier from sklearn.naive_bayes import GaussianNB from sklearn.tree import DecisionTreeClassifier from sklearn.svm import SVC</pre>	Импорт в Python классификаторов (для логистической регрессии и т.д.)																																								
41.	Представлена код Python разделение данных на обучающие и тестовые наборы. Кратко опишите действия кода. <pre>from sklearn.model_selection import train_test_split SEED = 42 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.25, random_state=SEED)</pre>	Этот фрагмент кода выбирает 75 % данных для обучения и 25 % данных для тестирования. Например , изменив значение test_size на 0,3, вы сможете тренироваться с 70% данных и тестировать с 30%.																																								
42.	Проведен анализ классификации с использованием алгоритма К-ближайших соседей (K-Nearest Neighbours - KNN) реализованном с помощью библиотеки Python Scikit-Learn. С помощью методов confusion_matrix() и classification_report() модуля sklearn.metrics провели расчет и отображение показателей классификации. Укажите количество записей и точность тестового набора. <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>1</td><td>0.75</td><td>0.78</td><td>0.76</td><td>1292</td></tr><tr><td>2</td><td>0.49</td><td>0.56</td><td>0.53</td><td>1283</td></tr><tr><td>3</td><td>0.51</td><td>0.51</td><td>0.51</td><td>1292</td></tr><tr><td>4</td><td>0.76</td><td>0.62</td><td>0.69</td><td>1293</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.62</td><td>5160</td></tr><tr><td>macro avg</td><td>0.63</td><td>0.62</td><td>0.62</td><td>5160</td></tr><tr><td>weighted avg</td><td>0.63</td><td>0.62</td><td>0.62</td><td>5160</td></tr></tbody></table>		precision	recall	f1-score	support	1	0.75	0.78	0.76	1292	2	0.49	0.56	0.53	1283	3	0.51	0.51	0.51	1292	4	0.76	0.62	0.69	1293	accuracy			0.62	5160	macro avg	0.63	0.62	0.62	5160	weighted avg	0.63	0.62	0.62	5160	Результаты показывают, что KNN смог классифицировать все 5160 записей в тестовом наборе с точностью 62%, что выше среднего.
	precision	recall	f1-score	support																																						
1	0.75	0.78	0.76	1292																																						
2	0.49	0.56	0.53	1283																																						
3	0.51	0.51	0.51	1292																																						
4	0.76	0.62	0.69	1293																																						
accuracy			0.62	5160																																						
macro avg	0.63	0.62	0.62	5160																																						
weighted avg	0.63	0.62	0.62	5160																																						
43.	Проведен анализ классификации с использованием алгоритма К-ближайших соседей (K-Nearest Neighbours - KNN) реализованном с помощью библиотеки Python Scikit-Learn. С помощью методов confusion_matrix() и classification_report() модуля sklearn.metrics провели расчет и	Указаны результаты метрик для каждого из 4 классов. Исходя из этого, мы можем заметить, что у него class 2 была самая низкая точность, самая низкая recall и самая низкая f1-score. Class 3 сразу позади class 2 по наименьшим баллам, а затем идут																																								

Номер задания	Содержание вопроса	Правильный ответ на задание																																								
	отображение показателей классификации. Укажите количество классов, проанализируйте точность. <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>1</td><td>0.75</td><td>0.78</td><td>0.76</td><td>1292</td></tr><tr><td>2</td><td>0.49</td><td>0.56</td><td>0.53</td><td>1283</td></tr><tr><td>3</td><td>0.51</td><td>0.51</td><td>0.51</td><td>1292</td></tr><tr><td>4</td><td>0.76</td><td>0.62</td><td>0.69</td><td>1293</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.62</td><td>5160</td></tr><tr><td>macro avg</td><td>0.63</td><td>0.62</td><td>0.62</td><td>5160</td></tr><tr><td>weighted avg</td><td>0.63</td><td>0.62</td><td>0.62</td><td>5160</td></tr></tbody></table>		precision	recall	f1-score	support	1	0.75	0.78	0.76	1292	2	0.49	0.56	0.53	1283	3	0.51	0.51	0.51	1292	4	0.76	0.62	0.69	1293	accuracy			0.62	5160	macro avg	0.63	0.62	0.62	5160	weighted avg	0.63	0.62	0.62	5160	class 1 лучшие баллы, за которыми следует class 4.
	precision	recall	f1-score	support																																						
1	0.75	0.78	0.76	1292																																						
2	0.49	0.56	0.53	1283																																						
3	0.51	0.51	0.51	1292																																						
4	0.76	0.62	0.69	1293																																						
accuracy			0.62	5160																																						
macro avg	0.63	0.62	0.62	5160																																						
weighted avg	0.63	0.62	0.62	5160																																						
44.	Проведен анализ классификации с использованием алгоритма К-ближайших соседей (K-Nearest Neighbours - KNN) реализованном с помощью библиотеки Python Scikit-Learn. С помощью методов confusion_matrix() и classification_report() модуля sklearn.metrics провели расчет и отображение показателей классификации. Проанализируйте матрицу путаницы. <table><thead><tr><th></th><th>class 1</th><th>class 2</th><th>class 3</th><th>class 4</th></tr></thead><tbody><tr><th>class 1</th><td>1012</td><td>238</td><td>31</td><td>11</td></tr><tr><th>class 2</th><td>256</td><td>723</td><td>260</td><td>44</td></tr><tr><th>class 3</th><td>69</td><td>374</td><td>656</td><td>193</td></tr><tr><th>class 4</th><td>20</td><td>130</td><td>339</td><td>804</td></tr></tbody></table>		class 1	class 2	class 3	class 4	class 1	1012	238	31	11	class 2	256	723	260	44	class 3	69	374	656	193	class 4	20	130	339	804	Глядя на матрицу путаницы Python, видим, что: - class 1 чаще всего принимался за class 2 в 238 случаях - class 2 для class 1256 записей и для class 3260 случаев - class 3 чаще всего ошибался class 2, 374 записей, и class 4, в 193 случаях - class 4 было ошибочно классифицировано как class 3 для 339 записей, так и class 2 для 130 случаев. Также стоит обратить внимание, что диагональ отображает истинные положительные значения, при взгляде на нее это ясно видно class 2 и class 3 имеет наименее правильно предсказанные значения.															
	class 1	class 2	class 3	class 4																																						
class 1	1012	238	31	11																																						
class 2	256	723	260	44																																						
class 3	69	374	656	193																																						
class 4	20	130	339	804																																						
45.	Дайте краткое описание понятию Смешивание в Python.	Смешивание — это метод ансамблевого машинного обучения, который использует модель машинного обучения, чтобы узнать, как наилучшим образом объединить прогнозы из нескольких моделей — участников ансамбля.																																								
46.	В для каких целей используется RandomForestClassifier	Алгоритм RandomForestClassifier использует в качестве ансамбля Случайный лес (несколько деревьев) RandomForest.																																								
47.	Дайте краткое описание программному коду Python model=tree.DecisionTreeClassifier(max_depth=4) model.fit(df_cut[['Growth','Weight','Hair length','Children number','Coin','Animal','Army']].values.reshape(-1,7), y=df_cut['Sex'].values)	Строим дерево решений.																																								
48.	Перечислите библиотеки Python необходимые для построения деревьев решений.	Библиотеки необходимые для построения деревьев решений: Библиотека pandas предназначена для анализа уже структурированных данных, то есть размещённых не хаотично, а в таблицах. Seaborn — это библиотека для																																								

