Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Выборнова Любовь Алексеевна
Должность: Ректор
Дата подписания: 30.10.2026 08:54:24
Уникальный программный ключ:
c3b3b9c625f6c113afa2a2c42baff9e05a38b76e

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение высшего образования «Поволжский государственный университет сервиса» (ФГБОУ ВО «ПВГУС»)

Колледж креативных индустрий и предпринимательства

УТВЕРЖДЕНО
на заседании ПЦК

по образовательной программе 09.02.07 Информационные системы и программирование

Протокол от 24.10.2025 г. № 2

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ
(фонд оценочных средств) для проведения промежуточной аттестации

по дисциплине
учебному предмету, дисциплине, междисциплинарному курсу, профессиональному модулю

«Основы алгоритмизации и программирования»
наименование учебного предмета, дисциплины, междисциплинарного курса, профессионального модуля

по образовательной программе среднего профессионального образования –

программе подготовки специалистов среднего звена
программе подготовки специалистов среднего звена / программе подготовки квалифицированных рабочих, служащих

Информационные системы и программирование
наименование образовательной программы

09.02.07 Информационные системы и программирование
шифр, наименование специальности / профессии

Составитель Ляпина К.В., преподаватель Колледжа креативных индустрий и предпринимательства
ФИО, должность, структурное подразделение, ученая степень, ученое звание

Тольятти
2025

1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ (ДАЛЕЕ – ФОС)

1.1. Планируемые результаты обучения по дисциплине

Код компетенции	Наименование компетенции
ОК 02	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности
ПК 3.1	Осуществлять ревьюирование программного кода в соответствии с технической документацией
ПК 3.3	Производить исследование созданного программного кода с использованием специализированных программных средств с целью выявления ошибок и отклонения от алгоритма

В результате освоения дисциплины обучающийся должен:

иметь практический опыт:

-проведение ревью программного кода с сопоставлением с требованиями технической документации;

- выявление несоответствий и ошибок в коде относительно технических спецификаций, взаимодействие с разработчиками для обсуждения и исправления замечаний по коду.

-проведение анализа программного кода с применением специализированных инструментов. выявление ошибок и несоответствий алгоритмической логике в коде. подготовка отчетов о найденных ошибках и взаимодействие с разработчиками для их устранения.

-оптимизировать программный код с использованием специализированных программных средств;

- использовать основные методологии процессов разработки программного обеспечения

уметь:

-определять задачи для поиска информации;

-определять необходимые источники информации;

-анализировать программный код на соответствие технической документации;

-формулировать конструктивные замечания и предложения по улучшению кода;

-применять методы и инструменты для проведения эффективного ревью кода;

-использовать специализированные программные средства для анализа и отладки кода. идентифицировать ошибки и отклонения от заданного алгоритма;

-формулировать рекомендации по исправлению и оптимизации программного кода.

-использовать современное программное обеспечение; использовать различные цифровые средства для решения профессиональных задач.

-выполнять оптимизацию программного кода с использованием специализированных программных средств;

- использовать методы и технологии тестирования и ревьюирования кода и проектной документации.

знать:

-номенклатуру информационных источников, применяемых в профессиональной деятельности;

-приемы структурирования информации;

-формат оформления результатов поиска информации;

-структура и содержание технической документации по программному обеспечению;

-принципы и цели ревьюирования кода;

- принципы работы специализированных средств анализа и отладки программного обеспечения;

основы алгоритмизации и структуры программного кода, типы ошибок и отклонений, возникающих в процессе разработки ПО;

-основные методы и инструменты ревьюирования программного кода;

-современные средства и устройства информатизации; порядок их применения и программное обеспечение в профессиональной деятельности, в том числе с использованием цифровых средств.

1.2. Содержание учебной дисциплины

№	Тема (раздел дисциплины) (в соответствии с РПД)	Код компетенции
1	Основы алгоритмизации	ОК 2, ПК 3.1, ПК 3.3
2	Структуры данных	ОК 2, ПК 3.1, ПК 3.3
3	Алгоритмы сортировки	ОК 2, ПК 3.1, ПК 3.3
4	Алгоритмы поиска	ОК 2, ПК 3.1, ПК 3.3
5	Введение в Python	ОК 2, ПК 3.1, ПК 3.3
6	Типы данных и переменные	ОК 2, ПК 3.1, ПК 3.3
7	Условный оператор	ОК 2, ПК 3.1, ПК 3.3
8	Циклические операторы	ПК 3.1, ПК 3.3
9	Функции	ПК 3.1, ПК 3.3
10	Списки и кортежи	ПК 3.1, ПК 3.3

11	Словари и множества	ПК 3.1, ПК 3.3
12	Строки	ПК 3.1, ПК 3.3
13	Модули и пакеты	ПК 3.1, ПК 3.3
14	Работа с файлами	ПК 3.1, ПК 3.3

1.3. Система оценивания по учебной дисциплине

Курс изучается в течение двух семестров. Форма промежуточной аттестации по дисциплине – контрольная работа (3 семестр), дифференцированный зачёт (4 семестр).

