

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. В.Ф. УТКИНА**

Кафедра «Автоматики и информационных технологий в управлении»

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДИСЦИПЛИНЫ

Б1.О.31 «ПРАКТИКУМ ПО ПРОГРАММИРОВАНИЮ И АНАЛИЗУ ДАННЫХ »

Направление подготовки

01.03.02 «Прикладная математика и информатика»

Направленность (профиль) подготовки

«Программирование и анализ данных»

Уровень подготовки

академический бакалавриат

Квалификация выпускника – бакалавр

Формы обучения – очная

Рязань 2024 г.

Оценочные материалы – это совокупность учебно-методических материалов (контрольных заданий, описаний форм и процедур), предназначенных для оценки качества освоения обучающимися данной дисциплины как части основной профессиональной образовательной программы.

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

Основная задача – обеспечить оценку уровня сформированности общекультурных, общепрофессиональных и профессиональных компетенций, приобретаемых обучающимися в соответствии с этими требованиями.

Контроль знаний проводится в форме текущего контроля и промежуточной аттестации.

Текущий контроль успеваемости проводится с целью определения степени усвоения учебного материала, своевременного выявления и устранения недостатков в подготовке обучающихся и принятия необходимых мер по совершенствованию методики преподавания учебной дисциплины (модуля), организации работы обучающихся в ходе учебных занятий и оказания им индивидуальной помощи.

К контролю текущей успеваемости относятся проверка знаний, умений и навыков, приобретенных обучающимися в ходе выполнения индивидуальных заданий на практических занятиях и лабораторных работах. При оценивании результатов освоения практических занятий и лабораторных работ применяется шкала оценки «зачтено – не зачтено». Количество лабораторных и практических работ и их тематика определена рабочей программой дисциплины, утвержденной заведующим кафедрой.

Результат выполнения каждого индивидуального задания должен соответствовать всем критериям оценки в соответствии с компетенциями, установленными для заданного раздела дисциплины.

Промежуточный контроль по дисциплине осуществляется проведением зачета с оценкой.

Форма проведения – письменный ответ по утвержденным экзаменационным билетам, сформулированным с учетом содержания учебной дисциплины. После выполнения письменной работы обучаемого производится ее оценка преподавателем и, при необходимости, проводится теоретическая беседа с обучаемым для уточнения экзаменационной оценки.

Паспорт оценочных материалов по дисциплине

№ п/п	Контролируемые разделы (темы) дисциплины (результаты по разделам)	Код контролируемой компетенции (или её части)	Вид, метод, форма оценочного мероприятия
1	2	3	4
1	<i>Раздел 1</i> Основы ООП в Python	ОПК-5.1; ОПК-5.2	Зачет с оценкой
2	<i>Раздел 2</i> Библиотека Tkinter. Основы разработки оконных приложений.	ОПК-5.1; ОПК-5.2	Зачет с оценкой
3	<i>Раздел 3</i> Регулярные выражения. Работа с датой и временем.	ОПК-5.1; ОПК-5.2	Зачет с оценкой

Критерии оценивания компетенций (результатов)

- 1). Уровень усвоения материала, предусмотренного программой.
- 2). Умение анализировать материал, устанавливать причинно-следственные связи.
- 3). Ответы на вопросы: полнота, аргументированность, убежденность, умение
- 4). Качество ответа (его общая композиция, логичность, убежденность, общая эрудиция)
- 5). Использование дополнительной литературы при подготовке ответов.

Уровень освоения сформированности знаний, умений и навыков по дисциплине оценивается в форме бальной отметки:

«Отлично» заслуживает студент, обнаруживший всестороннее, систематическое и глубокое знание учебно-программного материала, умение свободно выполнять задания, предусмотренные программой, усвоивший основную и знакомый с дополнительной литературой, рекомендованной программой. Как правило, оценка «отлично» выставляется студентам, усвоившим взаимосвязь основных понятий дисциплины в их значении для приобретаемой профессии, проявившим творческие способности в понимании, изложении и использовании учебно-программного материала.

«Хорошо» заслуживает студент, обнаруживший полное знание учебно-программного материала, успешно выполняющий предусмотренные в программе задания, усвоивший основную литературу, рекомендованную в программе. Как правило, оценка «хорошо» выставляется студентам, показавшим систематический характер знаний по дисциплине и способным к их самостоятельному пополнению и обновлению в ходе дальнейшей учебной работы и профессиональной деятельности.

«Удовлетворительно» заслуживает студент, обнаруживший знания основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по специальности, справляющийся с выполнением заданий, предусмотренных программой, знакомый с основной литературой, рекомендованной программой. Как правило, оценка «удовлетворительно» выставляется студентам, допустившим погрешности в ответе на экзамене и при выполнении экзаменационных заданий, но обладающим необходимыми знаниями для их устранения под руководством преподавателя.

«Неудовлетворительно» выставляется студенту, обнаружившему пробелы в знаниях основного учебно-программного материала, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий. Как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение или приступить к профессиональной деятельности по окончании вуза без дополнительных занятий по соответствующей дисциплине.

Типовые контрольные задания или иные материалы

Вопросы к зачету с оценкой по дисциплине

1. Основные отличительные особенности языка Python.
2. Меню и панель инструментов среды Jupyter Notebook.
3. Магические команды Jupyter Notebook.
4. Типы и модель данных языка Python.
5. Функции ввода и вывода языка Python.
6. Арифметические операции языка Python.
7. Математические функции языка Python.
8. Оператор if языка Python.
9. Циклы в языке Python.
10. Коллекции в языке Python.
11. Списки в языке Python.
12. Кортежи в языке Python.
13. Множества в языке Python.
14. Словари в языке Python.
15. Срезы списков в языке Python.
16. Работа со строками в языке Python.
17. Пользовательские функции в языке Python.
18. Создание массивов array с использованием NumPy.
19. Создание специальных массивов с использованием NumPy.
20. Обращение к элементам и фрагментам массива array.
21. Представления и изменение размерностей массива array.
22. Конкатенация и разделение массивов array.
23. Срезы массивов array.
24. Основные операции над массивами array.

