

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Сыров Игорь Анатольевич
Должность: Директор
Дата подписания: 20.05.2024 16:35:35
Уникальный программный ключ:
b683afe664d7e9f64175886cf9626a198149ad36

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«БАШКИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Стерлитамакский филиал

Колледж

Рабочая программа дисциплины

дисциплина

ОП.04 Основы алгоритмизации и программирования

Общепрофессиональный цикл, обязательная часть

цикл дисциплины и его часть (обязательная, вариативная)

специальность

09.02.07

Информационные системы и программирование

код

наименование специальности

квалификация

Администратор баз данных

Год начала подготовки

2022

Разработчик (составитель)

Кучер А.М.

ученая степень, ученое звание,
категория, Ф.И.О.

Стерлитамак 2022

ОГЛАВЛЕНИЕ

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ	3
1.1. Область применения рабочей программы	3
1.2. Место учебной дисциплины в структуре основной профессиональной образовательной программы	3
1.3. Цель и планируемые результаты освоения дисциплины:	3
2. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ	4
2.1 Объем дисциплины и виды учебной работы	4
2.2. Тематический план и содержание дисциплины	5
3. ФОРМЫ КОНТРОЛЯ И МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, ПРАКТИЧЕСКОГО ОПЫТА, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ.....	14
4. УСЛОВИЯ РЕАЛИЗАЦИИ ДИСЦИПЛИНЫ	14
4.1. Требования к минимальному материально-техническому обеспечению	14
4.2. Учебно-методическое и информационное обеспечение дисциплины (модуля)	Ошибка! Закладка не определена.
4.2.1. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)	Ошибка! Закладка не определена.
4.2.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет» (далее - сеть «Интернет»), необходимых для освоения дисциплины (модуля)	Ошибка! Закладка не определена.
4.3.3. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем (при необходимости)....	Ошибка! Закладка не определена.
ПРИЛОЖЕНИЕ № 1	16
ПРИЛОЖЕНИЕ 2	18

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ

1.1. Область применения рабочей программы

Рабочая программа дисциплины является частью основной образовательной программы в соответствии с ФГОС для специальности: 09.02.07 Информационные системы и программирование (укрупненная группа специальности 09.00.00 Информатика и вычислительная техника) для обучающихся очно-заочной формы обучения.

1.2. Место учебной дисциплины в структуре основной профессиональной образовательной программы

Дисциплина относится к общепрофессиональному циклу. Дисциплина реализуется в рамках обязательной части.

1.3. Цель и планируемые результаты освоения дисциплины:

Код ОК, ПК	Умения	Знания
ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам	Решать задачу на языке программирования на основе поставленной задачи предметной области.	Методов и подходов для решения задач профессиональной области с применением языков программирования.
ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности	Производить поиск, анализ и интерпретацию доступной информации с целью решения задач профессиональной деятельности.	Уметь работать с информацией: искать, аккумулировать, анализировать, перерабатывать для решения поставленной задачи.
ОК 04. Эффективно взаимодействовать и работать в коллективе и команде	Умение решать задачи в команде. Решение задач в формате проектной деятельности.	Решение задач на основе принципа декомпозиции с учетом разделения обязанностей в команде, распределения обязанностей внутри команды согласно навыкам и знаниям.
ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста.	Умение транслировать текстовую задачу в алгоритм и программу на языке программирования.	Принципы решения задач Выбор языка программирования для каждой конкретной задачи, использование соответствующих типов данных, методов языка.
ОК 09 Пользоваться профессиональной	Умение применять методы программирования для	Синтаксис и семантика языка программирования, основы

документацией на государственном и иностранном языках	решения задач профессиональной области	формирования языка программирования
ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения.	Уметь писать тестовые программы с целью проверки правильности работы программного кода.	Особенности написания тестовых программ с целью проверки правильности работы.
ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.	Уметь использовать набор правил и соглашений, которые используются для написания исходного кода с целью оптимизации по времени и памяти.	Знать основы написания программного кода с использованием набора правил и соглашений.

2. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

2.1 Объем дисциплины и виды учебной работы

<i>Вид учебной работы</i>	<i>Объем часов</i>
Объем образовательной программы	264
Работа обучающихся во взаимодействии с преподавателем	42
в том числе:	
лекции (уроки)	18
в форме практической подготовки (если предусмотрено)	*
практические занятия	24
в форме практической подготовки (если предусмотрено)	*
лабораторные занятия	-
в форме практической подготовки (если предусмотрено)	*
курсовая работа (проект) (если предусмотрена)	-
Самостоятельная работа обучающегося (всего) (если предусмотрена)	216
Консультации (если предусмотрена)	4
Промежуточная аттестация в форме итоговой контрольной работы в 1 и 2 семестрах В форме экзамена в 3 семестре	6

2.2. Тематический план и содержание дисциплины

Наименование разделов и тем	Содержание учебного материала	Объем часов	Осваиваемые элементы компетенций
1	2	3	4
Раздел № 1. Основные конструкции языка программирования Pascal		14/26	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 1.1 Вводная лекция. История развития программирования. Метаязык.	Самостоятельная работа 1. Первые языки программирования 2. Языки программирования высокого уровня 3. Классификация языков программирования 4. Метаязык	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 1.2 Начальные сведения о программах на языке Pascal	Самостоятельная работа 1. Алгоритм и программа 2. Комментарии 3. Идентификаторы 4. Переменные и типы данных 5. Константы 6. Простейшие операции 7. Метки и безусловный переход 8. Ввод и вывод: консоль	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 1.3 Типы данных и операции языка Pascal	Содержание учебного материала 1. Типы данных языка Pascal 2. Операции и выражения 3. Совместимость типов данных	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 1.4 Операторы ветвления	Содержание учебного материала 1. Условный оператор if 2. Оператор выбора case 3. Иллюстрация if и case	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 1.5 Операторы циклов	Содержание учебного материала 1. for-to и for-downto 2. while и repeat-until 3. break и continue 4. Примеры использования циклов	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 1.6 Понятие массива. Операции, используемые в массиве	Самостоятельная работа 1. Описание массива 2. Обращение к компонентам массива 3. Задание массива константой 4. Многомерные массивы	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 1.7	Самостоятельная работа 1. Описание строки	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК

Символы и строки	2. Символ-константа и строка константа 3. Действия с символами 4. Стандартные процедуры и функции обработки строк 5. Операции со строками		2.4., ПК 2.5
Тема 1.8 Множества	Самостоятельная работа 1. Описание множеств 2. Множество-константа 3. Операции с множествами 4. Представление множеств массивами	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 1.9 Условный оператор. Практикум решения задач	Практическое занятие 1. Полный условный оператор 2. Использование сложных условий 3. Оператор вариантов (выбора)	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 1.10 Оператор цикла с параметром	Практическое занятие 1. Организация ввода данных по требуемому формату 2. Обработка данных во время ввода 3. Рекуррентные соотношения	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 1.11 Оператор цикла с условием	Практическое занятие 1. Обработка числовых последовательностей 2. Использование условного оператора в теле оператора цикла с предусловием 3. Использование условного оператора в теле оператора цикла с постусловием	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 1.12 Ввод и вывод информации: текстовые файлы	Самостоятельная работа 1. Ввод и вывод: файлы 2. Описание файлов 3. Примеры использования файлов 4. Изменение реакции на ошибку	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 1.13 Записи. Бинарные файлы	Самостоятельная работа 1. Записи. 2. Типизированные файлы 3. Нетипизированные файлы 4. Подпрограммы обработки директорий	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 1.14 Процедуры и функции	Самостоятельная работа 1. Подпрограммы 2. Способы подстановки аргументов 3. Область действия имен 4. Нетипизированные параметры 5. Открытые параметры 6. Процедурный тип данных	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 1.15 Модульная структура программы	Самостоятельная работа 1. Модульность программ 2. Стандартные модули языка Pascal	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5