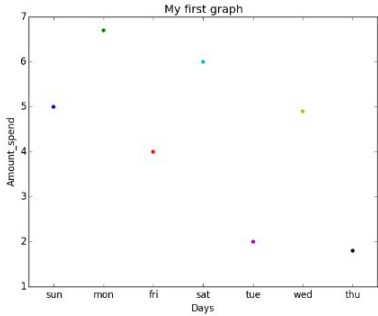
Номер задания	Содержание вопроса	Правильный ответ на задание
		создания статистических графиков на Python. Sklearn - это библиотека машинного обучения для языка программирования Python, которая предоставляет множество возможностей, таких как многоступенчатый анализ, регрессия и алгоритмы кластеризации.
49.	По предложенному коду Python определите действие метода get_dummies. df_cut=pd.get_dummies(df_cut) df_cut.head()	Метод get_dummies, расщепляет признаки по своим категориям
50.	По результату и программному коду определите действие кода. 	Получены метрики качества задачи (классификации).
1.	Представлена структура наследования в Python. Определите тип наследования class Base1: pass class Base2: pass class MultiDerived(Base1, Base2): pass A) множественное наследование B) обычное наследование C) иерархическое наследование D) перекрестное наследование	A
2.	Выберите правильный вариант ответа. Результат выполнения кода Python from abc import ABC, abstractmethod class Car(ABC): def mileage(self): pass class Tesla(Car): def mileage(self): print("The mileage is 30kmph") class Suzuki(Car): def mileage(self): print("The mileage is 25kmph ") class Duster(Car): def mileage(self): print("The mileage is 24kmph ") class Renault(Car): def mileage(self):	B

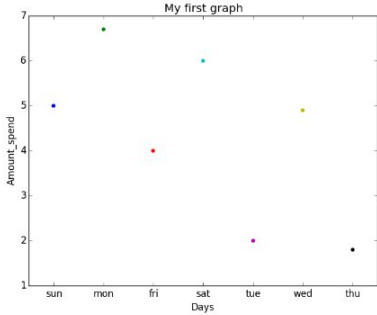
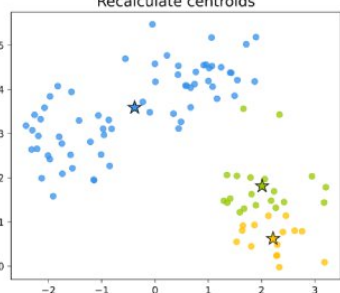
Номер задания	Содержание вопроса	Правильный ответ на задание
	<pre>print("The mileage is 27kmph ")</pre> <pre>r = Renault() r.mileage()</pre> A) The mileage is 30kmph B) The mileage is 27kmph C) The mileage is 25kmph D) The mileage is 24kmph	
3.	Выберите правильный вариант ответа. Результат выполнения кода Python <pre>from abc import ABC, abstractmethod class Car(ABC): def mileage(self): pass</pre> <pre>class Tesla(Car): def mileage(self): print("The mileage is 30kmph") class Suzuki(Car): def mileage(self): print("The mileage is 25kmph ") class Duster(Car): def mileage(self): print("The mileage is 24kmph ")</pre> <pre>class Renault(Car): def mileage(self): print("The mileage is 27kmph ")</pre> <pre>d = Duster() d.mileage()</pre> A) The mileage is 30kmph B) The mileage is 27kmph C) The mileage is 25kmph D) The mileage is 24kmph	D
4.	Выберите правильный вариант ответа. При работе Python библиотеки Pandas с данными файла Internet Speed 2022.csv прописана команда приведенная ниже. Определите как назван датасет. <pre>df = pd.read_csv('/content/Internet Speed 2022.csv')</pre> A) read_csv B) df C) pd D) md	B
5.	Выберите правильный вариант ответа. При работе Python библиотеки Pandas с данными файла *.csv получен результат. Дайте описание столбце country.	A