25. Транслирование массивов array.
26. Создание двумерных графиков с использованием функции plot.
27. Построение нескольких графиков в одних координатных осях и их форматирование.
28. Метки координатных осей и легенды.
29. Построение нескольких графиков в разных осях.
30. Создание ступенчатых, стековых, stem и точечных графиков.
31. Создание гистограмм, столбчатых и круговых диаграмм.
32. Показ изображений в Matplotlib.
33. Создание трехмерных графиков и их форматирование.
31. Представления в SQLite.
32. Индексирование таблиц в SQLite.
33. Транзакции в SQLite.
34. Триггеры в SQLite..

Типовые задания для самостоятельной работы

1. Быстродействие программ на языке Python.
2. Отладка программ в среде Jupyter Notebook.
3. Команда %timeit в среде Jupyter Notebook.
4. Форматированный вывод в языке Python.
5. Работа с файлами в Python.
6. Модули и пакеты в Python.
7. Работа с исключениями в Python.
8. Модуль linalg библиотеки NumPy.
9. Множества (unique) и операции над ними в NumPy.
10. Файловый ввод-вывод массивов array.
11. Генерация псевдослучайных чисел в NumPy.
12. Сортировка массивов array.
13. Логарифмический масштаб у координатных осей в Matplotlib.
14. Текстовые элементы графиков Matplotlib.
15. Функции contour, contourf и tricontour, tricontourf в Matplotlib.
16. Работа с Anaconda Navigator.
17. IDE для языка Python.

Лабораторный практикум

№ п/п	№ раздела дисциплины	Наименование лабораторной работы	Трудоемкость, час
1	1	Основы ООП в Python	12
2	1	Библиотека Tkinter. Основы разработки оконных приложений.	12
3	2	Регулярные выражения. Работа с датой и временем.	8

СПИСОК
заданий на проверку знания
основ программирования на языке Python

Задание 1. Составить линейный алгоритм и программу для предложенного варианта, в которой вводятся исходные данные, вычисляются заданные арифметические выражения и выводятся на экран дисплея результаты вычислений (исходные данные выбираются произвольно, исходя из области допустимых значений функции).

Вариант 1

$$y = \sqrt{a_0 + a_1 x^2} + 2x^5 + \frac{0,13725 \cdot 10^{-3} + a_0^2}{2 + \sin^2 3x};$$
$$z = a_0 \sqrt[3]{x} + \ln|1 + a_0 + a_1 x^3|.$$

Вариант 2

$$y_1 = \ln|x^3| + \operatorname{tg} \alpha - e^{\alpha x^2 + x};$$
$$y_2 = \lg|a^7| + \operatorname{arctg} x^2 + \frac{\pi + 4,56 \cdot 10^{-4}}{\sqrt[4]{|a + x|}}.$$

Вариант 3

$$\alpha = 5a^{2x}(a + x) - \sqrt{|\cos x^2|}; y = 18ax + \sqrt[3]{y^2} + \sin \frac{\alpha}{2}.$$

Вариант 4

$$\omega = 5b^3 + \sin(ax^2 + b); z = \frac{x - a}{(1 + 2x)^{a^3}} - e^{\sqrt{1 + \omega^2}}.$$

Вариант 5

$$y = \sqrt[3]{a^2 + b^2}; z = \left| \sin^2 \frac{a}{2} \right| \cdot \cos \frac{b}{3}; x = \frac{(y^2 + z^3)}{e^{ay + b^2 z}}.$$

Вариант 6

$$x = \sqrt[5]{\frac{(a+b)}{(1+a)}}; z = e^{\sqrt{x-1}}; y = \frac{2\sin x + \cos \frac{x}{2}}{3 + \cos^2 x}.$$

Вариант 7

$$y = \frac{x + x^3 + \sqrt{x}}{e^{ax+3}}; z = \left| \sin \frac{y}{2} \right| \cdot \cos \frac{a}{2} + \sqrt{a^2 + x^3}.$$

Вариант 8

$$\alpha = \frac{\pi + (a + x^2)^3}{\ln|a + x| + 3,45 \cdot 10^{-3}}; \beta = \frac{1}{1 + \frac{1+x}{1 + \frac{1+x}{1+x^2}}}.$$

Вариант 9

$$a = \sin x; b = \cos^2 x; c = 1 + \operatorname{tg} x;$$

$$y = 4a^2 + x(b^3 + x(c^2 + x(ab + (a+b)^x))).$$

Вариант 10

$$\alpha = 1 + e^{-\sqrt{|2x|}}; \beta = \frac{e^x + e^{-x}}{1 + e^{-2x}}; y = \operatorname{ctg} \frac{\alpha}{3} + \frac{\cos \beta + \sin^2 3\alpha}{1 + \sqrt[3]{1+x}}.$$

Вариант 11

$$y = \frac{a \cdot \sqrt{\sin x + 3} \cdot e^{-ax}}{\sqrt[3]{\ln(2b + c^2) + b^{ax}}}; z = \left(\frac{a^2}{a+b} + \frac{c}{ax^2 + bx} \right)^5 + \sin \frac{y}{3}.$$

Вариант 12

$$\alpha = e^{2x} - e^{-2x}; y = \left(\sqrt{\frac{ax^2 + b}{a^2x + b^3}} + \operatorname{tg} x \right)^{2/3}.$$