Шкалы оценки уровня сформированности результатов обучения		Шкала оценки уровня освоения дисциплины	
Уровневая шкала оценки компетенций	100 бальная шкала, %	100 бальная шкала, %	недифференцированная оценка
допороговый	ниже 61	ниже 61	не зачтено
пороговый	61-85,9	61-69,9	удовлетворительно
		70-85,9	хорошо
повышенный	86-100	86-100	отлично

2. ПЕРЕЧЕНЬ ОЦЕНОЧНЫХ МАТЕРИАЛОВ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

В ходе проведения промежуточной аттестации осуществляется контроль и оценка результатов освоения профессиональных компетенций и их частей.

Вопросы для подготовки к контрольной работе

№	Содержание вопроса
	3 семестр
	ОК 02 Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности
1.	Что такое алгоритм? A) Это последовательность шагов для решения задачи B) Это набор правил программирования C) Это план действий для выполнения работы D) Все вышеперечисленное верно
2.	Какие основные свойства алгоритма? A) Дискретность, результативность, массовость B) Конечность, определенность, массовость C) Детерминированность, конечность, массовость D) Массовость, конечность, универсальность
3.	Какой тип алгоритмической структуры используется для последовательного выполнения операций? A) Линейный алгоритм B) Ветвление C) Цикл D) Рекурсия
4.	Какой из перечисленных ниже алгоритмов сортировки является наиболее эффективным по времени исполнения для почти отсортированных данных? A) Пузырьковая сортировка

№	Содержание вопроса
	B) Быстрая сортировка C) Сортировка слиянием D) Сортировка вставками
5.	Какой из указанных алгоритмов сортировки обладает наилучшим временем работы в худшем случае? A) Быстрая сортировка B) Сортировка слиянием C) Сортировка выбора D) Пирамидальная сортировка
6.	Какой из алгоритмов сортировки основан на принципе "разделяй и властвуй"? A) Быстрая сортировка B) Сортировка кучей C) Сортировка слиянием D) Сортировка выбором
7.	Какой алгоритм поиска минимизирует количество сравнений при поиске в отсортированном массиве? A) Линейный поиск B) Бинарный поиск C) Поиск методом деления пополам D) Интерполяционный поиск
8.	Какой из алгоритмов поиска особенно эффективен при работе с большими объемами данных? A) Линейный поиск B) Бинарный поиск C) Jump search D) Depth-first search
9.	Какой алгоритм поиска находит элемент за минимальное время в массиве, отсортированном по возрастанию? A) Линейный поиск B) Бинарный поиск C) Interpolation search D) Fibonacci search
10.	Какие типы алгоритмических структур существуют?
11.	Назовите основные свойства алгоритма.
12.	Что такое алгоритм?
13.	Какой алгоритм сортировки является устойчивым?
14.	Какой алгоритм сортировки использует принцип "разделяй и властвуй"?
15.	Какой алгоритм сортировки самый эффективный для больших объемов данных?
16.	Каково основное свойство двоичного дерева поиска (BST)?
17.	Какой из алгоритмов поиска использует метод дихотомии (разбиения на части)?
18.	Какой из алгоритмов поиска относится к классу жадных алгоритмов?
ПК 3.1 Осуществлять ревьюирование программного кода в соответствии с технической документацией	
19.	Как в Python обозначаются комментарии? A) // B) # C) <!-- --> D) /* */
20.	Какой оператор используется для вывода текста на экран в Python? A) print()

№	Содержание вопроса
	B) echo() C) output() D) console.log()
21.	Как объявляется переменная в Python? A) С использованием ключевого слова var B) Просто присваивается значение C) С использованием символа $ D) С использованием ключевого слова let
22.	Как называется основной файл программы на Python? A) .exe B) .py C) .java D) .cpp
23.	Как выполняется сложение чисел в Python? A) add(5, 3) B) sum(5, 3) C) 5 + 3 D) concatenate(5, 3)
24.	Какой тип данных в Python используется для целых чисел? A) int B) float C) str D) bool
25.	Какая операция в Python возвращает остаток от деления? A) % B) / C) ** D) //
26.	Какой тип данных в Python используется для дробных чисел?
27.	Как называется динамическая типизация в Python?
28.	Какие основные типы данных есть в Python?
29.	Может ли одна переменная хранить различные типы данных в разное время?
30.	Как объявляются переменные в Python?
31.	Как запустить интерпретатор Python в интерактивном режиме?
32.	Как вывести сообщение "Привет, мир!" в Python?
33.	Как объявить переменную в Python?
34.	Как в Python создать комментарий?
ПК 3.3 Производить исследование созданного программного кода с использованием специализированных программных средств с целью выявления ошибок и отклонения от алгоритма	
35.	Как в Python записывается условный оператор if? A) if (условие): B) if условие: C) if {условие}: D) if [условие]:
36.	Как правильно записать условие "если число равно 5"? A) if x = 5: B) if x == 5: C) if x is 5: D) if x === 5:
37.	Как в Python добавить блок else, если условие не выполнено?

