

Министерство науки и высшего образования Российской Федерации

ЧИТИНСКИЙ ИНСТИТУТ (ФИЛИАЛ)
ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО ОБРАЗОВАТЕЛЬНОГО
УЧРЕЖДЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ
«БАЙКАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

УТВЕРЖДЕН

на заседании кафедры информационных
технологий и высшей математики
24 февраля 2025 г. протокол № 6

Заведующий кафедрой
Л.И. Трухина



**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ
(ФОНД ОЦЕНОЧНЫХ СРЕДСТВ)
ПО УЧЕБНОЙ ДИСЦИПЛИНЕ
Б1.О.20 Программирование**

Направление подготовки: 38.03.05 Бизнес-информатика
Направленность (профиль): Цифровая экономика
Квалификация выпускника: бакалавр

Чита, 2025 г.

Структура
фонда оценочных средств
по дисциплине «Программирование»

№ п/п	Этапы формирования компетенций	Перечень формируемых компетенций	ЗУНы (З.1, У1, Н1...)	Контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы	Описание показателей и критериев оценивания компетенций на различных этапах формирования, описания шкал оценивания
1	Основные конструкции языка и наборы данных	ОПК-3	З. Знать, как разрабатывать алгоритмы и программы, пригодные для практического применения У. Уметь разрабатывать алгоритмы и программы, пригодные для практического применения Н. Владеть	Т, РЗ	5 баллов - тестирование; 10 баллов - решение задач для проверки умений; 10 баллов - решение задач для проверки навыков.

			<p>навыками разработки алгоритмов и программ, пригодных для практического применения</p>		
2	<p>Работа с файлами и базами данных</p>	ОПК-3	<p>З. Знать, как разрабатывать алгоритмы и программы, пригодные для практического применения</p> <p>У. Уметь разрабатывать алгоритмы и программы, пригодные для практического применения</p> <p>Н. Владеть навыками разработки алгоритмов и программ, пригодных для практического</p>	Т, РЗ	<p>5 баллов - тестирование; 10 баллов - решение задач для проверки умений; 10 баллов - решение задач для проверки навыков.</p>

			применения		
3	Графический интерфейс и модульность	ОПК-3	<p>З. Знать, как разрабатывать алгоритмы и программы, пригодные для практического применения</p> <p>У. Уметь разрабатывать алгоритмы и программы, пригодные для практического применения</p> <p>Н. Владеть навыками разработки алгоритмов и программ, пригодных для практического применения</p>	T, PЗ	<p>5 баллов - тестирование; 10 баллов - решение задач для проверки умений;</p> <p>10 баллов - решение задач для проверки навыков.</p>
4	Разработка игры	ОПК-3	<p>З. Знать, как разрабатывать алгоритмы и программы, пригодные для</p>	T, PЗ	<p>5 баллов - тестирование; 10 баллов - решение задач для проверки умений;</p>

			<p>практического применения</p> <p>У. Уметь разрабатывать алгоритмы и программы, пригодные для практического применения</p> <p>Н. Владеть навыками разработки алгоритмов и программ, пригодных для практического применения</p>		<p>10 баллов - решение задач для проверки навыков.</p>
5	Итого по текущей аттестации	ОПК-3			Всего 100 баллов.
6	Промежуточная аттестация	ОПК-3	<p>3. Знать, как разрабатывать алгоритмы и программы, пригодные для практического применения</p> <p>У. Уметь</p>	Т, РЗ	<p>20 баллов - тестирование; 40 баллов - решение задач для проверки умений; 40 баллов - решение задач для проверки</p>

			разрабатывать алгоритмы и программы, пригодные для практического применения Н. Владеть навыками разработки алгоритмов и программ, пригодных для практического применения		навыков. 100 баллов
--	--	--	---	--	----------------------------

Министерство науки и высшего образования Российской Федерации
ЧИТИНСКИЙ ИНСТИТУТ (ФИЛИАЛ)
ФГБОУ ВО «БАЙКАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Кафедра информационных технологий и высшей математики

Оценочные средства текущего контроля:

- Тема 1.1. Тест, решение задач (Приложение 1)
- Тема 1.2. Тест, решение задач (Приложение 2)
- Тема 1.3. Тест, решение задач (Приложение 3)
- Тема 1.4. Тест, решение задач (Приложение 4)

Оценочные средства промежуточного контроля:

Материалы для промежуточного контроля в виде Экзамена в семестре 2.1.
(Приложение 5)

Билеты к экзамену в 1-м семестре на 2-м курсе
(материалы к экзамену приведены в Приложении 5)

Министерство науки и высшего образования
Российской Федерации
Читинский институт (филиал)
Федерального государственного бюджетного
образовательного учреждения
высшего образования
«БАЙКАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ»
(ЧИ ФГБОУ ВО «БГУ»)

Направление - 38.03.05
Бизнес-информатика
Профиль - Цифровая экономика
Кафедра информационных
технологий и высшей математики
Дисциплина –
Программирование

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ

1. Тест.
2. Две случайно выбранные задачи для проверки умений.
3. Две случайно выбранные задачи для проверки навыков.

Составитель _____ С.В. Бочкарев
Заведующий кафедрой _____ Л.И. Трухина

Приложение 1. Материалы для текущего контроля по теме 1.1.

1. Вопросы для проверки знаний:

1. Какой оператор используется для ветвления в Python?

- a) loop
- b) if
- c) switch
- d) select

2. Какой цикл выполняется, пока условие истинно?

- a) for
- b) while
- c) loop
- d) repeat

3. Как добавить элемент в список lst?

- a) lst.add(5)
- b) lst.append(5)
- c) lst.insert(5)
- d) lst.push(5)

4. Как получить длину списка lst?

- a) len(lst)
- b) lst.length()
- c) lst.size()
- d) length(lst)

5. Как создать кортеж из одного элемента?

- a) t = (1)
- b) t = (1,)
- c) t = [1]
- d) t = tuple(1)

6. Как проверить наличие ключа key в словаре d?

- a) d.has_key(key)
- b) key in d
- c) d.exists(key)
- d) d.contains(key)

7. Как удалить элемент с ключом key из словаря d?

- a) d.remove(key)
- b) del d[key]
- c) d.pop(key)
- d) d.delete(key)

8. Как объявить множество?

- a) `s = list()`
- b) `s = set()`
- c) `s = []`
- d) `s = ()`

9. Какой метод строки разбивает её по разделителю?

- a) `join()`
- b) `split()`
- c) `divide()`
- d) `partition()`

10. Какой модуль используется для регулярных выражений?

- a) `regex`
- b) `re`
- c) `string`
- d) `pattern`

11. Какой тип данных неизменяем?

- a) Список
- b) Кортеж
- c) Словарь
- d) Множество

12. Как получить последний элемент списка `lst`?

- a) `lst[last]`
- b) `lst[-1]`
- c) `lst[len(lst)]`
- d) `lst.get(-1)`

13. Как отсортировать список `lst` по убыванию?

- a) `lst.sort()`
- b) `lst.sort(reverse=True)`
- c) `sorted(lst)`
- d) `lst.reverse()`

14. Как получить список ключей словаря `d`?

- a) `d.keys()`
- b) `list(d.keys())`
- c) `d.items()`
- d) `d.values()`

15. Как удалить дубликаты из списка `lst`?

- a) `lst.unique()`
- b) `list(set(lst))`
- c) `lst.distinct()`
- d) `remove_duplicates(lst)`

16. Какой метод добавляет элементы в конец списка?

- a) insert()
- b) extend()
- c) append()
- d) add()

17. Как получить индекс элемента x в списке lst?

- a) lst.find(x)
- b) lst.index(x)
- c) index(x, lst)
- d) lst.position(x)

18. Как перебрать пары ключ-значение в словаре?

- a) for key in d:
- b) for key, value in d.items():
- c) for pair in d:
- d) for value in d.values():

19. Как преобразовать строку s в нижний регистр?

- a) s.toLower()
- b) s.lower()
- c) s.casefold()
- d) s.downcase()

20. Как добавить элемент в множество s?