	3. Создание модульной программы 4. Передача аргументов из командной строки		
Тема 1.16 Методы создания пользовательского интерфейса	Самостоятельная работа 1. Текстовый режим 2. Создание дружественного интерфейса 3. Графический режим	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 1.17 Технология программирования и отладки	Самостоятельная работа 1. Советы по технологии написания быстро отлаживаемых программ 2. Отладка и тестирование 3. Оптимизация программ	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 1.18 Построение графика функции	Самостоятельная работа 1. Работа с графическим редактором в Pascal 2. Построение графиков функций	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 1.19 Построение и редактирование ломаной	Самостоятельная работа 1. Программа графического ввода и редактирования ломаной 2. Выделение самого краткого отрезка ломаной	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 1.20 Итоговая контрольная работа	Итоговая контрольная работа Подведение итогов за семестр	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Раздел 2. Основы Python		14/26	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 2.1 Введение в Python	Самостоятельная работа 1. Введение в идеологию Python 2. Синтаксис и первый код 3. Переменные, ввод/вывод строк exceptions (ошибки и исключения)	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 2.2 Типы данных – числа	Содержание учебного материала 1. Виды чисел и ошибки при работе с ними 2. Порядок операций 3. Обработка вещественных чисел в Python	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 2.3 Условные операторы	Содержание учебного материала 1. Логические выражения (bool expr) 2. Условная инструкция if 3. Оператор if и секция else 4. Вложенные if и elif	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 2.4 Понятие функции в Python	Содержание учебного материала 1. Общая информация 2. Объявление функции 3. Результат функции 4. Аргументы функции	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 2.5 Строки – тип данных str	Самостоятельная работа 1. Стандартный ввод 2. Экранирование символов	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5

	3. Символы и кодировка 4. Мутабельность строк		
Тема 2.6 Методы строк	Самостоятельная работа 1. Методы строки, возвращающие логические значения True/False 2. Методы, возвращающие числа 3. Методы, создающие новые строки	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 2.7 Линейные программы	Практическое занятие 1. Оформление программ 2. Написание линейных программ	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 2.8 Применение условного оператора при решении задач	Практическое занятие 1. Неполное ветвление 2. Полное ветвление 3. Множественное ветвление 4. Вложенные конструкции	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 2.9 Практикум решения задач с применением функций	Практическое занятие 1. Локальные функции 2. Организация подпрограммы и функция main 3. Параметры функции	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 2.10 Рекурсивные функции	Самостоятельная работа 1. Написание программ, содержащих подпрограммы 2. Рекурсивные алгоритмы	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 2.11 Применение строк для кодирования информации	Самостоятельная работа 1. Кодирование информации 2. Написание программ для кодировки текста	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 2.12 Unicode и bytes в Python	Самостоятельная работа 1. Ограничение ASCII 2. Переход в Unicode 3. Кодировка UTF-8	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 2.13 Циклы и области видимости. Цикл for, цикл While	Самостоятельная работа 1. Цикл for 2. Функции reversed() и range() 3. Синтаксис цикла while и его применение 4. Область видимости в циклах	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 2.14 Списки – тип данных list	Самостоятельная работа 1. Переход от строк к спискам 2. Индексы и срезы 3. Перебор списков с элементами разных типов	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 2.15 Кортежи – тип данных tuple	Самостоятельная работа 1. Определение и основные характеристики 2. Стандартные операции 3. Поиск и сортировка	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5

Тема 2.16 Матрицы в Python	Самостоятельная работа 1. Общее представление о структуре матриц 2. Инициализация матриц 3. Обращение к матрицам	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 2.17 Файлы в Python	Самостоятельная работа 1. Открытие и закрытие файлов 2. Тестовые файлы 3. Запись в текстовый файл 4. Чтение файла	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 2.18 Модули в Python	Самостоятельная работа 1. Модули 2. Подключение функциональности модуля в глобальное пространство имен 3. Установка псевдонимов 4. Имя модуля	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 2.19 Модуль math	Самостоятельная работа 1. Основные функции модуля math 2. Примеры применения некоторых функций 3. Решение задач с применением функций модуля math	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 2.20 Итоговая контрольная работа	Итоговая контрольная работа Подведение итогов за семестр	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Раздел 3. Алгоритмы в Python		14/160	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.1 Введение в алгоритмы.	Самостоятельная работа 1. Что такое алгоритм 2. Виды алгоритмов 3. Свойства алгоритмов	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.2 Оценка алгоритмов. О-нотация	Содержание учебного материала 1. Зачем нужна оценка алгоритмов 2. Время и память 3. О-нотация 4. Основные оценки 5. Память	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.3 Оценка производительности кода	Самостоятельная работа 1. Общая информация 2. Оценка вручную 3. Модуль timeit	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.4 Выполнение оценки различных алгоритмов	Самостоятельная работа 1. Решение одной задачи разными способами 2. Оценка времени работы каждого алгоритма	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5