Номер задания	Содержание вопроса	Правильный ответ на задание																		
	 df.dtypes <pre>country object broadband float64 mobile float64 dtype: object</pre> А) данный столбец представляет собой тип object. Это тип данных для строковых и смешанных значений В) данный столбец имеют тип данных float, то есть относятся к числам с плавающей точкой С) данный столбец имеют тип данных dtype, то есть относятся к числам с плавающей точкой D) данный столбец имеют тип данных float64, то есть относятся к числам с плавающей точкой																			
6.	Выберите правильный вариант ответа. При работе Python библиотеки Pandas с данными файла *.csv получен результат. Дайте описание столбце mobile.  df.dtypes <pre>country object broadband float64 mobile float64 dtype: object</pre> А) данный столбец представляет собой тип object. Это тип данных для строковых и смешанных значений В) данный столбец имеют тип данных float, то есть относятся к числам с плавающей точкой С) данный столбец имеют тип данных dtype, то есть относятся к числам с плавающей точкой D) данный столбец имеют тип данных float64, то есть относятся к числам с плавающей точкой	В																		
7.	Выберите правильный вариант ответа. При работе Python библиотеки Pandas с данными файла выделено несколько столбцов в отдельной таблице, получен результат.  [14] df_cat= df [['Period', 'Data_value']] df_cat.head() <table border="1"> <thead> <tr> <th></th> <th>Period</th> <th>Data_value</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>2011.06</td> <td>80078.0</td> </tr> <tr> <td>1</td> <td>2011.09</td> <td>78324.0</td> </tr> <tr> <td>2</td> <td>2011.12</td> <td>85850.0</td> </tr> <tr> <td>3</td> <td>2012.03</td> <td>90743.0</td> </tr> <tr> <td>4</td> <td>2012.06</td> <td>81780.0</td> </tr> </tbody> </table> К данному набору применена команда		Period	Data_value	0	2011.06	80078.0	1	2011.09	78324.0	2	2011.12	85850.0	3	2012.03	90743.0	4	2012.06	81780.0	А
	Period	Data_value																		
0	2011.06	80078.0																		
1	2011.09	78324.0																		
2	2011.12	85850.0																		
3	2012.03	90743.0																		
4	2012.06	81780.0																		

Номер задания	Содержание вопроса	Правильный ответ на задание
	<pre>df_filter = df[df['STATUS'] == 'R'] df_filter.head()</pre> А) Создан срез данных с определенным статусом R Б) Создан срез данных с определенным статусом R и Data_value > 80000 В) Подсчитано количество записей Г) Подсчитано количество записей среза R	
8.	Выберите правильный вариант ответа. При работе Python библиотеки Pandas с данными файла выделено несколько столбцов в отдельной таблице, получен результат.  К данному набору применена команда <code>df[(df['STATUS'] == 'R') & (df['Data_value'] >= 80000)].head()</code> А) Создан срез данных с определенным статусом R Б) Создан срез данных с определенным статусом R и Data_value > 80000 В) Подсчитано количество записей Г) Подсчитано количество записей среза R	В
9.	Выберите правильный вариант ответа. При работе Python библиотеки Pandas с данными файла выделено несколько столбцов в отдельной таблице, получен результат.  К данному набору применена команда <code>df[(df['STATUS'] == 'R') + (df['Data_value'] >= 80000)].shape</code> А) Создан срез данных с определенным статусом R	С

Номер задания	Содержание вопроса	Правильный ответ на задание
	В) Создан срез данных с определенным статусом R и Data_value>80000 С) Подсчитано количество записей среза данных с определенным статусом R и Data_value>80000 Д) Подсчитано количество записей среза R	
10.	Выберите правильный вариант ответа. При работе Python библиотеки Pandas с данными файла выделено несколько столбцов в отдельную таблицу. К набору данных применена команда. Результат? <pre>maximums = data.max() print(maximums)</pre> А) Полученный результат — максимальные значения в каждом столбце В) Полученный результат — максимальное значение в столбце «sugar» С) Полученный результат — строка таблицы с максимальным значением Д) Полученный результат — абзац таблицы с максимальным значением	А
11.	Выберите правильный вариант ответа. Представлен код Python библиотеки Pandas с данными файла и полученный результат. Какая функция используется для определения типа графика и для его отображения на экране с помощью. <pre># Python program to illustrate # Plotting categorical scatter # plots with Seaborn # importing the required module import matplotlib.pyplot as plt import seaborn as sns # x axis values x = ['sun', 'mon', 'fri', 'sat', 'tue', 'wed', 'thu'] # y axis values y = [5, 6.7, 4, 6, 2, 4.9, 1.8] # plotting strip plot with seaborn ax = sns.stripplot(x, y); # giving labels to x-axis and y-axis ax.set(xlabel='Days', ylabel='Amount_spend') # giving title to the plot plt.title('My first graph');</pre>	В

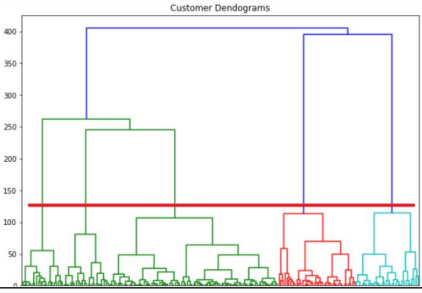
Номер задания	Содержание вопроса	Правильный ответ на задание
	# function to show plot plt.show()  A) .set() B) .striplot() C) .title() D) .show()	
12.	Выберите правильный вариант ответа. Представлен код Python библиотеки Pandas с данными файла и полученный результат. Какая функция используется для установки меток по осям x и y. # Python program to illustrate # Plotting categorical scatter # plots with Seaborn <pre># importing the required module import matplotlib.pyplot as plt import seaborn as sns # x axis values x=['sun', 'mon', 'fri', 'sat', 'tue', 'wed', 'thu'] # y axis values y=[5, 6.7, 4, 6, 2, 4.9, 1.8] # plotting strip plot with seaborn ax = sns.stripplot(x, y); # giving labels to x-axis and y-axis ax.set(xlabel='Days', ylabel='Amount_spend') # giving title to the plot plt.title('My first graph'); # function to show plot plt.show()</pre>	A