- a) s.append(x)
- b) s.add(x)
- c) s.insert(x)
- d) s.update(x)

21. Какой метод строки удаляет пробелы с обоих концов?

- a) trim()
- b) strip()
- c) clean()
- d) cut()

22. Какой оператор возвращает значение из функции?

- a) yield
- b) return
- c) break
- d) exit

23. Какой тип данных используется для хранения пар ключ-значение?

- a) Список
- b) Словарь
- c) Множество
- d) Кортеж

24. Как получить список значений словаря d?

- a) d.keys()
- b) d.values()
- c) d.items()
- d) d.get_values()

25. Какой тип данных неупорядочен?

- a) Список
- b) Кортеж
- c) Множество
- d) Строка

2. Задачи для проверки умений:

Задача 1. Слияние интервалов

Дан список интервалов, некоторые из которых пересекаются. Требуется получить новый список, в котором пересекающиеся интервалы будут объединены.

Например: `[[1,3],[2,6],[8,10],[15,18]]` \Rightarrow `[[1,6],[8,10],[15,18]]`

Задача 2. Проверка валидности скобок с учетом вложенных типов

Дана строка, в которой расставлены скобки разных видов `()`, `[]`, `{}`. Требуется определить верно ли расставлены скобки.

Например: `"([{}])"` \Rightarrow `True` или `"([{}])"` \Rightarrow `False`

Задача 3. Поиск всех анаграмм в списке слов

Дан список слов. Требуется получить новый список, где слова-анаграммы сгруппированы вместе. Например:

`["eat","tea","tan","ate","nat","bat"]` \Rightarrow `["eat","tea","ate"], ["tan","nat"], ["bat"]`

Задача 4. Поиск k-го наибольшего элемента в массиве

Дан массив натуральных чисел и номер k. Требуется найти k-й наибольший элемент без полной сортировки массива.

Например: `[3,2,1,5,6,4]`, 2 \Rightarrow 5

Задача 5. Подсчет островов на матрице

Дана матрица заполненная нулями и единицами. Требуется найти количество "островов" из 1 в матрице.

`[['1','1','0','0','0'],
 ['1','1','0','0','0'],
 ['0','0','1','0','0'],
 ['0','0','0','1','1']]` \Rightarrow 3

Задача 6. Поиск минимального окна подстроки

Дана длинная строка s и короткая строка t . Требуется найти в строке s минимальную подстроку, содержащую все символы строки t .

Например: "ADOBECODEBANC", "ABC" \Rightarrow "BANC"

Задача 7. Найти все уникальные тройки чисел, дающих сумму 0

Дан массив произвольных целых чисел и число k . Требуется получить массив различных троек чисел, составленных из чисел заданного массива, сумма которых равна числу k .

Например: [-1, 0, 1, 2, -1, -4], 0 \Rightarrow [[-1, -1, 2], [-1, 0, 1]]

3. Задачи для проверки навыков:

Задача 1. Найти пропущенное число

Дан массив натуральных чисел от 0 до n с одним пропущенным числом. Требуется найти его.

Например: [3, 0, 1] \Rightarrow 2

Задача 2. Проверить, можно ли преобразовать строку в другую строку сдвигом букв

Даны две строки. Требуется проверить, можно ли получить вторую строку из первой циклическим сдвигом.

Например: "abcde", "cdeab" \Rightarrow True или "abc", "acb" \Rightarrow False

Задача 3. Найти все числа, исчезнувшие в массиве

Дан список чисел от 1 до n , некоторые числа пропущены. Найти все пропущенные числа.

Например: [4, 3, 2, 7, 8, 2, 3, 1] \Rightarrow [5, 6]

Задача 4. Проверить, является ли число палиндромом

Дано натуральное число. Требуется проверить, является ли целое число палиндромом без преобразования в строку. Число-палиндром – это число, которое читается одинаково слева направо и справа налево.

Например: 121 \Rightarrow True или 123 \Rightarrow False

Задача 5. Найти ближайший меньший элемент для каждого элемента массива

Дан массив натуральных чисел. Требуется получить новый массив, где каждый элемент заданного массива заменён на ближайший слева элемент этого массива, который меньше заменяемого элемента, или на -1, если такого элемента нет.

Например: [4, 5, 2, 10, 8] \Rightarrow [-1, 4, -1, 2, 2]

Задача 6. Найти все возможные перестановки массива

Дан массив натуральных чисел. Требуется получить массив всех возможных перестановок элементов этого массива.

Например: $[1, 2, 3] \Rightarrow [[1, 2, 3], [1, 3, 2], [2, 1, 3], [2, 3, 1], [3, 2, 1], [3, 1, 2]]$

Задача 7. Найти подмассив с максимальным произведением

Дан массив с произвольными целыми числами. Требуется найти подмассив, произведение элементов которого максимально. Подмассив может содержать любое количество подряд идущих чисел заданного массива.

Например: $[2, 3, -2, 4] \Rightarrow [2, 3]$

Задача 8. Найти все возможные подмножества массива

Дан массив произвольных натуральных чисел. Требуется получить массив всех возможных подмножеств массива (без дубликатов).

Например: $[1, 2, 3] \Rightarrow [[], [1], [2], [1, 2], [3], [1, 3], [2, 3], [1, 2, 3]]$

Приложение 2. Материалы для текущего контроля по теме 1.2.

1. Вопросы для проверки знаний:

1. Какой режим открытия файла используется для записи с удалением содержимого?

- a) 'a'
- b) 'w'
- c) 'r+'
- d) 'x'

2. Как правильно открыть файл для автоматического закрытия?

- a) `file = open("data.txt")`
- b) `with open("data.txt") as file:`
- c) `open("data.txt").close()`
- d) `using open("data.txt")`

3. Как прочитать все содержимое файла в строку?

- a) `file.read(100)`
- b) `file.read()`
- c) `file.readlines()`
- d) `file.readline()`

4. Как записать список строк в файл?

- a) `file.save(lines)`
- b) `file.writelines(lines)`
- c) `file.write(lines)`
- d) `file.append(lines)`

5. Как подключиться к SQLite базе данных?

- a) `import mysql.connector`
- b) `import sqlite3`
- c) `import database`
- d) `import psycopg2`

6. Какой SQL-запрос создает таблицу?

- a) `MAKE TABLE users`
- b) `CREATE TABLE users (id INTEGER)`
- c) `NEW TABLE users`
- d) `TABLE CREATE users`

7. Как вставить данные в таблицу?

- a) `INSERT TO users VALUES (1)`
- b) `INSERT INTO users VALUES (1)`
- c) `ADD TO users VALUES (1)`
- d) `UPDATE users SET id = 1`

8. Как выбрать все данные из таблицы?

- a) GET * FROM users
- b) SELECT * FROM users
- c) FIND * IN users
- d) READ * FROM users

9. Какой метод выполняет SQL-запрос?

- a) cursor.run()
- b) cursor.execute()
- c) cursor.do()
- d) cursor.query()

10. Как получить все результаты запроса?

- a) cursor.get()
- b) cursor.fetchall()
- c) cursor.results()
- d) cursor.all()

11. Какой метод файла читает одну строку?

- a) file.read()
- b) file.readline()
- c) file.getline()
- d) file.line()

12. Какой режим открывает файл для чтения и записи?

- a) 'rw'
- b) 'r+'
- c) 'a+'
- d) 'w+'

13. Как переместить указатель в файле?

- a) file.move()
- b) file.seek()
- c) file.position()
- d) file.goto()

14. Как обновить данные в таблице?

- a) CHANGE users SET name='John'
- b) UPDATE users SET name='John'
- c) MODIFY users SET name='John'
- d) ALTER users SET name='John'

15. Как удалить данные из таблицы?

- a) REMOVE FROM users WHERE id=1
- b) DELETE FROM users WHERE id=1
- c) DROP FROM users WHERE id=1
- d) CLEAR users WHERE id=1

16. Как создать индекс в SQL?