№	Содержание вопроса
	A) else if: B) elif: C) else: D) otherwise:
38.	Как записать условие "если число больше 0 и меньше 10"? A) if 0 < x and x < 10: B) if 0 > x or x > 10: C) if 0 <= x >= 10: D) if 0 && x || x > 10:
39.	Как выглядит запись условия "если число не равно 0"? A) if not x == 0: B) if x != 0: C) if !x == 0: D) if x <> 0:
40.	Как в Python записать условие "если число больше нуля"?
41.	Как указать действие, которое выполняется, если ни одно из условий не выполнено?
42.	Как добавить альтернативное условие в Python?
43.	Как соединить несколько условий в Python?
44.	Как в Python реализовать тернарный оператор?
Вопросы для подготовки к дифференцированному зачёту (4 семестр)	
ПК 3.1 Осуществлять ревьюирование программного кода в соответствии с технической документацией	
45.	Как в Python записать цикл for, который будет перебирать элементы списка? A) for i in range(len(list)): B) for element in list: C) foreach element in list: D) while element in list:
46.	Как в Python организовать бесконечный цикл? A) while True: B) loop forever: C) repeat until False: D) infinite loop:
47.	Как в Python определяется функция? A) def function_name(): B) function function_name(): C) func function_name(): D) define function_name():
48.	Что такое массив? A) Набор объектов, хранимых последовательно в памяти B) Неупорядоченная коллекция уникальных элементов C) Древовидная структура данных D) Списки элементов, объединенные связями
49.	Можно ли изменить элемент кортежа после его создания? A) Да B) Нет C) Только если кортеж пустой D) Только если кортеж содержит один элемент
50.	Как добавить новый элемент в словарь в Python? A) dict[key] = value

№	Содержание вопроса
	B) dict.add(key, value) C) dict.insert(key, value) D) dict.put(key, value)
51.	Как получить длину строки в Python? A) len(string) B) string.length() C) string.size() D) length(string)
52.	Как пропустить текущую итерацию цикла и перейти к следующей в Python?
53.	Как в Python записать цикл while, который будет выполняться до тех пор, пока условие истинно?
54.	Как вернуть значение из функции в Python?
55.	Как в Python объявляется функция?
56.	Как создать кортеж в Python?
57.	Как создать пустой список в Python?
58.	Как создать пустой словарь в Python?
59.	Как создать пустое множество в Python?
60.	Как извлечь подстроку из строки в Python?
61.	Как объединить две строки в Python?
62.	Как открыть файл для чтения в Python?
63.	Как записать данные в файл в Python?
ПК 3.3 Производить исследование созданного программного кода с использованием специализированных программных средств с целью выявления ошибок и отклонения от алгоритма	
64.	Как импортировать модуль в Python? A) import module B) load module C) require module D) module.load()
65.	Как импортировать конкретную функцию из модуля в Python? A) from module import function B) import module.function C) use module.function D) module.import(function)
66.	Как создать пакет в Python? A) Создать папку с файлом __init__.py внутри B) Создать файл с расширением .pkg C) Использовать команду python create_package D) Создать папку с файлом package.py внутри
67.	Как установить сторонний пакет в Python? A) pip install package_name B) python install package_name C) install package_name D) download package_name
68.	Как записать данные в файл в Python? A) file.write(data) B) write(file, data) C) save(file, data) D) put(file, data)
69.	Как прочитать содержимое файла построчно в Python? A) for line in file.readlines():

№	Содержание вопроса
	B) for line in file.read_lines(): C) for line in file.lines(): D) for line in file.line_by_line():
70.	Как организовать бесконечный цикл в Python?
71.	Как прервать выполнение цикла в Python?
72.	Как вызвать функцию в Python?
73.	Можно ли изменить элемент кортежа после его создания?
74.	Можно ли изменить элемент списка после его создания?
75.	Чем отличается словарь от множества в Python?
76.	Как проверить, содержится ли подстрока в строке в Python?