Тема 3.5 Алгоритмы сортировки массивов	Содержание учебного материала 1. Общая информация 2. Определение порядка и сортировки 3. Стандартные сортировки, реализованные в Python 4. Алгоритмы сортировки и их асимптотика	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.6 Решение практических задач сортировки массивов различными методами	Практическое занятие 1. Реализация метода «пузырька» при работе с матрицами 2. Реализация метода вставки при работе с матрицами 3. Реализация метода выбора при работе с матрицами	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.7 Алгоритмы поиска элементов в массиве	Самостоятельная работа 1. Общая информация 2. Линейный поиск 3. Бинарный поиск	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.8 Повторы в массиве – левый и правый бинарный поиск	Самостоятельная работа 1. Левый бинарный поиск 2. Правый бинарный поиск 3. Модуль bisect 4. Задача поиска двух элементов массива с фиксированной суммой	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.9 Динамическое программирование для поиска путей	Самостоятельная работа 1. Задача о кузнечике 2. Задача о роботе 3. Задача о роботе с препятствиями	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.10 Рекурсия. Алгоритмы поиска НОД и НОК	Самостоятельная работа 1. Делимость и простые числа 2. Определение НОД и НОК 3. Алгоритм нахождения НОД и НОК	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.11 Практикум решения задач на нахождение НОД и НОК	Самостоятельная работа 1. Реализация алгоритма нахождения НОД при решении задач 2. Решение задач на делимость чисел	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.12 Решето Эратосфена — поиск простых чисел	Самостоятельная работа 1. Простые числа 2. Решето Эратосфена 3. Сложность алгоритма	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.13 Практикум решения задач на поиск простых чисел	Самостоятельная работа 1. Реализация алгоритма решета Эратосфена на компьютере 2. Решение задач на определение простых чисел	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.14 Числа Фибоначчи	Самостоятельная работа 1. Описание чисел Фибоначчи 2. Алгоритм поиска чисел Фибоначчи	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.15 Практикум решения задач с	Самостоятельная работа 1. Задача о кроликах и золотое сечение	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5

числами Фибоначчи	2. Решение задач с числами Фибоначчи		
Тема 3.16 Решение различных задач на алгоритмы с числами	Самостоятельная работа 1. Решение задач на алгоритмы с числами	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.17 Словари — тип данных dict	Содержание учебного материала 1. Словари в Python 2. Операции над словарями 3. Методы словаря 4. Хеш-таблицы	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.18 Решение задач со словарями	Практическое занятие 1. Работа с ключами и значениями словаря 2. Генератор словарей	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.20 Множества — тип данных set	Самостоятельная работа 1. Что такое множества 2. Операции над множествами 3. Неизменяемое множество frozenset()	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.21 Решение задач с множествами	Самостоятельная работа 1. Работа с элементами множества 2. Решение задач на применение операций над множествами	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.22 Решение задач с неизменяемыми множествами	Самостоятельная работа 1. Применение неизменяемых множеств 2. Работа с операциями над неизменяемыми множествами	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.23 Алгоритмы со строками. Проверка на палиндром	Самостоятельная работа 1. Проверка на палиндром 2. Решение и оптимизация задачи	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.24 Вхождение подмножества символов и перестановки. Поиск подстроки в строке	Практическое занятие 1. Поиск символа 2. Словарь со счетчиком 3. Префикс-функция и алгоритм Кнута-Морриса-Пратта	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.25 Функции. Способы передачи аргументов в функцию	Самостоятельная работа 1. Множественное присваивание и оператор 2. Передача произвольного количества позиционных аргументов 3. Передача произвольного количества именованных аргументов 4. Комбинация *args и *kwargs	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.26 Передача произвольного количества позиционных аргументов	Самостоятельная работа 1. Решение задач на передачу произвольного количества позиционных аргументов в функцию	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.27 Передача произвольного количества именованных	Самостоятельная работа 1. Решение задач на передачу произвольного количества именованных аргументов	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5