- a) MAKE INDEX idx_name ON users(name)
- b) CREATE INDEX idx_name ON users(name)
- c) ADD INDEX idx_name FOR users(name)
- d) INDEX idx_name ON users(name)

17. Как получить количество записей в таблице?

- a) COUNT * FROM users
- b) SELECT COUNT(*) FROM users
- c) SUM(*) FROM users
- d) LEN(*) FROM users

18. Какой оператор группирует данные?

- a) ORDER BY
- b) GROUP BY
- c) SORT BY
- d) COLLECT BY

19. Какой оператор фильтрует группы?

- a) WHERE
- b) HAVING
- c) FILTER
- d) GROUP WHERE

20. Какой метод возвращает одну строку результата?

- a) cursor.fetch()
- b) cursor.fetchone()
- c) cursor.getone()
- d) cursor.first()

21. Какой метод используется для массовой вставки?

- a) cursor.insertmany()
- b) cursor.executemany()
- c) cursor.bulk_insert()
- d) cursor.many()

22. Какой оператор SQL изменяет структуру таблицы?

- a) UPDATE
- b) ALTER
- c) MODIFY
- d) CHANGE

23. Как удалить таблицу?

- a) DELETE TABLE users
- b) DROP TABLE users
- c) REMOVE TABLE users
- d) CLEAR TABLE users

24. Как получить список таблиц в SQLite?

- a) SHOW TABLES
- b) SELECT name FROM sqlite_master WHERE type='table'
- c) LIST TABLES
- d) GET TABLES

25. Какой метод предотвращает SQL-инъекции?

- a) Конкатенация строк
- b) Использование параметризованных запросов
- c) Экранирование символов
- d) Все варианты верны

2. Задачи для проверки умений:

Задача 1. Поиск и замена слова в файле

Напишите программу, которая заменяет все вхождения слова "old" на "new" в файле text.txt и сохраняет результат в new_text.txt.

Задача 2. Объединение двух файлов

Напишите программу, которая объединяет содержимое файлов file1.txt и file2.txt и сохраняет результат в merged.txt.

Задача 3. Подсчёт гласных в файле

Напишите программу, которая считает количество гласных букв (a, e, i, o, u) в файле poem.txt.

Задача 4. Генератор CSV с случайными данными

Напишите программу, которая создаёт CSV-файл random_data.csv с 10 случайными числами (от 1 до 100) в одном столбце.

Задача 5. Добавление данных в SQLite

Напишите программу, которая добавляет 3 записи ((1, 'Alex'), (2, 'Kate'), (3, 'John')) в таблицу users в базе app.db.

Задача 6. Обновление данных в SQLite

Напишите программу, которая обновляет имя пользователя с id = 1 на "Alice" в таблице users базы app.db.

Задача 7. Экспорт SQLite в CSV

Напишите программу, которая экспортирует данные из таблицы users базы app.db в файл users.csv.

3. Задачи для проверки навыков:

Задача 1. Удаление пустых строк из файла

Напишите программу, которая удаляет все пустые строки из файла `input.txt` и сохраняет результат в `output.txt`.

Задача 2. Сортировка строк файла

Напишите программу, которая сортирует строки файла `words.txt` в алфавитном порядке и сохраняет результат в `sorted_words.txt`.

Задача 3. Поиск самой длинной строки в файле

Напишите программу, которая находит и выводит самую длинную строку в файле `lines.txt`.

Задача 4. Фильтрация данных в CSV

Напишите программу, которая читает `data.csv` и сохраняет в `filtered.csv` только строки, где значение во втором столбце больше 50.

Задача 5. Подсчёт частоты слов в файле

Напишите программу, которая подсчитывает, сколько раз каждое слово встречается в файле `text.txt` (без учёта регистра).

Задача 6. Удаление данных из SQLite

Напишите программу, которая удаляет пользователя с `id = 2` из таблицы `users` в базе `app.db`.

Задача 7. Поиск пользователя в SQLite

Напишите программу, которая ищет пользователя по имени "Alice" в таблице `users` базы `app.db` и выводит его `id`.

Задача 8. Резервное копирование SQLite

Напишите программу, которая создаёт резервную копию базы `app.db` в файл `app_backup.db`.

Приложение 3. Материалы для текущего контроля по теме 1.3.

1. Вопросы для проверки знаний:

1. Как создать главное окно в Tkinter?

- a) window = Window()
- b) root = Tk()
- c) main = MainWindow()
- d) app = Application()

2. Какой метод запускает главный цикл обработки событий?

- a) start()
- b) mainloop()
- c) run()
- d) execute()

3. Как добавить кнопку в окно?

- a) addButton(text="Click")
- b) Button(root, text="Click").pack()
- c) new Button(root, "Click")
- d) create_button("Click")

4. Какой метод размещает виджеты в таблицу?

- a) place()
- b) grid()
- c) table()
- d) pack()

5. Как обработать нажатие кнопки?

- a) action
- b) command
- c) onclick
- d) function

6. Как создать текстовую метку?

- a) TextLabel(root, "Hello")
- b) Label(root, text="Hello")
- c) new Label(root, "Hello")
- d) Label.create("Hello")

7. Как получить текст из Entry?

- a) entry.text()
- b) entry.get()
- c) entry.value()
- d) entry.content()

8. Какой виджет используется для многострочного текста?

- a) Entry
- b) Text
- c) Label
- d) Input

9. Как изменить текст Label после создания?

- a) label.text = "New"
- b) label.config(text="New")
- c) label.set("New")
- d) label.change_text("New")

10. Как добавить пункты в Listbox?

- a) listbox.add("Item")
- b) listbox.insert(END, "Item")
- c) listbox.append("Item")
- d) listbox.items("Item")

11. Как обработать событие мыши?

- a) onclick
- b) bind
- c) attach
- d) listen

12. Как создать диалоговое окно сообщения?

- a) Dialog.show("Message")
- b) messagebox.showinfo("Title", "Message")
- c) Popup.message("Message")
- d) Alert("Message")

13. Как изменить цвет фона виджета?

- a) widget.color("red")
- b) widget.config(bg="red")
- c) widget.background("red")
- d) widget.set_bg("red")

14. Как добавить изображение в интерфейс?

- a) ImageWidget()
- b) Label(image=photo)
- c) Picture()
- d) PhotoViewer()

15. Как загрузить изображение для Tkinter?

- a) Image.open()
- b) PhotoImage(file="img.png")
- c) TkImage.load()
- d) Picture.load()

16. Как создать собственный модуль?

- a) Создать файл .module
- b) Создать файл .py с функциями
- c) Использовать new_module()
- d) Только через setup.py

17. Как импортировать функцию из модуля?

- a) include module.func
- b) from module import func
- c) require module.func
- d) using module.func

18. Как организовать код большого приложения?

- a) Все в одном файле
- b) Разделить на модули/пакеты
- c) Использовать только классы
- d) Автоматически генерировать

19. Как обработать закрытие окна?

- a) on_close
- b) protocol("WM_DELETE_WINDOW")
- c) close_event
- d) window_close

20. Как изменить размер окна?

- a) root.resize(300,200)
- b) root.geometry("300x200")
- c) root.size(300,200)
- d) root.set_size(300,200)

21. Как сделать окно не изменяемым?

- a) root.fixed()
- b) root.resizable(False, False)
- c) root.lock_size()
- d) root.set_resizable(0)

22. Как центрировать окно на экране?

- a) root.center()
- b) Рассчитать через wininfo_screenwidth()
- c) root.position("center")
- d) Автоматически

23. Как получить размер виджета?

- a) widget.size()
- b) widget.wininfo_width()
- c) widget.get_size()
- d) widget.measure()

24. Как скрыть/показать виджет?

- a) widget.hide()/widget.show()
- b) pack_forget()/pack()
- c) widget.visible(False)
- d) widget.set_visible()

25. Как обработать правую кнопку мыши?

- a) < Button-1>
- b) <Button-2>
- c) < Button-3>
- d) < Button-4>

2. Задачи для проверки умений:

Задача 1. Кнопка с изменяемым текстом

Создать кнопку, которая меняет текст при нажатии.