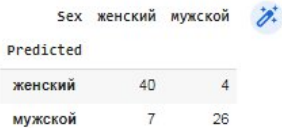
Номер задания	Содержание вопроса	Правильный ответ на задание
	 A) .set() B) .striplot() C) .title() D) .show()	
13.	Выберите правильный вариант ответа. Для качественного анализа данных нам необходимо удалить строки с незаполненными данными A) <code>df=df.dropna()</code> B) <code>df=df.fillna(df.mean())</code> C) <code>df=df.fillna(0)</code> D) <code>df=df.mean(0)</code>	A
14.	Выберите правильный вариант ответа. Для качественного анализа данных нам необходимо заполнить пропуски значением (Например средним значением по столбцу) A) <code>df=df.dropna()</code> B) <code>df=df.fillna(df.median())</code> C) <code>df=df.fillna(0)</code> D) <code>df=df.mean(0)</code>	B
15.	Выберите правильный вариант ответа. Представлен результат кластеризации Python. Определите метод кластеризации.  A) DBSCAN кластеризация B) кластеризация методом К средних C) иерархическая кластеризация D) внутригрупповая кластеризация	C
16.	При создании в Python вычисляемых атрибутов класса используют специальные методы. Дайте краткое описание методу <code>object. getattr ()</code>.	<code>object.__getattr__()</code> вызывается при обращении, к несуществующему атрибуту
17.	При создании в Python вычисляемых	<code>object. getattribute ()</code> вызывается

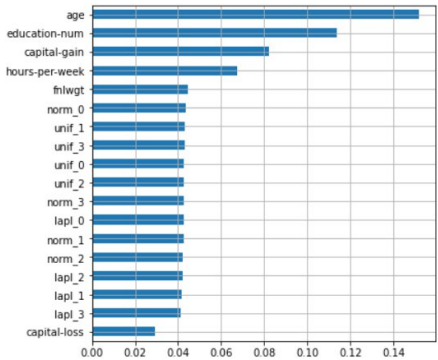
Номер задания	Содержание вопроса	Правильный ответ на задание
	атрибутов класса используют специальные методы. Дайте краткое описание методу <code>object.getattribute()</code> .	всегда, при обращении к любому атрибуту
18.	При создании в Python вычисляемых атрибутов класса используют специальные методы. Дайте краткое описание методу <code>object.setattr()</code> .	<code>object.__setattr__()</code> вызывается при попытке присвоения значения любому атрибуту
19.	При создании в Python вычисляемых атрибутов класса используют специальные методы. Дайте краткое описание методу <code>object.delattr()</code> .	<code>object.__delattr__()</code> вызывается при попытке удаления любого атрибута
20.	При создании в Python вычисляемых атрибутов класса используют специальные методы. Дайте краткое описание методу <code>object.dir()</code> .	<code>object.__dir__()</code> вызывается функцией <code>dir()</code>
21.	Пропишите синтаксис деструктора в Python.	<code>def __del__(self):</code> # тело деструктора
22.	Дайте описание понятию «порядок разрешения методов» (MRO – method resolution order).	Под MRO дочернего класса понимается его линеаризация - список предков класса, включая сам класс, отсортированный в порядке "удалённости".
23.	Представлен код Python. Опишите действия метода <code>super()</code> . <code>class C(B, A):</code> <code>def __init__(self):</code> <code>super().__init__()</code>	Функция <code>super()</code> занимается делегированием вызовов методов классу C в дереве предков экземпляра.
24.	Представлен код Python. Опишите взаимосвязь метода <code>__init__</code> и параметра <code>self</code> . <code>class C(B, A):</code> <code>def __init__(self):</code> <code>super().__init__()</code>	Атрибуты объекта перечисляют внутри <code>__init__</code> метода класса – он вызывается каждый раз при создании экземпляра класса. Параметр <code>self</code> создает ссылку на экземпляр класса и позволяет получить доступ к атрибутам и методам объекта.
25.	Какими методами необходимо воспользоваться в Python для получения доступа к приватным атрибутам вне класса?	Для решения проблемы получения доступа к приватным атрибутам вне класса необходимо воспользоваться: - получающими методами (геттерами), которые позволяют нам получать значения приватных атрибутов извне класса; - устанавливающими методами (сеттерами), которые позволяют нам устанавливать значения частных атрибутов извне класса.
26.	Перечислите преимущества сокрытия данных в Python.	Преимущества сокрытия данных включают: - Защиту данных от неправильного использования и изменения. - Упрощение поддержки и развития

Номер задания	Содержание вопроса	Правильный ответ на задание
		кода, так как изменения внутренней реализации класса не повлияют на внешний код, который использует публичный интерфейс. - Улучшение безопасности, так как только разрешенные методы имеют доступ к приватным данным.
27.	Представлен код Python. Результат выполнения кода. <pre>from abc import ABC class Polygon(ABC): # abstract method def sides(self): pass class Triangle(Polygon): def sides(self): print("Triangle has 3 sides") class Pentagon(Polygon): def sides(self): print("Pentagon has 5 sides") class Hexagon(Polygon): def sides(self): print("Hexagon has 6 sides") class square(Polygon): def sides(self): print("I have 4 sides") s = square() s.sides()</pre>	Square has 4 sides
28.	При работе Python библиотеки Pandas с данными файла *.csv использовался метод dropna(). Опишите данный метод.	Метод dropna() удаляет все строки с пропусками значений.