Вариант 13

$$y = \frac{(\operatorname{arctg} x^3 + \cos \sqrt{x})^{2x}}{e^x + \ln|2,4x^2|}; z = ay^5 + b \cdot \cos|y| + \operatorname{arctg} y^x.$$

Вариант 14

$$y = \frac{3x^5 + 25e^{x^2}}{|x^{10}| + \sqrt{ax^3 + 2}} + \ln(x+1); z = \pi x^2 - a^2 x + a \cdot \sin\left(\frac{x}{3} + \frac{\pi}{8}\right).$$

Вариант 15

$$z = \frac{x^2 + \frac{x}{2}}{e^x + \sin^3 x} + 16 \cdot e^{x^2} \cdot \ln x^2;$$

$$y = a + \frac{x}{7,5 - 3,2x^2} + \frac{x^3 \cdot (a-1)}{\ln|x^3 - 6|}.$$

Вариант 16

$$z = \frac{b + b^3}{e^x + \sin^2 2x} + \frac{3,45 \cdot 10^{-6} + x^2}{\sqrt{bx^2 + x + 1}}; y = \frac{\pi}{8} \cdot x^2 + \frac{\pi}{4} \cdot x + \frac{\pi}{2}.$$

Вариант 17

$$z = -a^2 x^5 + b \cdot \sin x + \frac{x^2}{x + e^x} - \ln|x^2 + 2x + 3|;$$

$$y = a^3 \cdot \left(\frac{x}{a+x}\right)^{2/3} - 2 \cdot \sin x^2.$$

Вариант 18

$$t = \frac{1}{2\pi} \cdot \sqrt{\frac{m}{c^2}}; z = e^{mc^2} + t^3 \cdot \sin \frac{t}{8} - \operatorname{tg}^2 \frac{t \cdot \pi}{12}.$$

Вариант 19

$$x = \operatorname{arctg} a + \frac{\sin^2 a}{2} - e^{\sqrt{|2a|}}; y = ax^3 + \frac{a}{2} \cdot x^2 + \sqrt{a} \cdot x + e^{-a}.$$

Вариант 20

$$s = \frac{\omega^2 \cdot t}{2} \cdot \sin \frac{\omega}{3} + \omega t \cdot \cos \frac{\omega}{2}; \vartheta = \frac{\omega}{3} \cdot \cos \omega + \frac{\pi + 2}{3} \cdot \operatorname{tg} \frac{\omega}{4}.$$

Задание 2.

Вариант 1. Составить алгоритм и программу вычисления функции $f(x)$ при произвольном x :

$$f(x) = \begin{cases} 2x + 4, & \text{если } -2 \leq x \leq -1; \\ 2x^2, & \text{если } -1 < x \leq 1; \\ -2x + 2, & \text{если } 1 < x \leq 2. \end{cases}$$

При $x < -2$ и $x > 2$ функция $f(x)$ не определена.

Вариант 2. Составить алгоритм и программу вычисления функции $y = f(x)$, график которой изображен на рисунке.

Вариант 3. Составить алгоритм и программу вычисления функции $f(x)$ при произвольном x :

$$f(x) = \begin{cases} 1, & \text{если } x \leq -1; \\ 2x^2 - 1, & \text{если } -1 < x \leq 1; \\ x^2, & \text{если } 1 < x \leq 2. \end{cases}$$

При $x > 2$ функция $f(x)$ не определена.

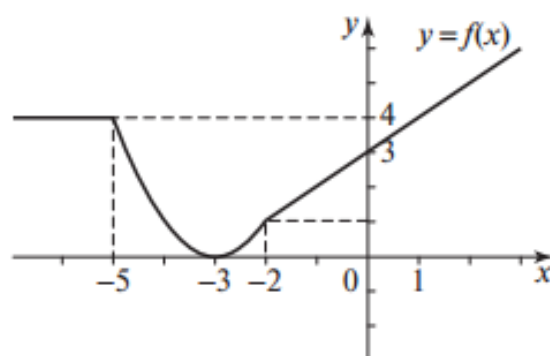


Рис. к варианту 2

Вариант 4. Составить алгоритм и программу, которая определяет, принадлежит ли точка с некоторыми координатами (x, y) заштрихованной области, изображенной на рисунке.

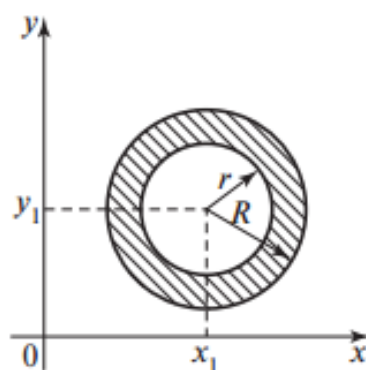


Рис. к варианту 4

Вариант 5. Составить функцию пользователя, определяющую максимальное значение из двух аргументов. Используя функцию, определить максимальное значение из заданных x_1, x_2, x_3, x_4 .

Вариант 6. Заданы длины трех отрезков x_1, x_2 и x_3 . Разработать алгоритм и программу, которая по результатам анализа вводимых длин отрезков выводит на экран дисплея одно из следующих сообщений:

- «треугольник построить нельзя»;
- «разносторонний треугольник»;
- «равнобедренный треугольник»;
- «равносторонний треугольник».

Вариант 7. Заданы следующие параметры геометрических фигур:

x, y, z — стороны треугольника;

a — сторона квадрата;

r — радиус круга.

Вывести на экран дисплея наименование и числовое значение площади фигуры с максимальной площадью.