Задача 2. Поле ввода с отображением текста

Выводить текст из Entry в Label при нажатии кнопки.

Задача 3. Переключатель темы

Переключать между светлой и темной темой при нажатии кнопки

Задача 4. Простой счетчик кликов

Создать кнопку, которая считает и отображает количество нажатий.

Задача 5. Мини-калькулятор (сложение)

Сложить два числа из Entry и вывести результат.

Задача 6. Флажок (Checkbutton)

Создать флажок для показа/скрытия текста.

Задача 7. Радиокнопки (Radiobutton)

Выбор размера текста с помощью радиокнопок.

3. Задачи для проверки навыков:

Задача 1. Ползунок (Scale) для изменения размера

Менять размер текста с помощью ползунка.

Задача 2. Меню (Menu)

Создать выпадающее меню с действиями.

Задача 3. Диалоговое окно (messagebox)

Показать диалоговое окно с вопросом.

Задача 4. Случайное число

Генерировать случайное число от 1 до 100 при нажатии на кнопку.

Задача 5. Мини-блокнот

Простой текстовый редактор с сохранением записей в файл и чтением записей из файла.

Задача 6. Проверка пароля

Проверить, совпадают ли введенные пароли.

Задача 7. Мини-игра "Угадай число"

Угадать число от 1 до 10. Число вводится в Entry и проверяется при нажатии на кнопку.

Задача 8. Список дел (Listbox)

Добавлять и удалять задачи из списка Listbox.

Приложение 4. Материалы для текущего контроля по теме 1.4.

1. Вопросы для проверки знаний:

1. Какой метод Tkinter используется для создания анимации?

- a) animate()
- b) after()
- c) loop()
- d) update()

2. Как переместить объект на Canvas?

- a) canvas.set_position()
- b) canvas.move()
- c) canvas.relocate()
- d) canvas.shift()

3. Какой параметр after() задает время до выполнения функции (в мс)?

- a) interval
- b) delay
- c) time
- d) duration

4. Как проверить столкновение двух объектов на Canvas?

- a) canvas.collission()
- b) Сравнение координат через coords()
- c) objects_overlap()
- d) check_collission()

5. Как обработать нажатие клавиш в игре?

- a) key_press()
- b) bind("<Key>", handler)
- c) on_key_press()
- d) input_handler()

6. Как сделать движение объекта при зажатой клавише?

- a) Использовать while key_pressed
- b) Обрабатывать события <KeyPress> и <KeyRelease>
- c) Проверять состояние клавиатуры каждый кадр
- d) Через key.ispressed()

7. Как реализовать гравитацию в игре?

- a) object.gravity = True
- b) Увеличивать скорость по Y каждый кадр
- c) canvas.add_gravity()
- d) Использовать физический движок

8. Как загрузить изображение для спрайта?

- a) `open("image.png")`
- b) `PhotoImage(file="image.png")`
- c) `load_image()`
- d) `Image.get()`

9. Как сделать анимацию спрайта?

- a) `sprite.animate()`
- b) Менять изображение через `itemconfig()`
- c) Использовать gif
- d) `Canvas.animation()`

10. Как сохранить рекорд игры?

- a) В оперативной памяти
- b) В файл (JSON/txt)
- c) В облако
- d) В реестр Windows

11. Как сделать плавное движение объекта?

- a) Увеличить скорость
- b) Уменьшить интервал `after()`
- c) Использовать `threading`
- d) `canvas.smooth_move()`

12. Как проверить выход за границы экрана?

- a) `canvas.check_borders()`
- b) Сравнить координаты с размерами Canvas
- c) `object.out_of_bounds()`
- d) Автоматически в Tkinter

13. Как сделать "камеру", следящую за игроком?

- a) `Camera.follow()`
- b) Сместить все объекты в противоположную сторону
- c) `canvas.move_view()`
- d) Использовать `Scrollbar`

14. Как сделать взрыв/анимацию частиц?

- a) `ParticleSystem()`
- b) Несколько анимированных объектов
- c) `canvas.explosion()`
- d) Специальный модуль

15. Как добавить тень объекту?

- a) `object.shadow()`
- b) Дополнительный полупрозрачный объект
- c) `Canvas.effect()`
- d) Через стили CSS

16. Как сделать плавное появление/исчезновение?

- a) widget.fade()
- b) Менять прозрачность через itemconfig
- c) alpha_channel()
- d) Canvas.transition()

17. Как реализовать физику отскоков?

- a) object.bounce()
- b) Изменять направление скорости при столкновении
- c) Физический движок
- d) Canvas.collision_response()

18. Как сделать игру полноэкранной?

- a) root.fullscreen()
- b) root.attributes("-fullscreen", True)
- c) root.maximize()
- d) Установить размер как у экрана

19. Как ускорить рендеринг многих объектов?

- a) Уменьшить FPS
- b) Использовать tag для групповых операций
- c) Рисовать на PIL.Image
- d) Отключить анимации

20. Как добавить тень тексту?

- a) text.shadow()
- b) Вывести тот же текст со смещением и другим цветом
- c) Специальный параметр shadow
- d) Через CSS-стили

21. Как сделать анимацию движения по кривой?

- a) move_along_curve()
- b) Рассчитывать позицию по формуле кривой
- c) Специальный модуль
- d) Canvas.curve_motion()

22. Как добавить "дыхание" объекту (пульсацию)?

- a) object.pulse()
- b) Циклически менять размер
- c) Эффект встроен в Canvas
- d) Через CSS-анимации

23. Как добавить тряску камеры?

- a) Camera.shake()
- b) Случайно смещать все объекты
- c) Встроенный эффект
- d) Анимировать окно

24. Как реализовать систему настроек?

- a) Settings.load()
- b) Хранить в файле и загружать при старте
- c) Реестр Windows
- d) Глобальные переменные

25. Как добавить "заморозку" при ударе?

- a) time.stop()
- b) Пропустить несколько кадров анимации
- c) canvas.freeze()
- d) Специальный эффект

2. Задачи для проверки умений:

Задача 1. Прыгающий мяч

Анимация прыгающего мяча

Задача 2. Кликер-игра

Считать клики по кнопке за 10 секунд

Задача 3. Игра "Поймай шарик"

Кликать по появляющимся шарикам

Задача 4. Анимация радуги

Плавно менять цвет фона

Задача 5. Простая анимация загрузки

Анимировать вращающийся круг

Задача 6. Игра "Убеги от курсора"

Квадрат убегает от курсора мыши

Задача 7. Игра "Найди пару" (упрощенная)

Открывать две карточки с числами

3. Задачи для проверки навыков:

Задача 1. Анимация взрыва

Анимация расходящихся кругов

Задача 2. Игра "Космический корабль"

Управлять кораблем клавишами

Задача 3. Анимация текста

Текст, меняющий размер и цвет

Задача 4. Бегущая строка

Анимация бегущей строки

Задача 5. Летающий НЛО

Анимация летающего НЛО

Задача 6. Анимация волн

Анимация волн

Задача 7. Анимация дождя

Анимация дождя

Задача 8. Игра "Пинг-Понг" (упрощенная)

Анимация прямолинейного движения шарика, отскакивающего от границ окна, причём угол падения равен углу отскока.

Приложение 5.

Материалы для промежуточного контроля в виде Экзамена в семестре 2.1.

ВОПРОСЫ ДЛЯ ПРОВЕРКИ ЗНАНИЙ

Компетенция ОПК-3: Способен управлять процессами создания и использования продуктов и услуг в сфере информационно-коммуникационных технологий, в том числе разрабатывать алгоритмы и программы для их практической реализации.

Знания: знать, как разрабатывать алгоритмы и программы, пригодные для практического применения.

1. Какой оператор используется для ветвления в Python?

- a) loop
- b) if
- c) switch
- d) select

2. Какой цикл выполняется, пока условие истинно?

- a) for
- b) while
- c) loop
- d) repeat

3. Как добавить элемент в список lst?