аргументов			
Тема 3.28 Комбинация *args и **kwargs	Самостоятельная работа 1. Решение задач на комбинирование *args и **kwargs	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.29 Обработка исключений	Самостоятельная работа 1. Обработка исключений 2. Распространение исключений 3. Конструкция try-except-else--finally	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.31 Создание классов	Самостоятельная работа 1. Объекты в Python 2. Экземпляры класса 3. Классы 4. Атрибуты класса 5. Инспекция 6. Атрибуты экземпляра класса	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.32 Методы экземпляра класса	Самостоятельная работа 1. Функции внутри класса 2. Методы экземпляра класса 3. Инициализация объекта 4. Магические методы __repr__ и __str__ 5. Создание класса по принципу DRY	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.33 Решение задач на создание методов экземпляра класса	Практическое занятие 1. Создание классов и их атрибутов 2. Применение методов экземпляров классов при решении задач	6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.34 Магические методы __repr__ и __str__	Самостоятельная работа 1. Решение задач с применением магического метода __repr__ 2. Решение задач с применением магического метода __str__	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.35 Решение задач с применением магических методов	Самостоятельная работа 1. Решение задач с применением всех свойств класса	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.36 Принципы ООП. Полиморфизм и наследование	Самостоятельная работа 1. Полиморфизм в операциях 2. Полиморфизм в функциях 3. Наследование	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.37 Реализация принципа полиморфизма на конкретных задачах	Самостоятельная работа 1. Полиморфизм в ООП 2. Реализация принципа полиморфизма на конкретных примерах	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Тема 3.38 Реализация принципа наследования на конкретных	Самостоятельная работа 1. Перезапись методов 2. Делегирование	2	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5

задачах			
Консультация		4	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Экзамен		6	ОК 01., ОК 02., ОК 04., ОК 05., ОК 09., ПК 2.4., ПК 2.5
Всего часов		264	

Последовательное тематическое планирование содержания рабочей программы дисциплины, календарные объемы, виды занятий , формы организации самостоятельной работы также конкретизируются в календарно-тематическом плане (Приложение № 1)

3. ФОРМЫ КОНТРОЛЯ И МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, ПРАКТИЧЕСКОГО ОПЫТА, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ

Фонд оценочных средств (далее – ФОС) - комплект методических и контрольных материалов, используемых при проведении текущего контроля освоения результатов обучения и промежуточной аттестации. ФОС предназначен для контроля и управления процессом приобретения обучающимися необходимых знаний, умений и компетенций, определенных во ФГОС (Приложение № 2).

4. УСЛОВИЯ РЕАЛИЗАЦИИ ДИСЦИПЛИНЫ

4.1. Требования к минимальному материально-техническому обеспечению

Реализация программы дисциплины требует наличия учебных аудиторий

Аудитория № 5. Учебная аудитория для проведения: лекционных, семинарских, практических занятий, уроков, для проведения групповых и индивидуальных консультаций, для проведения текущего контроля и промежуточной аттестации, курсового проектирования (выполнения курсовых работ)

Учебная мебель, доска.

Аудитория № 204. Лаборатория технологий и методов программирования.

Доска, проектор, экран, учебная мебель, компьютеры, учебно-наглядные пособия.

Перечень лицензионного программного обеспечения:

Office Standart 2007 Russian OpenLicensePack NoLevel Acdmc;

Mathcad University Classroom Perpetual-15 Floating.

Аудитория № 144. Читальный зал. Помещение для самостоятельной работы обучающихся.

Учебная мебель, компьютеры.