Вариант 8. Для отрезков a, b и c определить, можно ли из них построить треугольник и является ли этот треугольник прямоугольным (a, b, c — целые числа).

Вариант 9. Составить алгоритм и программу, которая определяет, принадлежит ли точка с координатами (x, y) заштрихованной области, изображенной на рисунке.

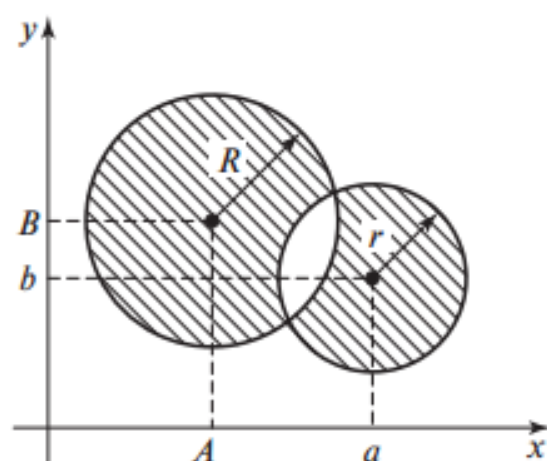


Рис. к варианту 9

Вариант 10. Составить алгоритм и программу вычисления функции $f(x)$ при произвольном x :

$$f(x) = \begin{cases} 2\sin \frac{3x}{4}, & \text{если } x \leq \frac{\pi}{2}; \\ \frac{x}{2} \operatorname{tg} \frac{x+1}{3}, & \text{если } x > \frac{\pi}{2}. \end{cases}$$

В тех случаях, когда тангенс не имеет значений, вывести сообщение «функция не существует».

Вариант 11. Дана функция $f(x) = \sqrt{x}$. Проверить, что для любых произвольно выбранных аргументов $x_1 > 0$ и $x_2 > 0$ имеет место неравенство $f\left(\frac{x_1 + x_2}{2}\right) \geq \frac{f(x_1) + f(x_2)}{2}$.

Вариант 12. Дана функция $y = \frac{x}{4x^2 + 9x}$. Найти значение функции y при произвольно заданных значениях аргумента x_1 , x_2 и x_3 . На экран дисплея вывести минимальное значение функции.

Вариант 13. Составить алгоритм и программу вычисления функции $f(x)$ при произвольном x :

$$f(x) = \begin{cases} \frac{8}{x}, & \text{если } x \leq -2; \\ x^3 + 4, & \text{если } -2 < x \leq 0; \\ \frac{4}{x^2 + 1}, & \text{если } x > 0. \end{cases}$$

Вариант 14. Функция $y = 2 + \sqrt{3 - 5x}$ убывает на $(-\infty; 0,6)$. Выбрать произвольно три значения аргумента $x_1 > x_2 > x_3$ и убедиться, что $f(x_1) > f(x_2) > f(x_3)$.

Вариант 15. Определить знак выражений:

$$\sin \frac{5\pi}{6} \cos \frac{5\pi}{7} \operatorname{tg} \frac{5\pi}{8} \operatorname{ctg} \frac{5\pi}{9};$$

$$\sin \frac{4\pi}{7} \cos \left(-\frac{4\pi}{9}\right) \operatorname{tg} \frac{4\pi}{9} \operatorname{ctg} \left(-\frac{4\pi}{11}\right).$$

Вариант 16. Составить алгоритм и программу, которая определяет, принадлежит ли точка с некоторыми координатами (x, y) заштрихованной области, изображенной на рисунке.

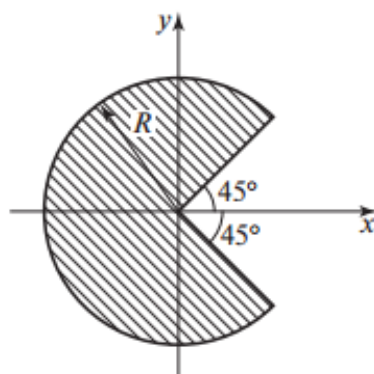


Рис. к варианту 16

Вариант 17. Определить минимальное значение среди заданных x_1, x_2, x_3 и x_4 .

Вариант 18. Составить алгоритм и программу вычисления функции $f(x)$ при произвольном x :

$$f(x) = \begin{cases} \frac{x}{2} \sqrt[3]{1+x}, & \text{если } 1 \leq x \leq 3; \\ \frac{\sin 2x}{2 + \cos 3x}, & \text{если } x > 3. \end{cases}$$

При $x < 1$ функция $f(x)$ не определена.

Вариант 19. Составить алгоритм и программу вычисления значения функции (см. рисунок) при произвольном значении аргумента x .

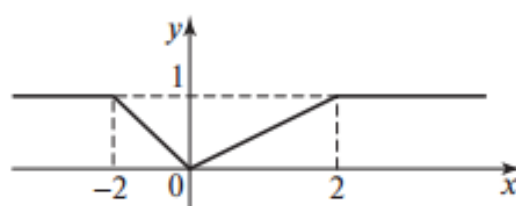


Рис. к варианту 19

Вариант 20. Заданы стороны двух треугольников: (a_1, b_1, c_1) и (a_2, b_2, c_2) . Определить треугольник с максимальной площадью.

СПИСОК
заданий на проверку знания
основ работы с библиотекой NumPy

Задание 1.

Вариант 1. Сформировать матрицу A по правилу

$$a_{ij} = \frac{2b_{ij} + b_{\min}}{b_{\max}},$$

где $b_{\min}$, $b_{\max}$ — минимальный и максимальный элементы произвольно заданной матрицы B размером 3×4 .

Вариант 2. В произвольно заданных матрицах X и Y размером 5×4 определить максимальные элементы и поменять их значения местами. В матрице X все отрицательные элементы заменить максимальным значением.