- a) lst.add(5)
- b) lst.append(5)
- c) lst.insert(5)
- d) lst.push(5)

4. Как получить длину списка lst?

- a) len(lst)
- b) lst.length()
- c) lst.size()
- d) length(lst)

5. Как создать кортеж из одного элемента?

- a) t = (1)
- b) t = (1,)
- c) t = [1]
- d) t = tuple(1)

6. Как проверить наличие ключа key в словаре d?

- a) `d.has_key(key)`
- b) `key in d`
- c) `d.exists(key)`
- d) `d.contains(key)`

7. Как удалить элемент с ключом key из словаря d?

- a) `d.remove(key)`
- b) `del d[key]`
- c) `d.pop(key)`
- d) `d.delete(key)`

8. Как объявить множество?

- a) `s = list()`
- b) `s = set()`
- c) `s = []`
- d) `s = ()`

9. Какой метод строки разбивает её по разделителю?

- a) `join()`
- b) `split()`
- c) `divide()`
- d) `partition()`

10. Какой модуль используется для регулярных выражений?

- a) `regex`
- b) `re`
- c) `string`
- d) `pattern`

11. Какой тип данных неизменяем?

- a) Список
- b) Кортеж
- c) Словарь
- d) Множество

12. Как получить последний элемент списка lst?

- a) `lst[last]`
- b) `lst[-1]`
- c) `lst[len(lst)]`
- d) `lst.get(-1)`

13. Как отсортировать список lst по убыванию?

- a) `lst.sort()`
- b) `lst.sort(reverse=True)`
- c) `sorted(lst)`
- d) `lst.reverse()`

14. Как получить список ключей словаря d?

- a) d.keys()
- b) list(d.keys())
- c) d.items()
- d) d.values()

15. Как удалить дубликаты из списка lst?

- a) lst.unique()
- b) list(set(lst))
- c) lst.distinct()
- d) remove_duplicates(lst)

16. Какой метод добавляет элементы в конец списка?

- a) insert()
- b) extend()
- c) append()
- d) add()

17. Как получить индекс элемента x в списке lst?

- a) lst.find(x)
- b) lst.index(x)
- c) index(x, lst)
- d) lst.position(x)

18. Как перебрать пары ключ-значение в словаре?

- a) for key in d:
- b) for key, value in d.items():
- c) for pair in d:
- d) for value in d.values():

19. Как преобразовать строку s в нижний регистр?

- a) s.toLower()
- b) s.lower()
- c) s.casefold()
- d) s.downcase()

20. Как добавить элемент в множество s?

- a) s.append(x)
- b) s.add(x)
- c) s.insert(x)
- d) s.update(x)

21. Какой метод строки удаляет пробелы с обоих концов?

- a) trim()
- b) strip()
- c) clean()
- d) cut()

22. Какой оператор возвращает значение из функции?

- a) yield
- b) return
- c) break
- d) exit

23. Какой тип данных используется для хранения пар ключ-значение?

- a) Список
- b) Словарь
- c) Множество
- d) Кортеж

24. Как получить список значений словаря d?

- a) d.keys()
- b) d.values()
- c) d.items()
- d) d.get_values()

25. Какой тип данных неупорядочен?

- a) Список
- b) Кортеж
- c) Множество
- d) Строка

26. Какой режим открытия файла используется для записи с удалением содержимого?

- a) 'a'
- b) 'w'
- c) 'r+'
- d) 'x'

27. Как правильно открыть файл для автоматического закрытия?

- a) file = open("data.txt")
- b) with open("data.txt") as file:
- c) open("data.txt").close()
- d) using open("data.txt")

28. Как прочитать все содержимое файла в строку?

- a) file.read(100)
- b) file.read()
- c) file.readlines()
- d) file.readline()

29. Как записать список строк в файл?

- a) file.save(lines)
- b) file.writelines(lines)
- c) file.write(lines)
- d) file.append(lines)

30. Как подключиться к SQLite базе данных?

- a) import mysql.connector
- b) import sqlite3
- c) import database
- d) import psycopg2

31. Какой SQL-запрос создает таблицу?

- a) MAKE TABLE users
- b) CREATE TABLE users (id INTEGER)
- c) NEW TABLE users
- d) TABLE CREATE users

32. Как вставить данные в таблицу?

- a) INSERT TO users VALUES (1)
- b) INSERT INTO users VALUES (1)
- c) ADD TO users VALUES (1)
- d) UPDATE users SET id = 1

33. Как выбрать все данные из таблицы?

- a) GET * FROM users
- b) SELECT * FROM users
- c) FIND * IN users
- d) READ * FROM users

34. Какой метод выполняет SQL-запрос?

- a) cursor.run()
- b) cursor.execute()
- c) cursor.do()
- d) cursor.query()

35. Как получить все результаты запроса?

- a) cursor.get()
- b) cursor.fetchall()
- c) cursor.results()
- d) cursor.all()

36. Какой метод файла читает одну строку?

- a) file.read()
- b) file.readline()
- c) file.getline()
- d) file.line()

37. Какой режим открывает файл для чтения и записи?

- a) 'rw'
- b) 'r+'
- c) 'a+'
- d) 'w+'

38. Как переместить указатель в файле?

- a) file.move()
- b) file.seek()
- c) file.position()
- d) file.goto()

39. Как обновить данные в таблице?

- a) CHANGE users SET name='John'
- b) UPDATE users SET name='John'
- c) MODIFY users SET name='John'
- d) ALTER users SET name='John'

40. Как удалить данные из таблицы?

- a) REMOVE FROM users WHERE id=1
- b) DELETE FROM users WHERE id=1
- c) DROP FROM users WHERE id=1
- d) CLEAR users WHERE id=1

41. Как создать индекс в SQL?

- a) MAKE INDEX idx_name ON users(name)
- b) CREATE INDEX idx_name ON users(name)
- c) ADD INDEX idx_name FOR users(name)
- d) INDEX idx_name ON users(name)

42. Как получить количество записей в таблице?

- a) COUNT * FROM users
- b) SELECT COUNT(*) FROM users
- c) SUM(*) FROM users
- d) LEN(*) FROM users

43. Какой оператор группирует данные?

- a) ORDER BY
- b) GROUP BY
- c) SORT BY
- d) COLLECT BY

44. Какой оператор фильтрует группы?

- a) WHERE
- b) HAVING
- c) FILTER
- d) GROUP WHERE

45. Какой метод возвращает одну строку результата?

- a) cursor.fetch()
- b) cursor.fetchone()
- c) cursor.getone()
- d) cursor.first()

46. Какой метод используется для массовой вставки?

- a) cursor.insertmany()
- b) cursor.executemany()
- c) cursor.bulk_insert()
- d) cursor.many()

47. Какой оператор SQL изменяет структуру таблицы?

- a) UPDATE
- b) ALTER
- c) MODIFY
- d) CHANGE

48. Как удалить таблицу?

- a) DELETE TABLE users
- b) DROP TABLE users
- c) REMOVE TABLE users
- d) CLEAR TABLE users

49. Как получить список таблиц в SQLite?

- a) SHOW TABLES
- b) SELECT name FROM sqlite_master WHERE type='table'
- c) LIST TABLES
- d) GET TABLES

50. Какой метод предотвращает SQL-инъекции?

- a) Конкатенация строк
- b) Использование параметризованных запросов
- c) Экранирование символов
- d) Все варианты верны

51. Как создать главное окно в Tkinter?

- a) window = Window()
- b) root = Tk()
- c) main = MainWindow()
- d) app = Application()

52. Какой метод запускает главный цикл обработки событий?

- a) start()
- b) mainloop()
- c) run()
- d) execute()

53. Как добавить кнопку в окно?

- a) addButton(text="Click")
- b) Button(root, text="Click").pack()
- c) new Button(root, "Click")
- d) create_button("Click")

54. Какой метод размещает виджеты в таблицу?

- a) place()
- b) grid()
- c) table()
- d) pack()

55. Как обработать нажатие кнопки?