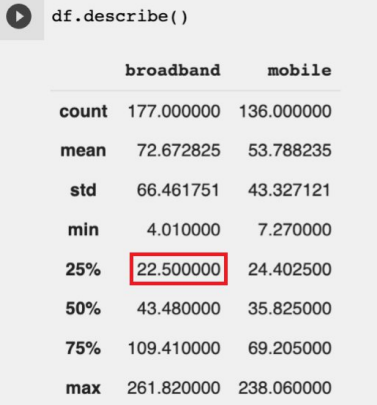
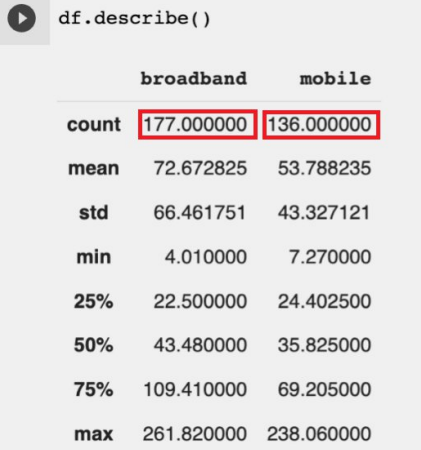
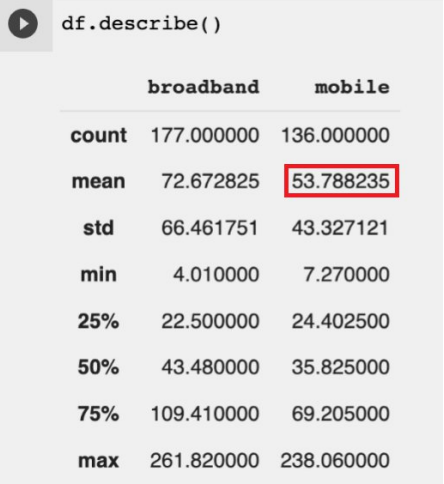
Номер задания	Содержание вопроса	Правильный ответ на задание																																																
	<pre>df.dropna()</pre> <table><thead><tr><th></th><th>country</th><th>broadband</th><th>mobile</th></tr></thead><tbody><tr><td>1</td><td>Singapore</td><td>255.83</td><td>105.01</td></tr><tr><td>2</td><td>Hong Kong</td><td>254.70</td><td>83.58</td></tr><tr><td>3</td><td>Romania</td><td>232.17</td><td>59.67</td></tr><tr><td>4</td><td>Switzerland</td><td>229.96</td><td>135.70</td></tr><tr><td>5</td><td>Denmark</td><td>227.91</td><td>112.03</td></tr><tr><td>...</td><td>...</td><td>...</td><td>...</td></tr><tr><td>168</td><td>Algeria</td><td>10.43</td><td>18.10</td></tr><tr><td>169</td><td>Mozambique</td><td>9.27</td><td>25.02</td></tr><tr><td>171</td><td>Afghanistan</td><td>8.26</td><td>7.27</td></tr><tr><td>173</td><td>Sudan</td><td>7.92</td><td>14.51</td></tr><tr><td>176</td><td>Cuba</td><td>4.01</td><td>25.02</td></tr></tbody></table> 136 rows x 3 columns		country	broadband	mobile	1	Singapore	255.83	105.01	2	Hong Kong	254.70	83.58	3	Romania	232.17	59.67	4	Switzerland	229.96	135.70	5	Denmark	227.91	112.03	...	...	...	...	168	Algeria	10.43	18.10	169	Mozambique	9.27	25.02	171	Afghanistan	8.26	7.27	173	Sudan	7.92	14.51	176	Cuba	4.01	25.02	
	country	broadband	mobile																																															
1	Singapore	255.83	105.01																																															
2	Hong Kong	254.70	83.58																																															
3	Romania	232.17	59.67																																															
4	Switzerland	229.96	135.70																																															
5	Denmark	227.91	112.03																																															
...	...	...	...																																															
168	Algeria	10.43	18.10																																															
169	Mozambique	9.27	25.02																																															
171	Afghanistan	8.26	7.27																																															
173	Sudan	7.92	14.51																																															
176	Cuba	4.01	25.02																																															
29.	При работе Python библиотеки Pandas с данными файла использовалась функция <code>sort_values()</code> . Дайте описание работе данной функции.	Порядок сортировки по умолчанию функции <code>sort_values()</code> – возрастающий. Чтобы отсортировать в порядке убывания, используют аргумент <code>ascending = False</code> методу <code>sort_values()</code> .																																																
30.	Представлен результат построения графика зависимости в Python. Дайте описание.	График зависимости пола (женский – синие точки и мужского – оранжевые точки) от размера обуви и роста.																																																
31.	Опишите кратко алгоритм кластеризации методом k-средних, используемый в Python.	Исходной задачей будет распределение произвольного количества n-мерных точек по k кластерам. 1. Случайным образом создаются k точек, в дальнейшем будем называть их центрами кластеров; 2. Для каждой точки ставится в соответствии ближайший к ней центр кластера; 3. Вычисляются средние арифметические точек, принадлежащих к определённому кластеру. Именно эти значения становятся новыми центрами кластеров; 4. Шаги 2 и 3 повторяются до тех пор, пока пересчёт центров кластеров																																																