Вариант 3. Сформировать одномерный массив, состоящий из максимальных значений положительных элементов соответствующих строк произвольно заданной матрицы B размером 5×6 .

Вариант 4. Определить максимальный элемент среди элементов квадратной матрицы размером 5×5 , лежащих выше главной диагонали, и минимальный элемент среди элементов, лежащих ниже главной диагонали. Поменять найденные значения местами.

Вариант 5. Определить максимальный элемент в произвольно заданной матрице Z размером $m \times n$ и обнулить все элементы строки и столбца, на пересечении которых расположено найденное значение.

Вариант 6. Сформировать одномерный массив, каждый элемент которого равен максимальному элементу соответствующего столбца произвольно заданной матрицы размером $m \times n$. В сформированном массиве найти минимальный элемент.

Вариант 7. Сформировать произвольный двумерный массив целочисленных значений размером 6×6 . Определить число повторений каждого из значений первой строки.

Вариант 8. В произвольно заданном двумерном массиве размером 4×5 определить три элемента с наибольшими значениями.

Вариант 9. В произвольно заданном двумерном массиве поменять местами строки, содержащие минимальный и максимальный элементы. Если минимальный и максимальный элементы принадлежат одной строке, то поменять местами соответствующие столбцы.

Вариант 10. В произвольно заданной матрице размером 4×6 определить строку с максимальной суммой элементов и столбец с минимальной суммой.

Вариант 11. Из произвольно заданной матрицы размером 5×5 сформировать построчный одномерный массив из положительных элементов исходной матрицы.

Вариант 12. В двух произвольно заданных двумерных массивах поменять местами строки, содержащие максимальные элементы. Вывести на экран исходные и измененные матрицы.

Вариант 13. Сформировать два произвольных двумерных массива размером 5×4 . Поменять местами столбцы исходных матриц, содержащие минимальные элементы.

Вариант 14. Определить минимальный элемент в произвольно заданной матрице размером 4×6 и заменить на это минимальное значение элементы строки и столбца, которым принадлежат найденное значение.

Вариант 15. Найти минимальное значение главной диагонали и максимальное значение вспомогательной диагонали в квадратной матрице размером 5×5 . Найденные значения поменять местами. Если эти значения принадлежат одному элементу, то это значение присвоить всем элементам главной и вспомогательной диагоналей.

Вариант 16. Определить и поменять местами максимальное и минимальное значения среди элементов, расположенных выше главной и ниже вспомогательной диагоналей в произвольно заданной квадратной матрице размером 6×6 .

Вариант 17. В произвольно заданной матрице A размером 5×4 определить минимальный элемент и обнулить значение элементов, расположенных ниже и правее найденного элемента.

Вариант 18. В трех произвольно заданных положительных матрицах размером 3×3 определить максимальные элементы. Считая найденные значения длинами отрезков, определить возможность построения из них треугольника.

Вариант 19. Сформировать одномерный массив, элементами которого являются средние значения строк произвольно заданной матрицы размером 8×3 . Упорядочить значения одномерного массива по возрастанию.

Вариант 20. Сформировать одномерный массив из положительных элементов произвольно заданной матрицы и упорядочить отобранные значения по убыванию.

СПИСОК

заданий на проверку знания основ работы с библиотекой Matplotlib

Задание 1.

1. Изобразить график функции $y = 2\pi x \cdot e^x$ на интервале (0,5) с шагом 0.02 и отобразить на нем красные окружности (точки) с шагом 0.1. График рисуется дважды: первый раз изображается синяя линия, а второй раз – маркеры в виде красных окружностей:

- отобразить сетку, изменить цвет и вид линий;
- добавить на график текст, используя форматирование LaTeX для вывода греческих символов (размер шрифта 16);
- сохранить рисунок в формате PNG.

2. Построить график функции $y = x^2$ на отрезке $[-10, 10]$ с шагом изменения аргумента 0.01.

При построении использовать третий параметр, который позволяет изменять цвет и тип графика. Значение этого параметра по умолчанию равно 'b-' (сплошная синяя линия без маркеров):

- изобразить точки в виде кружков зеленого цвета;
- представить точки кружочками диаметром 10;
- отобразить сетку, изменить цвет и вид линий;
- добавить на график текст, используя форматирование LaTeX для вывода греческих символов (размер шрифта 16);
- сохранить рисунок в формате PNG.

3. Точечные и линейные графики:

- создать три набора данных (xData, yData1 и yData2);
- создать новый рисунок (под номером 1) шириной 8 дюймов и высотой 6 дюймов;
- задать название графика и подписи для осей x и y (все с размером шрифта 14);
- построить график первого набора данных, yData1, как функции набора данных xData в виде точечного графика с круглыми маркерами и подписью "y1 data";
- построить график второго набора данных, yData2, как функции набора данных xData в виде непрерывной линии красного цвета без маркеров и с подписью "y2 data";
- разместить легенду в верхнем левом углу графика;
- сохранить рисунок в формате PNG.

4. Гистограммы:

- создать набор данных из 1000 нормально распределенных случайных чисел;
- создать новый рисунок (под номером 1) шириной 8 дюймов и высотой 6 дюймов;
- задать название графика и подписи для осей x и y (все с размером шрифта 14);
- построить гистограмму из 40 столбцов с нижней и верхней границами -10 и 10 соответственно;
- добавить на график текст, используя форматирование LaTeX для вывода греческих символов (размер шрифта 16);
- сохранить рисунок в формате PNG.