Перечень лицензионного программного обеспечения:

Office Standart 2007 Russian OpenLicensePack NoLevel Acdmc

4.2. Учебно-методическое и информационное обеспечение дисциплины (модуля)

4.2.1. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)

Основная учебная литература:

Трофимов, В. В. Основы алгоритмизации и программирования : учебник для среднего профессионального образования / В. В. Трофимов, Т. А. Павловская ; под редакцией В. В. Трофимова. — Москва : Издательство Юрайт, 2020. — 137 с. — (Профессиональное образование). — ISBN 978-5-534-07321-8. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/454452>.

Кудрина, Е. В. Основы алгоритмизации и программирования на языке C# : учебное пособие для среднего профессионального образования / Е. В. Кудрина, М. В. Огнева. — Москва : Издательство Юрайт, 2020. — 322 с. — (Профессиональное образование). — ISBN 978-5-534-10772-2. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/456221>.

Дополнительная учебная литература:

Нагаева, И.А. Основы алгоритмизации и программирования: практикум : [12+] / И.А. Нагаева, И.А. Кузнецов. — Москва ; Берлин : Директ-Медиа, 2021. — 169 с. : схем. — Режим доступа: по подписке. — URL: <https://biblioclub.ru/index.php?page=book&id=598404> (дата обращения: 08.12.2020). — Библиогр.: с. 162-163. — ISBN 978-5-4499-1612-9. — DOI 10.23681/598404. — Текст : электронный.

4.2.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет» (далее - сеть «Интернет»), необходимых для освоения дисциплины (модуля)

№	Наименование электронной библиотечной системы
1	Договор на ЭБС «ЭБС ЮРАЙТ» (полная коллекция) между БашГУ в лице директора СФ БашГУ и ООО «Электронное издательство ЮРАЙТ» № 5-20 от 04.02.2020
2	Договор на ЭБС «Университетская библиотека онлайн» между БашГУ и «Нексмедиа» № 1132 от 23.09.2020
3	Договор на ЭБС между БашГУ и издательством «Лань» № 1130 от 28.09.2020
4	Договор на ЭБС между БашГУ и издательством «Лань» № 1131 от 28.09.2020
5	Договор о подключении к НЭБ и о предоставлении доступа к объектам НЭБ между БашГУ в лице директора СФ БашГУ с ФГБУ «РГБ» № 101/НЭБ/1438-П от 11.06.2019

	Адрес (URL)
1.	http://cyberguru.ru/
2.	http://fcior.edu.ru/
3.	http://www.ict.edu.ru/

4.2.3. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем (при необходимости)

Наименование программного обеспечения
Mathcad University Classroom Perpetual-15 Floating
Office Standard 2007 Russian OpenLicensePackNoLevelAcademic
PascalABC.NET
Python версия 3.8 и выше
PyCharm

ПРИЛОЖЕНИЕ № 1

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«БАШКИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

Стерлитамакский филиал

Колледж

Календарно-тематический план

по дисциплине

ОП.04 Основы алгоритмизации и программирования

	специальность
<i>09.02.07</i>	<i>Информационные системы и программирование</i>
код	наименование специальности
	квалификация
	<i>Администратор баз данных</i>

Разработчик (составитель)

Кучер А.М.

ученая степень, ученое звание,
категория, Ф.И.О.