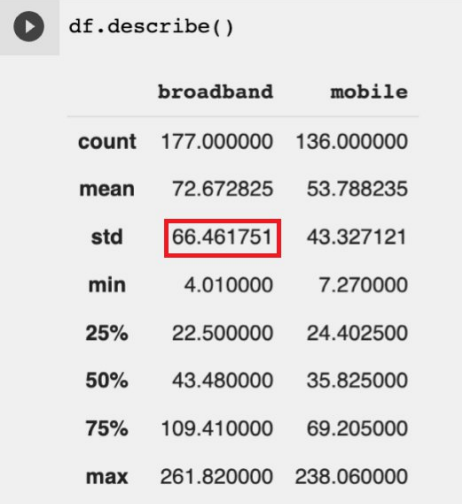
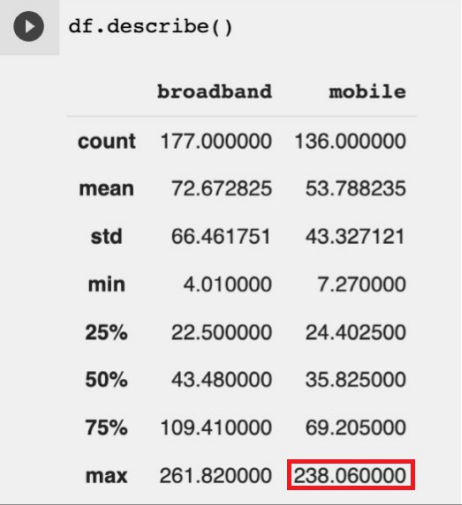
Номер задания	Содержание вопроса	Правильный ответ на задание
		будет приносить плоды. Как только высчитанные центры кластеров совпадут с предыдущими, алгоритм будет окончен.
32.	Представлен результат построения дендрограммы с использованием библиотеки <code>scipy</code> Python. Предложите количество кластеров. 	Если мы нарисуем горизонтальную линию, которая проходит через самое длинное расстояние без горизонтальной линии, мы получим 5 кластеров, как показано на рисунке.
33.	Дайте определение понятию агломеративная иерархическая кластеризация.	Агломеративная иерархическая кластеризация: это подход "снизу вверх", при котором каждая точка данных начинается в своем собственном кластере, и по мере продвижения вверх по иерархии похожие пары кластеров объединяются. Иерархическая кластеризация с разделением: это подход "сверху вниз", при котором все точки данных начинаются в одном кластере, и по мере продвижения вниз по иерархии кластеры рекурсивно разделяются.
34.	Дайте определение понятию иерархическая кластеризация с разделением.	Иерархическая кластеризация с разделением: это подход "сверху вниз", при котором все точки данных начинаются в одном кластере, и по мере продвижения вниз по иерархии кластеры рекурсивно разделяются.
35.	Методы количественной оценки качества прогнозов классификации Python.	Есть различные для оценки качества прогнозов модели: 1) Метод оценки оценщика : <code>score</code> метод, обеспечивающий критерий оценки по умолчанию для проблемы, для решения которой они предназначены. 2) Параметр оценки: инструменты оценки модели с использованием перекрестной проверки (например, <code>model_selection.cross_val_score</code> и <code>model_selection.GridSearchCV</code>) 3) Метрические функции : В sklearn.metrics модуле реализованы

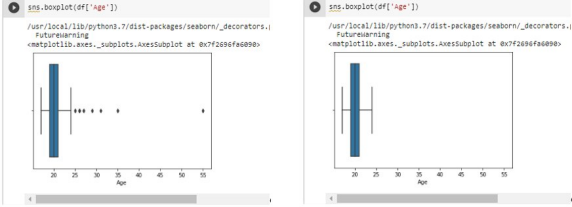
Номер задания	Содержание вопроса	Правильный ответ на задание
		функции оценки ошибки прогноза для конкретных целей.
36.	Построена матрица сопряженности или матрица предсказанных значений определения пола по различным показателям в Python. Проанализируйте полученные данные. <pre>[40] pd.crosstab(df_test_cut['Predicted'],df_test_cut['Sex'])</pre> 	Анализ матрицы сопряжения: 40 – реальный женский пол предсказанный как женский 26 – реальный мужской пол предсказанный как мужской 7 - реальный женский пол предсказанный как мужской 4 - реальный мужской пол предсказанный как женский
37.	В Python провели вероятностную характеристику данных принадлежности к двум классам. Опишите первую строку полученных данных. <pre>result=model.predict_proba(df_test_cut[['Growth','Weight','Hair length','Children number']]).values.reshape(-1,4)</pre> <pre>print(result)</pre> <pre>[[0.10106635 0.89893365] [0.9514851 0.0485149] [0.89584814 0.10415186] [0.93596323 0.06403677] [0.07445031 0.92554969] [0.95999634 0.04000366]]</pre>	Просмотр таблицы говорит что первый экземпляр на 10,10% принадлежит одному классу и на 89,89% другому классу
38.	Приведите не менее трех методов отбора для машинного обучения на Python отбора признаков, имеющих наиболее тесные взаимосвязи с целевой переменной.	Три примера можно выбрать из данного списка методов отбора для машинного обучения на Python отбора признаков: - Одномерный отбор признаков - Рекурсивное исключение признаков - Метод главных компонент - Отбор на основе важности признаков - Распределение x2-квадрат - Отбор признаков с использованием случайного леса
39.	Представлен код Python. Описать кратко действие кода. <pre># Feature Importance with Extra Trees Classifier from pandas import read_csv from sklearn.ensemble import ExtraTreesClassifier # load data url = "https://archive.ics.uci.edu/ml/machine-learning-databases/pima-indians-diabetes/pima-indians-diabetes.data" names = ['preg', 'plas', 'pres', 'skin', 'test', 'mass', 'pedi', 'age', 'class'] dataframe = read_csv(url, names=names) array = dataframe.values</pre>	В представленном примере мы обучаем классификатор ExtraTreesClassifier, чтобы с его помощью определить важность признаков.

Номер задания	Содержание вопроса	Правильный ответ на задание
	<pre> X = array[:,0:8] Y = array[:,8] # feature extraction model = ExtraTreesClassifier() model.fit(X, Y) print(model.feature importances) </pre>	
40.	Представлен график проведённой кросс-валидации на данных. Определить важность признаков. <pre> scores = cross_val_score(estimator=rf, X=X, y=y, cv=skf, scoring='roc_auc', n_jobs=-1) print('scores = {} \nmean score = {:.5f} +/- {:.5f}'.format(scores, scores.mean(), scores.std())) plot_features_scores(model=rf, data=X, target=y, column_names=X.columns, model type='rf') </pre> 	Важными для определения класса являются первые четыре признака указанные на графике.
41.	Сравните понятия «Случайный лес» и «дерево решений»	Случайный лес — это набор из множества деревьев решений.
42.	В популярной Python библиотеке Scikit-learn есть модуль metrics, который можно использовать для вычисления метрик в матрице ошибок. Приведите примеры метрик.	Примеры метрик: 1) Accuracy 2) Precision 3) Recall
43.	В обучающем множестве должно быть задано целевое значение, так как деревья решений — модели, создаваемые на основе обучения с учителем. Какие бывают деревья по типу целевого значения?	По типу переменной выделяют два типа деревьев: 1) дерево классификации — когда целевая переменная дискретная; 2) дерево регрессии — когда целевая переменная непрерывная.
44.	Какой алгоритм поддерживает Python при формировании решающих деревьев?	Python поддерживает алгоритм решающий деревьев CART
45.	Кратко опишите представленный код Python. <pre> model=tree.DecisionTreeClassifier(max_dept h=3) </pre>	Задали глубину дерева решений 3 с помощью параметра max_depth.
46.	Представлено дерево решений Python определения пола по признакам. При попадании в оранжевый лист- точно женщина, при попадании в синий лист-	Лист , в него вошло 64 женщины и 0 мужчин, т.е. найденное количество вопросов эффективно по