5. Круговые диаграммы:

- создать набор данных, состоящий из пяти чисел;
- создать новый рисунок (под номером 1) шириной 8 дюймов и высотой 6 дюймов;
- добавить на рисунок оси с соотношением сторон 1:1;
- задать название графика (размер шрифта 14);
- построить график в виде круговой диаграммы;
- сохранить рисунок в формате PNG.

СПИСОК
заданий на проверку знания
основ работы с РБД

Прочитать и интерпретировать запрос на языке SQL:

Запрос 1.

```
SELECT Студ_фам, Студ_№, Студ_стип  
FROM Студенты;
```

Запрос 2.

```
SELECT Студ_фам, Студ_стип  
FROM (SELECT Студ_фам, Студ_№, Студ_стип  
      FROM Студенты);
```

Запрос 3.

```
SELECT Студ_фам, Предм_назв  
FROM Студенты, Предметы;
```

Запрос 4.

```
SELECT *  
FROM Студенты, Предметы;
```

Запрос 5.

```
SELECT Предм_назв, Предм_час, Преп_фам, Преп_имя, Преп_отч  
FROM Предметы INNER JOIN Преподаватели  
      ON Предметы.Преп_№ = Преподаватели.Преп_№;
```

Запрос 6.

```
SELECT Предм_назв, Студ_фам, Экз_оценка  
FROM Студенты INNER JOIN (Предметы INNER JOIN Экзамены  
      ON Предметы.Предм_№ =  
Экзамены.Предм_№)  
      ON Студенты.Студ_№ = Экзамены.Студ_№;
```

Запрос 7.

```
SELECT Студ_№  
FROM Экзамены;
```

Запрос 7а.

```
SELECT DISTINCT Студ_№  
FROM Экзамены;
```

Запрос 8.

```
SELECT Студ_фам, Студ_стип  
FROM Студенты  
WHERE Студ_стип >= 1400;
```

Запрос 9.

```
SELECT *  
FROM Экзамены  
WHERE Экз_оценка > 3 AND Предм_№ = '13';
```

Запрос 10.

```
SELECT *  
FROM Экзамены  
WHERE NOT (Экз_оценка > 3 AND Предм_№ = '13');
```

Запрос 11.

```
SELECT Предм_назв, Предм_час, Преп_фам, Преп_имя, Преп_отч  
FROM Предметы, Преподаватели  
WHERE Предметы.Преп_№ = Преподаватели.Преп_№;
```

Запрос 12.

```
SELECT *  
FROM Студенты  
WHERE Студ_имя In ('Анатолий', 'Владимир');
```

Запрос 13.

```
SELECT *  
FROM Экзамены  
WHERE Экз_оценка BETWEEN 3 AND 5;
```

Запрос 14.

```
SELECT Студ_фам, Студ_имя, Студ_отч  
FROM Студенты  
WHERE Студ_фам BETWEEN 'К' AND 'С';
```

Запрос 15.

```
SELECT Преподаватели.Преп_фам  
FROM Преподаватели  
WHERE Преп_фам LIKE 'К*';
```

Запрос 16.

```
SELECT Преподаватели.Преп_фам  
FROM Преподаватели  
WHERE Преп_фам LIKE '[А-Л]*';
```

Запрос 17.

```
SELECT *  
FROM Экзамены  
WHERE Экз_оценка IS NULL
```

Запрос 18.

```
SELECT Студ_фам, Студ_стип, Студ_стип * 1.05 AS Студ_стип_нов  
FROM Студенты;
```

Запрос 19.

```
SELECT Преп_фам AS Фамилии_на_букву_K  
FROM Преподаватели  
WHERE Преп_фам LIKE 'K*'  
UNION  
SELECT Студ_фам  
FROM Студенты  
WHERE Студ_фам LIKE 'K*';
```

Запрос 20.

```
SELECT 'Преподаватель' AS Должность, Преп_фам AS Фамилии_на_букву_K  
FROM Преподаватели  
WHERE Преп_фам LIKE 'K*'  
UNION  
SELECT 'Студент' AS Должность, Студ_фам AS Фамилии_на_букву_K  
FROM Студенты  
WHERE Студ_фам LIKE 'K*';
```

Запрос 21.

```
SELECT SUM(Студ_стип) AS Суммарная_стипендия  
FROM Студенты;
```

Запрос 22.

```
SELECT COUNT(Студ_стип) AS Количество_студентов  
FROM Студенты;
```

Запрос 23.

```
SELECT AVG(Студ_стип) AS Средняя_стипендия  
FROM Студенты;
```

Запрос 24.

```
SELECT AVG(Студ_стип) AS Средняя_стипендия  
FROM Студенты  
WHERE Студ_стип>0;
```

Запрос 25.

```
SELECT Студ_№, MIN(Экз_оценка) AS Наименьшая_оценка  
FROM Экзамены  
GROUP BY Студ_№;
```

Запрос 26.

```
ELECT Студ_№, Экз_дата, MIN(Экз_оценка) AS Наименьшая_оценка  
FROM Экзамены  
GROUP BY Студ_№, Экз_дата;
```

Запрос 27.

```
SELECT Студ_№, Экз_дата, MIN(Экз_оценка) AS Наименьшая_оценка
FROM Экзамены
GROUP BY Студ_№, Экз_дата
HAVING MIN(Экз_оценка) < 5;
```

Запрос 28.

```
SELECT Студ_№, Экз_дата, Min(Экз_оценка) AS Наименьшая_оценка
FROM Экзамены
GROUP BY Студ_№, Экз_дата
HAVING Экз_дата > #06/10/2008# AND Min(Экз_оценка) < 5;
```

Запрос 29.