- a) action
- b) command
- c) onclick
- d) function

56. Как создать текстовую метку?

- a) TextLabel(root, "Hello")
- b) Label(root, text="Hello")
- c) new Label(root, "Hello")
- d) Label.create("Hello")

57. Как получить текст из Entry?

- a) entry.text()
- b) entry.get()
- c) entry.value()
- d) entry.content()

58. Какой виджет используется для многострочного текста?

- a) Entry
- b) Text
- c) Label
- d) Input

59. Как изменить текст Label после создания?

- a) label.text = "New"
- b) label.config(text="New")
- c) label.set("New")
- d) label.change_text("New")

60. Как добавить пункты в Listbox?

- a) listbox.add("Item")
- b) listbox.insert(END, "Item")
- c) listbox.append("Item")
- d) listbox.items("Item")

61. Как обработать событие мыши?

- a) onclick
- b) bind
- c) attach
- d) listen

62. Как создать диалоговое окно сообщения?

- a) `Dialog.show("Message")`
- b) `messagebox.showinfo("Title", "Message")`
- c) `Popup.message("Message")`
- d) `Alert("Message")`

63. Как изменить цвет фона виджета?

- a) `widget.color("red")`
- b) `widget.config(bg="red")`
- c) `widget.background("red")`
- d) `widget.set_bg("red")`

64. Как добавить изображение в интерфейс?

- a) `ImageWidget()`
- b) `Label(image=photo)`
- c) `Picture()`
- d) `PhotoViewer()`

65. Как загрузить изображение для Tkinter?

- a) `Image.open()`
- b) `PhotoImage(file="img.png")`
- c) `TkImage.load()`
- d) `Picture.load()`

66. Как создать собственный модуль?

- a) Создать файл `.module`
- b) Создать файл `.py` с функциями
- c) Использовать `new_module()`
- d) Только через `setup.py`

67. Как импортировать функцию из модуля?

- a) `include module.func`
- b) `from module import func`
- c) `require module.func`
- d) `using module.func`

68. Как организовать код большого приложения?

- a) Все в одном файле
- b) Разделить на модули/пакеты
- c) Использовать только классы
- d) Автоматически генерировать

69. Как обработать закрытие окна?

- a) `on_close`
- b) `protocol("WM_DELETE_WINDOW")`
- c) `close_event`
- d) `window_close`

70. Как изменить размер окна?

- a) root.resize(300,200)
- b) root.geometry("300x200")
- c) root.size(300,200)
- d) root.set_size(300,200)

71. Как сделать окно не изменяемым?

- a) root.fixed()
- b) root.resizable(False, False)
- c) root.lock_size()
- d) root.set_resizable(0)

72. Как центрировать окно на экране?

- a) root.center()
- b) Рассчитать через wininfo_screenwidth()
- c) root.position("center")
- d) Автоматически

73. Как получить размер виджета?

- a) widget.size()
- b) widget.wininfo_width()
- c) widget.get_size()
- d) widget.measure()

74. Как скрыть/показать виджет?

- a) widget.hide()/widget.show()
- b) pack_forget()/pack()
- c) widget.visible(False)
- d) widget.set_visible()

75. Как обработать правую кнопку мыши?

- a) < Button-1 >
- b) < Button-2 >
- c) < Button-3 >
- d) < Button-4 >

76. Какой метод Tkinter используется для создания анимации?

- a) animate()
- b) after()
- c) loop()
- d) update()

77. Как переместить объект на Canvas?

- a) canvas.set_position()
- b) canvas.move()
- c) canvas.relocate()
- d) canvas.shift()

78. Какой параметр after() задает время до выполнения функции (в мс)?

- a) interval
- b) delay
- c) time
- d) duration

79. Как проверить столкновение двух объектов на Canvas?

- a) canvas.collission()
- b) Сравнение координат через coords()
- c) objects_overlap()
- d) check_collision()

80. Как обработать нажатие клавиш в игре?

- a) key_press()
- b) bind("<Key>", handler)
- c) on_key_press()
- d) input_handler()

81. Как сделать движение объекта при зажатой клавише?

- a) Использовать while key_pressed
- b) Обработать события <KeyPress> и <KeyRelease>
- c) Проверять состояние клавиатуры каждый кадр
- d) Через key.ispressed()

82. Как реализовать гравитацию в игре?

- a) object.gravity = True
- b) Увеличивать скорость по Y каждый кадр
- c) canvas.add_gravity()
- d) Использовать физический движок

83. Как загрузить изображение для спрайта?

- a) open("image.png")
- b) PhotoImage(file="image.png")
- c) load_image()
- d) Image.get()

84. Как сделать анимацию спрайта?

- a) sprite.animate()
- b) Менять изображение через itemconfig()
- c) Использовать gif
- d) Canvas.animation()

85. Как сохранить рекорд игры?

- a) В оперативной памяти
- b) В файл (JSON/txt)
- c) В облако
- d) В реестр Windows

86. Как сделать плавное движение объекта?

- a) Увеличить скорость
- b) Уменьшить интервал `after()`
- c) Использовать `threading`
- d) `canvas.smooth_move()`

87. Как проверить выход за границы экрана?

- a) `canvas.check_borders()`
- b) Сравнить координаты с размерами `Canvas`
- c) `object.out_of_bounds()`
- d) Автоматически в `Tkinter`

88. Как сделать "камеру", следящую за игроком?

- a) `Camera.follow()`
- b) Сместить все объекты в противоположную сторону
- c) `canvas.move_view()`
- d) Использовать `Scrollbar`

89. Как сделать взрыв/анимацию частиц?

- a) `ParticleSystem()`
- b) Несколько анимированных объектов
- c) `canvas.explosion()`
- d) Специальный модуль

90. Как добавить тень объекту?

- a) `object.shadow()`
- b) Дополнительный полупрозрачный объект
- c) `Canvas.effect()`
- d) Через стили `CSS`

91. Как сделать плавное появление/исчезновение?

- a) `widget.fade()`
- b) Менять прозрачность через `itemconfig`
- c) `alpha_channel()`
- d) `Canvas.transition()`

92. Как реализовать физику отскоков?

- a) `object.bounce()`
- b) Изменять направление скорости при столкновении
- c) Физический движок
- d) `Canvas.collision_response()`

93. Как сделать игру полноэкранной?

- a) `root.fullscreen()`
- b) `root.attributes("-fullscreen", True)`
- c) `root.maximize()`
- d) Установить размер как у экрана

94. Как ускорить рендеринг многих объектов?

- a) Уменьшить FPS
- b) Использовать tag для групповых операций
- c) Рисовать на PIL.Image
- d) Отключить анимации

95. Как добавить тень тексту?

- a) text.shadow()
- b) Вывести тот же текст со смещением и другим цветом
- c) Специальный параметр shadow
- d) Через CSS-стили

96. Как сделать анимацию движения по кривой?

- a) move_along_curve()
- b) Рассчитывать позицию по формуле кривой
- c) Специальный модуль
- d) Canvas.curve_motion()

97. Как добавить "дыхание" объекту (пульсацию)?

- a) object.pulse()
- b) Циклически менять размер
- c) Эффект встроен в Canvas
- d) Через CSS-анимации

98. Как добавить тряску камеры?

- a) Camera.shake()
- b) Случайно смещать все объекты
- c) Встроенный эффект
- d) Анимировать окно

99. Как реализовать систему настроек?

- a) Settings.load()
- b) Хранить в файле и загружать при старте
- c) Реестр Windows
- d) Глобальные переменные

100. Как добавить "заморозку" при ударе?

- a) time.stop()
- b) Пропустить несколько кадров анимации
- c) canvas.freeze()
- d) Специальный эффект

ЗАДАЧИ ДЛЯ ПРОВЕРКИ УМЕНИЙ

Компетенция ОПК-3: Способен управлять процессами создания и использования продуктов и услуг в сфере информационно-коммуникационных технологий, в том числе разрабатывать алгоритмы и программы для их практической реализации.