Номер задания	Содержание вопроса	Правильный ответ на задание
	точно мужчина, при попадании в белый лист- 50% женщина, 50% мужчина. Дайте описание выделенному листу.	поиску женщин.
47.	Дайте краткое описание понятию Градиентный бустинг. Опишите его применение в Python.	Градиентный бустинг – это продвинутый алгоритм машинного обучения для решения задач классификации и регрессии. Он строит предсказание в виде ансамбля слабых предсказывающих моделей, которыми в основном являются деревья решений. Из нескольких слабых моделей в итоге мы собираем одну, но уже эффективную
48.	Приведите не менее трех типов классификаторов Python.	Три примера можно выбрать из данного списка типов классификаторов Python с помощью библиотеки Scikit-Learn: • Метод k-ближайших соседей (K-Nearest Neighbors); • Метод опорных векторов (Support Vector Machines); • Классификатор дерева решений (Decision Tree Classifier) / Случайный лес (Random Forests); • Наивный байесовский метод (Naive Bayes); • Линейный дискриминантный анализ (Linear Discriminant Analysis); • Логистическая регрессия (Logistic Regression).
49.	Представлена часть кода Python. Поясните действия второй строки кода. <pre>from sklearn.linear_model import LogisticRegression logreg_clf.predict(test_features) logreg_clf.fit(features, labels)</pre>	Создали экземпляр классификатора. Сделать это можно создав переменную и вызвав функцию, связанную с классификатором.
50.	Представлена часть кода Python. Перечислите сформированные датасеты. <pre>df=pd.read_csv('/content/drive/MyDrive/Cola b Notebooks/students.csv', delimiter=',') =df[['Growth','Weight','Sex']]</pre>	Сформировано два набора данных df и df_cut.
1.	При работе Python библиотеки Pandas с данными файла *.csv использовался специальный метод describe(). Опишите	В столбце broadband 25% всех значений не достигают значения 22.500000, 75% значений выше

Номер задания	Содержание вопроса	Правильный ответ на задание																											
	выделенное число полученных статистических данных  <pre>df.describe()</pre> <table border="1"> <thead> <tr> <th></th> <th>broadband</th> <th>mobile</th> </tr> </thead> <tbody> <tr> <td>count</td> <td>177.000000</td> <td>136.000000</td> </tr> <tr> <td>mean</td> <td>72.672825</td> <td>53.788235</td> </tr> <tr> <td>std</td> <td>66.461751</td> <td>43.327121</td> </tr> <tr> <td>min</td> <td>4.010000</td> <td>7.270000</td> </tr> <tr> <td>25%</td> <td>22.500000</td> <td>24.402500</td> </tr> <tr> <td>50%</td> <td>43.480000</td> <td>35.825000</td> </tr> <tr> <td>75%</td> <td>109.410000</td> <td>69.205000</td> </tr> <tr> <td>max</td> <td>261.820000</td> <td>238.060000</td> </tr> </tbody> </table>		broadband	mobile	count	177.000000	136.000000	mean	72.672825	53.788235	std	66.461751	43.327121	min	4.010000	7.270000	25%	22.500000	24.402500	50%	43.480000	35.825000	75%	109.410000	69.205000	max	261.820000	238.060000	22.500000 (25 процентиль равен 22.500000).
	broadband	mobile																											
count	177.000000	136.000000																											
mean	72.672825	53.788235																											
std	66.461751	43.327121																											
min	4.010000	7.270000																											
25%	22.500000	24.402500																											
50%	43.480000	35.825000																											
75%	109.410000	69.205000																											
max	261.820000	238.060000																											
2.	При работе Python библиотеки Pandas с данными файла *.csv использовался специальный метод describe(). Опишите выделенное число полученных статистических данных  <pre>df.describe()</pre> <table border="1"> <thead> <tr> <th></th> <th>broadband</th> <th>mobile</th> </tr> </thead> <tbody> <tr> <td>count</td> <td>177.000000</td> <td>136.000000</td> </tr> <tr> <td>mean</td> <td>72.672825</td> <td>53.788235</td> </tr> <tr> <td>std</td> <td>66.461751</td> <td>43.327121</td> </tr> <tr> <td>min</td> <td>4.010000</td> <td>7.270000</td> </tr> <tr> <td>25%</td> <td>22.500000</td> <td>24.402500</td> </tr> <tr> <td>50%</td> <td>43.480000</td> <td>35.825000</td> </tr> <tr> <td>75%</td> <td>109.410000</td> <td>69.205000</td> </tr> <tr> <td>max</td> <td>261.820000</td> <td>238.060000</td> </tr> </tbody> </table>		broadband	mobile	count	177.000000	136.000000	mean	72.672825	53.788235	std	66.461751	43.327121	min	4.010000	7.270000	25%	22.500000	24.402500	50%	43.480000	35.825000	75%	109.410000	69.205000	max	261.820000	238.060000	В столбце broadband количество заполненных строк в 177.000000. В столбце mobile количество заполненных строк в 136.000000. Заметно что во втором столбце количество пропущенных значений больше.
	broadband	mobile																											
count	177.000000	136.000000																											
mean	72.672825	53.788235																											
std	66.461751	43.327121																											
min	4.010000	7.270000																											
25%	22.500000	24.402500																											
50%	43.480000	35.825000																											
75%	109.410000	69.205000																											
max	261.820000	238.060000																											
3.	При работе Python библиотеки Pandas с данными файла *.csv использовался специальный метод describe(). Опишите выделенное число полученных статистических данных  <pre>df.describe()</pre> <table border="1"> <thead> <tr> <th></th> <th>broadband</th> <th>mobile</th> </tr> </thead> <tbody> <tr> <td>count</td> <td>177.000000</td> <td>136.000000</td> </tr> <tr> <td>mean</td> <td>72.672825</td> <td>53.788235</td> </tr> <tr> <td>std</td> <td>66.461751</td> <td>43.327121</td> </tr> <tr> <td>min</td> <td>4.010000</td> <td>7.270000</td> </tr> <tr> <td>25%</td> <td>22.500000</td> <td>24.402500</td> </tr> <tr> <td>50%</td> <td>43.480000</td> <td>35.825000</td> </tr> <tr> <td>75%</td> <td>109.410000</td> <td>69.205000</td> </tr> <tr> <td>max</td> <td>261.820000</td> <td>238.060000</td> </tr> </tbody> </table>		broadband	mobile	count	177.000000	136.000000	mean	72.672825	53.788235	std	66.461751	43.327121	min	4.010000	7.270000	25%	22.500000	24.402500	50%	43.480000	35.825000	75%	109.410000	69.205000	max	261.820000	238.060000	Среднее значение по столбцу mobile равно 53.788235.
	broadband	mobile																											
count	177.000000	136.000000																											
mean	72.672825	53.788235																											
std	66.461751	43.327121																											
min	4.010000	7.270000																											
25%	22.500000	24.402500																											
50%	43.480000	35.825000																											
75%	109.410000	69.205000																											
max	261.820000	238.060000																											
4.	При работе Python библиотеки Pandas с	Стандартное отклонение (важный																											