- а) SELECT *
 FROM Экзамены
 WHERE Студ_№ = (SELECT Студ_№
 FROM Студенты
 WHERE Студ_фам = 'Петров');
- б) SELECT Экз_№, Предм_№, Экзамены.Студ_№, Экз_оценка, Экз_дата
 FROM Экзамены INNER JOIN Студенты
 ON Экзамены.Студ_№=Студенты.Студ_№
 WHERE Студ_фам = 'Петров';

Запрос 30.

```
SELECT *
FROM Студенты
WHERE Студ_№ IN (SELECT Студ_№
                  FROM Экзамены);
```

Запрос 31.

```
SELECT *
FROM Студенты
WHERE Студ_№ IN (SELECT Студ_№
                  FROM Экзамены
                  WHERE Экз_оценка = 5);
```

Запрос 32.

```
SELECT Экз_оценка, COUNT(Студ_№) AS Количество_экзаменов
FROM Экзамены
GROUP BY Экз_оценка
HAVING Экз_оценка >= (SELECT AVG(Экз_оценка)
                      FROM Экзамены
                      WHERE Предм_№ = '13');
```

Запрос 33.

```
SELECT Студ_№, Студ_фам
FROM Студенты
WHERE 1 < (SELECT COUNT(*)
           FROM Экзамены
           WHERE Студенты.Студ_№ = Экзамены.Студ_№);
```

Запрос 34.

```
SELECT *
FROM Экзамены
WHERE Экз_оценка > (SELECT AVG(Экз_оценка)
                    FROM Экзамены AS Сессия
                    WHERE Экзамены.Предм_№ = Сес-
```

сия.Предм_№);

Запрос 35.

```
SELECT *
FROM Студенты
WHERE EXISTS (SELECT *
              FROM Экзамены
              WHERE Экзамены.Студ_№ = Студенты.Студ_№);
```

Запрос 36.

```
SELECT *
FROM Студенты
WHERE EXISTS (SELECT *
              FROM Экзамены
              WHERE Экзамены.Студ_№ = Студенты.Студ_№
              AND EXISTS (SELECT *
                          FROM Предметы
                          WHERE Предметы.Предм_№
```

=Экзамены.Предм_№

AND Предм_назв =

'Математика'));

Запрос 37.

```
SELECT *
FROM Экзамены
WHERE Экз_оценка > ALL (SELECT Экз_оценка
                        FROM Экзамены
                        WHERE Студ_№ = '8212');
```

Запрос 38.

```
SELECT *
FROM Экзамены
WHERE Экз_оценка > ANY (SELECT Экз_оценка
                        FROM Экзамены
```

WHERE Студ_№ = '8212');

Запрос 39.

```
SELECT *  
FROM Студенты  
ORDER BY Студ_фам;
```

Запрос 40.

```
SELECT *  
FROM Студенты  
ORDER BY Студ_отч, Студ_фам;
```

Запрос 41.

```
SELECT *  
FROM Студенты  
ORDER BY Студ_отч , Студ_фам DESC;
```

Запрос 42.

```
SELECT TOP 3 *  
FROM Студенты  
ORDER BY Студ_стип DESC;
```

Прочитать и интерпретировать запрос на языке SQL:

Запрос 1.

```
SELECT Студ_фам, Студ_№, Студ_стип  
FROM Студенты;
```

Запрос 2.

```
SELECT Студ_фам, Студ_стип  
FROM (SELECT Студ_фам, Студ_№, Студ_стип  
      FROM Студенты);
```

Запрос 3.

```
SELECT Студ_фам, Предм_назв  
FROM Студенты, Предметы;
```

Запрос 4.

```
SELECT *  
FROM Студенты, Предметы;
```

Запрос 5.

```
SELECT Предм_назв, Предм_час, Преп_фам, Преп_имя, Преп_отч  
FROM Предметы INNER JOIN Преподаватели  
ON Предметы.Преп_№ = Преподаватели.Преп_№;
```

Запрос 6.

```
SELECT Предм_назв, Студ_фам, Экз_оценка  
FROM Студенты INNER JOIN (Предметы INNER JOIN Экзамены  
ON Предметы.Предм_№ =
```

Экзамены.Предм_№)

```
ON Студенты.Студ_№ = Экзамены.Студ_№;
```


Запрос 7.

```
SELECT Студ_№  
FROM Экзамены;
```

Запрос 7а.

```
SELECT DISTINCT Студ_№  
FROM Экзамены;
```

Запрос 8.

```
SELECT Студ_фам, Студ_стип  
FROM Студенты  
WHERE Студ_стип >= 1400;
```

Запрос 9.

```
SELECT *  
FROM Экзамены  
WHERE Экз_оценка > 3 AND Предм_№ = '13';
```

Запрос 10.

```
SELECT *  
FROM Экзамены  
WHERE NOT (Экз_оценка > 3 AND Предм_№ = '13');
```

Запрос 11.

```
SELECT Предм_назв, Предм_час, Преп_фам, Преп_имя, Преп_отч  
FROM Предметы, Преподаватели  
WHERE Предметы.Преп_№ = Преподаватели.Преп_№;
```

Запрос 12.

```
SELECT *  
FROM Студенты  
WHERE Студ_имя In ('Анатолий', 'Владимир');
```

Запрос 13.

```
SELECT *  
FROM Экзамены  
WHERE Экз_оценка BETWEEN 3 AND 5;
```

Запрос 14.

```
SELECT Студ_фам, Студ_имя, Студ_отч  
FROM Студенты  
WHERE Студ_фам BETWEEN 'К' AND 'С';
```

Запрос 15.

```
SELECT Преподаватели.Преп_фам  
FROM Преподаватели  
WHERE Преп_фам LIKE 'К*';
```