Умения: уметь разрабатывать алгоритмы и программы, пригодные для практического применения.

Задача 1. Слияние интервалов

Дан список интервалов, некоторые из которых пересекаются. Требуется получить новый список, в котором пересекающиеся интервалы будут объединены.

Например: `[[1,3],[2,6],[8,10],[15,18]]` \Rightarrow `[[1,6],[8,10],[15,18]]`

Задача 2. Проверка валидности скобок с учетом вложенных типов

Дана строка, в которой расставлены скобки разных видов `()`, `[]`, `{}`. Требуется определить верно ли расставлены скобки.

Например: `"([{}])"` \Rightarrow `True` или `"([{}])"` \Rightarrow `False`

Задача 3. Поиск всех анаграмм в списке слов

Дан список слов. Требуется получить новый список, где слова-анаграммы сгруппированы вместе. Например:

`["eat","tea","tan","ate","nat","bat"]` \Rightarrow `["eat","tea","ate"], ["tan","nat"], ["bat"]`

Задача 4. Поиск k-го наибольшего элемента в массиве

Дан массив натуральных чисел и номер `k`. Требуется найти `k`-й наибольший элемент без полной сортировки массива.

Например: `[3,2,1,5,6,4]`, `2` \Rightarrow `5`

Задача 5. Подсчет островов на матрице

Дана матрица заполненная нулями и единицами. Требуется найти количество "островов" из 1 в матрице.

```
[ ['1','1','0','0','0'],  
  ['1','1','0','0','0'],  
  ['0','0','1','0','0'],  
  ['0','0','0','1','1']]  =>  3
```

Задача 6. Поиск минимального окна подстроки

Дана длинная строка `s` и короткая строка `t`. Требуется найти в строке `s` минимальную подстроку, содержащую все символы строки `t`.

Например: `"ADOBECODEBANC"`, `"ABC"` \Rightarrow `"BANC"`

Задача 7. Найти все уникальные тройки чисел, дающих сумму 0

Дан массив произвольных целых чисел и число k . Требуется получить массив различных троек чисел, составленных из чисел заданного массива, сумма которых равна числу k .

Например: $[-1, 0, 1, 2, -1, -4], 0 \Rightarrow [[-1, -1, 2], [-1, 0, 1]]$

Задача 8. Поиск и замена слова в файле

Напишите программу, которая заменяет все вхождения слова "old" на "new" в файле text.txt и сохраняет результат в new_text.txt.

Задача 9. Объединение двух файлов

Напишите программу, которая объединяет содержимое файлов file1.txt и file2.txt и сохраняет результат в merged.txt.

Задача 10. Подсчёт гласных в файле

Напишите программу, которая считает количество гласных букв (a, e, i, o, u) в файле poem.txt.

Задача 11. Генератор CSV с случайными данными

Напишите программу, которая создаёт CSV-файл random_data.csv с 10 случайными числами (от 1 до 100) в одном столбце.

Задача 12. Добавление данных в SQLite

Напишите программу, которая добавляет 3 записи ((1, 'Alex'), (2, 'Kate'), (3, 'John')) в таблицу users в базе app.db.

Задача 13. Обновление данных в SQLite

Напишите программу, которая обновляет имя пользователя с $id = 1$ на "Alice" в таблице users базы app.db.

Задача 14. Экспорт SQLite в CSV

Напишите программу, которая экспортирует данные из таблицы users базы app.db в файл users.csv.

Задача 15. Кнопка с изменяемым текстом

Создать кнопку, которая меняет текст при нажатии.

Задача 16. Поле ввода с отображением текста

Выводить текст из Entry в Label при нажатии кнопки.

Задача 17. Переключатель темы

Переключать между светлой и темной темой при нажатии кнопки

Задача 18. Простой счетчик кликов

Создать кнопку, которая считает и отображает количество нажатий.

Задача 19. Мини-калькулятор (сложение)

Сложить два числа из Entry и вывести результат.

Задача 20. Флажок (Checkbutton)

Создать флажок для показа/скрытия текста.

Задача 21. Радиокнопки (Radiobutton)

Выбор размера текста с помощью радиокнопок.

Задача 22. Прыгающий мяч

Анимация прыгающего мяча

Задача 23. Кликер-игра

Считать клики по кнопке за 10 секунд

Задача 24. Игра "Поймай шарик"

Кликать по появляющимся шарикам

Задача 25. Анимация радуги

Плавно менять цвет фона

Задача 26. Простая анимация загрузки

Анимировать вращающийся круг

Задача 27. Игра "Убеги от курсора"

Квадрат убегает от курсора мыши

Задача 28. Игра "Найди пару" (упрощенная)

Открывать две карточки с числами

ЗАДАЧИ ДЛЯ ПРОВЕРКИ НАВЫКОВ

Компетенция ОПК-3: Способен управлять процессами создания и использования продуктов и услуг в сфере информационно-коммуникационных технологий, в том числе разрабатывать алгоритмы и программы для их практической реализации.

Навыки: владеть навыками разработки алгоритмов и программ, пригодных для практического применения.

Задача 1. Найти пропущенное число

Дан массив натуральных чисел от 0 до n с одним пропущенным числом. Требуется найти его.

Например: [3, 0, 1] \Rightarrow 2

Задача 2. Проверить, можно ли преобразовать строку в другую строку сдвигом букв

Даны две строки. Требуется проверить, можно ли получить вторую строку из первой циклическим сдвигом.

Например: "abcde", "cdeab" \Rightarrow True или "abc", "acb" \Rightarrow False

Задача 3. Найти все числа, исчезнувшие в массиве

Дан список чисел от 1 до n , некоторые числа пропущены. Найти все пропущенные числа.

Например: [4, 3, 2, 7, 8, 2, 3, 1] \Rightarrow [5, 6]

Задача 4. Проверить, является ли число палиндромом

Дано натуральное число. Требуется проверить, является ли целое число палиндромом без преобразования в строку. Число-палиндром – это число, которое читается одинаково слева направо и справа налево.

Например: 121 \Rightarrow True или 123 \Rightarrow False

Задача 5. Найти ближайший меньший элемент для каждого элемента массива

Дан массив натуральных чисел. Требуется получить новый массив, где каждый элемент заданного массива заменён на ближайший слева элемент этого массива, который меньше заменяемого элемента, или на -1, если такого элемента нет.

Например: [4, 5, 2, 10, 8] \Rightarrow [-1, 4, -1, 2, 2]

Задача 6. Найти все возможные перестановки массива

Дан массив натуральных чисел. Требуется получить массив всех возможных перестановок элементов этого массива.

Например: [1, 2, 3] \Rightarrow [[1, 2, 3], [1, 3, 2], [2, 1, 3], [2, 3, 1], [3, 2, 1], [3, 1, 2]]

Задача 7. Найти подмассив с максимальным произведением

Дан массив с произвольными целыми числами. Требуется найти подмассив, произведение элементов которого максимально. Подмассив может содержать любое количество подряд идущих чисел заданного массива.

Например: [2, 3, -2, 4] => [2, 3]

Задача 8. Найти все возможные подмножества массива

Дан массив произвольных натуральных чисел. Требуется получить массив всех возможных подмножеств массива (без дубликатов).

Например: [1, 2, 3] => [[], [1], [2], [1, 2], [3], [1, 3], [2, 3], [1, 2, 3]]

Задача 9. Удаление пустых строк из файла

Напишите программу, которая удаляет все пустые строки из файла input.txt и сохраняет результат в output.txt.

Задача 10. Сортировка строк файла

Напишите программу, которая сортирует строки файла words.txt в алфавитном порядке и сохраняет результат в sorted_words.txt.

Задача 11. Поиск самой длинной строки в файле

Напишите программу, которая находит и выводит самую длинную строку в файле lines.txt.

Задача 12. Фильтрация данных в CSV

Напишите программу, которая читает data.csv и сохраняет в filtered.csv только строки, где значение во втором столбце больше 50.