Стерлитамак 2022

№ п/п	Наименование разделов и тем	Кол- во часов	Календа рные сроки изучени я (план)	Вид занятия	Домашнее задание обучающихся
Раздел № 1. Основные конструкции языка программирования Pascal					
1.	Типы данных и операции языка Pascal	2/2	сентябрь	Лекция	Учить конспект
2.	Операторы ветвления	2/4	сентябрь	Лекция	Учить конспект
3.	Операторы циклов	2/6	сентябрь	Лекция	Учить конспект
4.	Условный оператор. Практикум решения задач	2/8	сентябрь	Практическое занятие	Выполнить отчет по работе
5.	Оператор цикла с параметром	2/10	октябрь	Практическое занятие	Выполнить отчет по работе
6.	Оператор цикла с условием	2/12	октябрь	Практическое занятие	Выполнить отчет по работе
7.	Итоговая контрольная работа	2/14	декабрь		
Раздел 2. Основы Python					
8.	Типы данных – числа	2/16	январь	Лекция	Учить конспект
9.	Условные операторы	2/18	январь	Лекция	Учить конспект
10.	Понятие функции в Python	2/20	январь	Лекция	Учить конспект
11.	Линейные программы	2/22	февраль	Практические занятие	Выполнить отчет по работе
12.	Применение условного оператора при решении задач	2/24	февраль	Практические занятие	Выполнить отчет по работе
13.	Практикум решения задач с применением функций	2/26	февраль	Практические занятие	Выполнить отчет по работе
14.	Итоговая контрольная работа	2/28	июнь		
Раздел 3. Алгоритмы в Python					
15.	Оценка алгоритмов. О-нотация	2/30	сентябрь	Лекция	Учить конспект
16.	Алгоритмы сортировки массивов	2/32	сентябрь	Лекция	Учить конспект
17.	Решение практических задач сортировки массивов различными методами	2/34	сентябрь	Практическое занятие	Выполнить отчет по работе
18.	Словари — тип данных dict	2/36	октябрь	Лекция	Учить конспект
19.	Решение задач со словарями	2/38	октябрь	Практическое занятие	Выполнить отчет по работе
20.	Вхождение подмножества символов и перестановки. Поиск подстроки в строке	2/40	октябрь	Практическое занятие	Выполнить отчет по работе
21.	Решение задач на создание методов экземпляра класса	2/42	ноябрь	Практическое занятие	Выполнить отчет по работе
Экзамен					

ПРИЛОЖЕНИЕ 2

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«БАШКИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Стерлитамакский филиал

Колледж

Фонд оценочных средств

по дисциплине

ОП.04 Основы алгоритмизации и программирования

Общепрофессиональный цикл, обязательная часть

цикл дисциплины и его часть (обязательная, вариативная)

специальность

09.02.07

Информационные системы и программирование

код

наименование специальности

квалификация

Администратор баз данных

Разработчик (составитель)

Кучер А.М.

ученая степень, ученое звание,
категория, Ф.И.О.

Стерлитамак 2022

I Паспорт фондов оценочных средств

1. Область применения

Фонд оценочных средств (ФОС) предназначен для проверки результатов освоения дисциплины Основы алгоритмизации и программирования, входящей в состав программы подготовки специалистов среднего звена по специальности Информационные системы и программирование. **Работа обучающихся во взаимодействии с преподавателем 42, на самостоятельную работу 216.**

2. Объекты оценивания – результаты освоения дисциплины

ФОС позволяет оценить следующие результаты освоения дисциплины Основы алгоритмизации и программирования в соответствии с ФГОС специальности Информационные системы и программирование и рабочей программой дисциплины Основы алгоритмизации и программирования:

умения:

- Решать задачу на языке программирования на основе поставленной задачи предметной области.
- Производить поиск, анализ и интерпретацию доступной информации с целью решения задач профессиональной деятельности.
- Умение решать задачи в команде. Решение задач в формате проектной деятельности.
- Умение транслировать текстовую задачу в алгоритм и программу на языке программирования.
- Умение применять методы программирования для решения задач профессиональной области
- Использование документации языка с целью обучения и расширения знаний при разработке программного обеспечения
- Уметь разрабатывать программные алгоритмы и понимать ранее разработанные алгоритмы для решения задач профессиональной деятельности. Уметь писать программный код соответствующий техническому заданию.
- Уметь изменять внутреннюю структуру программного продукта не затрагивая её внешнего поведения, с целью облегчения понимания программного кода.
- Уметь писать тестовые программы с целью проверки правильности работы программного кода.
- Уметь использовать набор правил и соглашений, которые используются для написания исходного кода с целью оптимизации по времени и памяти.