Запрос 16.

```
SELECT Преподаватели.Преп_фам  
FROM Преподаватели  
WHERE Преп_фам LIKE '[А-Л]*';
```

Запрос 17.

```
SELECT *  
FROM Экзамены  
WHERE Экз_оценка IS NULL
```

Запрос 18.

```
SELECT Студ_фам, Студ_стип, Студ_стип * 1.05 AS Студ_стип_нов  
FROM Студенты;
```

Запрос 19.

```
SELECT Преп_фам AS Фамилии_на_букву_К  
FROM Преподаватели  
WHERE Преп_фам LIKE 'К*'  
UNION  
SELECT Студ_фам  
FROM Студенты  
WHERE Студ_фам LIKE 'К*';
```

Запрос 20.

```
SELECT 'Преподаватель' AS Должность, Преп_фам AS Фамилии_на_букву_К  
FROM Преподаватели  
WHERE Преп_фам LIKE 'К*'  
UNION  
SELECT 'Студент' AS Должность, Студ_фам AS Фамилии_на_букву_К  
FROM Студенты  
WHERE Студ_фам LIKE 'К*';
```

Запрос 21.

```
SELECT SUM(Студ_стип) AS Суммарная_стипендия  
FROM Студенты;
```

Запрос 22.

```
SELECT COUNT(Студ_стип) AS Количество_студентов  
FROM Студенты;
```

Запрос 23.

```
SELECT AVG(Студ_стип) AS Средняя_стипендия  
FROM Студенты;
```

Запрос 24.

```
SELECT AVG(Студ_стип) AS Средняя_стипендия
```

```
FROM Студенты
WHERE Студ_стип>0;
```

Запрос 25.

```
SELECT Студ_№, MIN(Экз_оценка) AS Наименьшая_оценка
FROM Экзамены
GROUP BY Студ_№;
```

Запрос 26.

```
ELECT Студ_№, Экз_дата, MIN(Экз_оценка) AS Наименьшая_оценка
FROM Экзамены
GROUP BY Студ_№, Экз_дата;
```

Запрос 27.

```
SELECT Студ_№, Экз_дата, MIN(Экз_оценка) AS Наименьшая_оценка
FROM Экзамены
GROUP BY Студ_№, Экз_дата
HAVING MIN(Экз_оценка) < 5;
```

Запрос 28.

```
SELECT Студ_№, Экз_дата, Min(Экз_оценка) AS Наименьшая_оценка
FROM Экзамены
GROUP BY Студ_№, Экз_дата
HAVING Экз_дата > #06/10/2008# AND Min(Экз_оценка) < 5;
```

Запрос 29.

- а) SELECT *
 FROM Экзамены
 WHERE Студ_№ = (SELECT Студ_№
 FROM Студенты
 WHERE Студ_фам = 'Петров');
- б) SELECT Экз_№, Предм_№, Экзамены.Студ_№, Экз_оценка, Экз_дата
 FROM Экзамены INNER JOIN Студенты
 ON Экзамены.Студ_№=Студенты.Студ_№
 WHERE Студ_фам = 'Петров';

Запрос 30.

```
SELECT *
FROM Студенты
WHERE Студ_№ IN (SELECT Студ_№
                  FROM Экзамены);
```

Запрос 31.

```
SELECT *
FROM Студенты
```

```
WHERE Студ_№ IN (SELECT Студ_№
                  FROM Экзамены
                  WHERE Экз_оценка = 5);
```

Запрос 32.

```
SELECT Экз_оценка, COUNT(Студ_№) AS Количество_экзаменов
FROM Экзамены
GROUP BY Экз_оценка
HAVING Экз_оценка >= (SELECT AVG(Экз_оценка)
                      FROM Экзамены
                      WHERE Предм_№ = '13');
```

Запрос 33.

```
SELECT Студ_№, Студ_фам
FROM Студенты
WHERE 1 < (SELECT COUNT(*)
          FROM Экзамены
          WHERE Студенты.Студ_№ = Экзамены.Студ_№);
```

Запрос 34.

```
SELECT *
FROM Экзамены
WHERE Экз_оценка > (SELECT AVG(Экз_оценка)
                   FROM Экзамены AS Сессия
                   WHERE Экзамены.Предм_№ = Сес-
```

сия.Предм_№);

Запрос 35.

```
SELECT *
FROM Студенты
WHERE EXISTS (SELECT *
              FROM Экзамены
              WHERE Экзамены.Студ_№ = Студенты.Студ_№);
```

Запрос 36.

```
SELECT *
FROM Студенты
WHERE EXISTS (SELECT *
              FROM Экзамены
              WHERE Экзамены.Студ_№ = Студенты.Студ_№
              AND EXISTS (SELECT *
                          FROM Предметы
                          WHERE Предметы.Предм_№
=Экзамены.Предм_№
                          AND Предм_назв =
'Математика'));
```

Запрос 37.

```
SELECT *
FROM Экзамены
WHERE Экз_оценка > ALL (SELECT Экз_оценка
                        FROM Экзамены
                        WHERE Студ_№ = '8212');
```

Запрос 38.

```
SELECT *
FROM Экзамены
WHERE Экз_оценка > ANY (SELECT Экз_оценка
                        FROM Экзамены
                        WHERE Студ_№ = '8212');
```

Запрос 39.

```
SELECT *
FROM Студенты
ORDER BY Студ_фам;
```

Запрос 40.

```
SELECT *
FROM Студенты
ORDER BY Студ_отч, Студ_фам;
```

Запрос 41.

```
SELECT *
FROM Студенты
ORDER BY Студ_отч , Студ_фам DESC;
```

Запрос 42.

```
SELECT TOP 3 *
FROM Студенты
ORDER BY Студ_стип DESC;
```