Задача 13. Подсчёт частоты слов в файле

Напишите программу, которая подсчитывает, сколько раз каждое слово встречается в файле text.txt (без учёта регистра).

Задача 14. Удаление данных из SQLite

Напишите программу, которая удаляет пользователя с id = 2 из таблицы users в базе app.db.

Задача 15. Поиск пользователя в SQLite

Напишите программу, которая ищет пользователя по имени "Alice" в таблице users базы app.db и выводит его id.

Задача 16. Резервное копирование SQLite

Напишите программу, которая создаёт резервную копию базы app.db в файл app_backup.db.

Задача 17. Ползунок (Scale) для изменения размера

Менять размер текста с помощью ползунка.

Задача 18. Меню (Menu)

Создать выпадающее меню с действиями.

Задача 19. Диалоговое окно (messagebox)

Показать диалоговое окно с вопросом.

Задача 20. Случайное число

Генерировать случайное число от 1 до 100 при нажатии на кнопку.

Задача 21. Мини-блокнот

Простой текстовый редактор с сохранением записей в файл и чтением записей из файла.

Задача 22. Проверка пароля

Проверить, совпадают ли введенные пароли.

Задача 23. Мини-игра "Угадай число"

Угадать число от 1 до 10. Число вводится в Entry и проверяется при нажатии на кнопку.

Задача 24. Список дел (Listbox)

Добавлять и удалять задачи из списка Listbox.

Задача 25. Анимация взрыва

Анимация расходящихся кругов

Задача 26. Игра "Космический корабль"

Управлять кораблем клавишами

Задача 27. Анимация текста

Текст, меняющий размер и цвет

Задача 28. Бегущая строка

Анимация бегущей строки

Задача 29. Летающий НЛО

Анимация летающего НЛО

Задача 30. Анимация волн

Анимация волн

Задача 31. Анимация дождя

Анимация дождя

Задача 32. Игра "Пинг-Понг" (упрощенная)

Анимация прямолинейного движения шарика, отскакивающего от границ окна, причём угол падения равен углу отскока.

Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Система критериев оценки определяет оценку успеваемости по каждому заданию (вопросу) экзаменационного билета или заданию для зачета с использованием интервальной шкалы баллов, применяемой в привязке к рейтинговой 100-балльной системе.

ОЦЕНКА ОТВЕТА НА ТЕОРЕТИЧЕСКИЙ ВОПРОС В УСТНОЙ ИЛИ ПИСЬМЕННОЙ ФОРМЕ:

Оценка «отлично» / «зачтено» (91-100 баллов) выставляется при соблюдении следующих условий: Ответ отличается глубиной и полнотой, свободным владением понятийно-категориальным (терминологическим) аппаратом изученной дисциплины. Отражает знание не только основной, но и дополнительной литературы. Приведены примеры, отражающие умение связать теорию с практикой. Ответ изложен логически последовательно, грамотно и корректно.

Оценка «хорошо» / «зачтено» (76-90 баллов) выставляется при соблюдении следующих условий: Ответ отличается полнотой, владением понятийно-категориальным (терминологическим) аппаратом изученной дисциплины, но в ответе могут присутствовать неточности. Отражает знание основной литературы. Приведены примеры, отражающие умение связать теорию с практикой. Ответ изложен логически последовательно, грамотно и корректно, но недостаточно аргументирован.

Оценка «удовлетворительно» / «зачтено» (61-75 баллов) выставляется при соблюдении следующих условий: В ответе отражено знание понятийно-категориального (терминологического) аппарата изучаемой дисциплины, но присутствуют отдельные ошибки и неточности. Ответ характеризуется недостаточным знанием рекомендованной литературы. Примеры, отражающие умение связать теорию с практикой, тривиальны, либо отсутствуют. Ответ неполный, носит фрагментарный, непоследовательный характер.

Оценка «неудовлетворительно» / «не зачтено» (0-60 баллов) выставляется при соблюдении следующих условий: Ответ характеризуется незнанием, либо фрагментарным представлением о понятийно-категориальном аппарате дисциплины, содержит множество ошибок. Примеры и иллюстрации отсутствуют. Ответ логически непоследователен.

ОЦЕНКА ВЫПОЛНЕНИЯ ЗАДАНИЯ В ФОРМЕ CASE-STUDY (СИТУАЦИИ)

Оценка «отлично» / «зачтено» (91-100 баллов) выставляется при соблюдении следующих условий: Четкая формулировка проблемы. Полное и соответствующее ситуации решение, основанное на знании правовых норм и технологий (опыте), применяемых в реальных организациях (известных компаниях). Предполагаемые действия описаны логично и последовательно. Даны дополнительные авторские комментарии и предложения к решению ситуации.

Оценка «хорошо» / «зачтено» (76-90 баллов) выставляется при соблюдении следующих условий: Понимание сути проблемы, но ее формулирование затруднено. Решение соответствует ситуации, отражает знание правовых норм и опыт работы других организаций при решении подобных ситуаций. Логика и последовательность действий не нарушены.

Оценка «удовлетворительно» / «зачтено» (61-75 баллов) выставляется при соблюдении следующих условий: Проблема не сформулирована. Приведен набор действий, потенциально способствующих улучшению ситуации и решению проблемы.

Оценка «неудовлетворительно» / «не зачтено» (0-60 баллов) выставляется при

соблюдении следующих условий: Предложенный перечень мероприятий не соответствует ситуации.

ОЦЕНКА РЕШЕНИЯ ЗАДАЧИ

Оценка «отлично» / «зачтено» (91-100 баллов) выставляется при соблюдении следующих условий: Полное верное решение - оценивается в n баллов (n – максимальное количество баллов за решение задачи в структуре экзаменационного билета/задания).

Оценка «хорошо» / «зачтено» (76-90 баллов) выставляется при соблюдении следующих условий: Верное решение; имеются небольшие недочеты, в целом не влияющие на решение – оценивается в диапазоне от $0,76*n$ баллов до $0,9*n$ баллов (n – максимальное количество баллов за решение задачи в структуре экзаменационного билета/задания).

Оценка «удовлетворительно» / «зачтено» (61-75 баллов) выставляется при соблюдении следующих условий: Решение в целом верное; однако оно содержит ряд ошибок, либо не учитывает отдельных случаев, но может стать правильным после некоторых исправлений или дополнений – оценивается в диапазоне от $0,61*n$ баллов до $0,75*n$ баллов (n – максимальное количество баллов за решение задачи в структуре экзаменационного билета/задания).

Оценка «неудовлетворительно» / «не зачтено» (0-60 баллов) выставляется при соблюдении следующих условий: Решение неверное; изначально выбран неверный ход решения, или решение отсутствует – оценивается в 0 баллов.

ОЦЕНКА ВЫПОЛНЕНИЯ ТЕСТОВОГО ЗАДАНИЯ

Подсчитывается доля набранных баллов в максимальной сумме баллов за все задания теста:

- Каждый правильный ответ на тестовый вопрос (тип выборочный, одинарный, множественный, открытый) оценивается в m баллов (число m определяется путем деления максимального количества баллов за выполнение теста в структуре экзаменационного билета/задания на количество тестовых заданий);
- Каждый частично правильный ответ на тестовый вопрос (тип выборочный, множественный, открытый) оценивается в $m/2$ баллов независимо от соотношения правильно/неправильно выбранных вариантов (число m определяется путем деления максимального количества баллов за выполнение теста в структуре экзаменационного билета/задания на количество тестовых заданий);
- Каждый неправильный ответ на тестовый вопрос (тип выборочный, одинарный) оценивается в 0 баллов.

Оценка «отлично»/ «зачтено» (91-100 баллов) выставляется, если доля набранных баллов составляет 91-100%.

Оценка «хорошо»/ «зачтено» (76-90 баллов), если доля набранных баллов составляет 76-90%.

Оценка «удовлетворительно»/ «зачтено» (61-75 баллов), если доля набранных баллов составляет 61-75%.

Оценка «неудовлетворительно»/ «не зачтено» (0-60 баллов), если доля набранных баллов составляет не более 60%.