Номер задания	Содержание вопроса	Правильный ответ на задание																											
	данными файла *.csv использовался специальный метод describe(). Опишите выделенное число полученных статистических данных  <pre>df.describe()</pre> <table border="1"> <thead> <tr> <th></th> <th>broadband</th> <th>mobile</th> </tr> </thead> <tbody> <tr> <td>count</td> <td>177.000000</td> <td>136.000000</td> </tr> <tr> <td>mean</td> <td>72.672825</td> <td>53.788235</td> </tr> <tr> <td>std</td> <td>66.461751</td> <td>43.327121</td> </tr> <tr> <td>min</td> <td>4.010000</td> <td>7.270000</td> </tr> <tr> <td>25%</td> <td>22.500000</td> <td>24.402500</td> </tr> <tr> <td>50%</td> <td>43.480000</td> <td>35.825000</td> </tr> <tr> <td>75%</td> <td>109.410000</td> <td>69.205000</td> </tr> <tr> <td>max</td> <td>261.820000</td> <td>238.060000</td> </tr> </tbody> </table>		broadband	mobile	count	177.000000	136.000000	mean	72.672825	53.788235	std	66.461751	43.327121	min	4.010000	7.270000	25%	22.500000	24.402500	50%	43.480000	35.825000	75%	109.410000	69.205000	max	261.820000	238.060000	статистический показатель, показывающий разброс значений) по столбцу broadband равно 66.461751.
	broadband	mobile																											
count	177.000000	136.000000																											
mean	72.672825	53.788235																											
std	66.461751	43.327121																											
min	4.010000	7.270000																											
25%	22.500000	24.402500																											
50%	43.480000	35.825000																											
75%	109.410000	69.205000																											
max	261.820000	238.060000																											
5.	При работе Python библиотеки Pandas с данными файла *.csv использовался специальный метод describe(). Опишите выделенное число полученных статистических данных  <pre>df.describe()</pre> <table border="1"> <thead> <tr> <th></th> <th>broadband</th> <th>mobile</th> </tr> </thead> <tbody> <tr> <td>count</td> <td>177.000000</td> <td>136.000000</td> </tr> <tr> <td>mean</td> <td>72.672825</td> <td>53.788235</td> </tr> <tr> <td>std</td> <td>66.461751</td> <td>43.327121</td> </tr> <tr> <td>min</td> <td>4.010000</td> <td>7.270000</td> </tr> <tr> <td>25%</td> <td>22.500000</td> <td>24.402500</td> </tr> <tr> <td>50%</td> <td>43.480000</td> <td>35.825000</td> </tr> <tr> <td>75%</td> <td>109.410000</td> <td>69.205000</td> </tr> <tr> <td>max</td> <td>261.820000</td> <td>238.060000</td> </tr> </tbody> </table>		broadband	mobile	count	177.000000	136.000000	mean	72.672825	53.788235	std	66.461751	43.327121	min	4.010000	7.270000	25%	22.500000	24.402500	50%	43.480000	35.825000	75%	109.410000	69.205000	max	261.820000	238.060000	Максимальное значение по столбцу broadband равно 238.060000.
	broadband	mobile																											
count	177.000000	136.000000																											
mean	72.672825	53.788235																											
std	66.461751	43.327121																											
min	4.010000	7.270000																											
25%	22.500000	24.402500																											
50%	43.480000	35.825000																											
75%	109.410000	69.205000																											
max	261.820000	238.060000																											
6.	Выберите правильный вариант ответа. При работе Python библиотеки Pandas с данными файла выделено несколько столбцов в отдельную таблицу. К набору данных применена команда. Результат? <pre>max_sugar = data['sugar'].max() print(max_sugar)</pre>	Полученный результат — максимальное значение в столбце «sugar»																											
7.	Выберите правильный вариант ответа. При работе Python библиотеки Pandas с данными файла выделено несколько столбцов в отдельную таблицу. К набору данных применена команда. Результат?	Полученный результат — строка таблицы с максимальным значением																											

Номер задания	Содержание вопроса	Правильный ответ на задание
	<pre>str = data[data['sugar']==data['sugar'].max()] print(str)</pre>	
8.	Представлены два графика. Проведите сравнение 	На первом графике (ящик с усами) отображает 7 аномальных выбросов данных. На втором графике аномальных выбросов нет, видимо данные с аномальными выбросами были удалены