знания:

- Методов и подходов для решения задач профессиональной области с применением языков программирования.
- Уметь работать с информацией: искать, аккумулировать, анализировать, перерабатывать для решения поставленной задачи.
- Решение задач на основе принципа декомпозиции с учетом разделения обязанностей в команде, распределения обязанностей внутри команды согласно навыкам и знаниям.
- Принципы решения задач
- Выбор языка программирования для каждой конкретной задачи, использование соответствующих типов данных, методов языка.

- Синтаксис и семантика языка программирования, основы формирования языка программирования
- Знать принципы и основы работы с документацией по выбранному языку программирования
- Знать основные алгоритмы и уметь разрабатывать свои.

Вышеперечисленные умения, знания направлены на формирование у обучающихся следующих **общих и профессиональных компетенций**:

ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам

ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности

ОК 04. Эффективно взаимодействовать и работать в коллективе и команде

ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста.

ОК 09 Пользоваться профессиональной документацией на государственном и иностранном языках

ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения.

ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.

3 Формы контроля и оценки результатов освоения дисциплины

Контроль и оценка результатов освоения – это выявление, измерение и оценивание знаний, умений и формирующихся общих и профессиональных компетенций в рамках освоения дисциплины.

В соответствии с учебным планом специальности Информационные системы и программирование, рабочей программой дисциплины Основы алгоритмизации и программирования предусматривается текущий и промежуточный контроль результатов освоения.

3.1 Формы текущего контроля

Текущий контроль успеваемости представляет собой проверку усвоения учебного материала, регулярно осуществляемую на протяжении курса обучения.

Текущий контроль результатов освоения дисциплины в соответствии с рабочей программой и календарно-тематическим планом происходит при использовании следующих обязательных форм контроля:

- *выполнение и защита практических работ,*
- проверка выполнения самостоятельной работы студентов,
- проверка выполнения контрольных работ,

Во время проведения учебных занятий дополнительно используются следующие формы текущего контроля – устный опрос, решение задач, тестирование по темам отдельных занятий.

Выполнение и защита практических работ. Практические работы проводятся с целью усвоения и закрепления практических умений и знаний, овладения профессиональными компетенциями. В ходе практической работы студенты приобретают умения, предусмотренные рабочей программой дисциплины, учатся использовать формулы, и применять различные методики расчета, анализировать полученные

результаты и делать выводы, опираясь на теоретические знания.

Список практических работ:

Ввод и вывод числовых данных. Целочисленная арифметика

Часть А

1. Дано трехзначное число. Найти число, полученное при прочтении его цифр справа налево.
2. Дано трехзначное число. В нем зачеркнули первую слева цифру и приписали ее в конце. Найти полученное число.
3. Дано трехзначное число. В нем зачеркнули последнюю справа цифру и приписали ее в начале. Найти полученное число.
4. Дано трехзначное число. Найти число, полученное при перестановке первой и второй цифр заданного числа.
5. Дано трехзначное число. Найти число, полученное при перестановке второй и третьей цифр заданного числа.
6. Дано трехзначное число, в котором все цифры различны. Получить шесть чисел, образованных при перестановке цифр заданного числа.
7. Дано четырехзначное число. Найти число, полученное при прочтении его цифр справа налево.
8. Дано четырехзначное число. Найти число, образуемое при перестановке первой и второй, третьей и четвертой цифр заданного числа. Например, из числа 5434 получить 4543, из числа 7048 — 784.
9. Дано четырехзначное число. Найти число, образуемое при перестановке второй и третьей цифр заданного числа. Например, из числа 5084 получить 5804.
10. Дано четырехзначное число. Найти число, образуемое при перестановке двух первых и двух последних цифр заданного числа. Например, из числа 4566 получить 6645, из числа 7304 — 473.
11. Дано четырехзначное число. Найти сумму и произведение его цифр.
12. Дано натуральное число n ($n > 9$). Найти число единиц и число десятков в нем.
13. Дано натуральное число n ($n > 99$). Найти число десятков и число сотен в нем.
14. Дано натуральное число n ($n > 999$). Найти число сотен и число тысяч в нем.
15. Дано трехзначное число. Найти число единиц и число десятков в нем; сумму и произведение его цифр.

Часть В

Найти значение выражения

1.	$a = \frac{\sqrt{ x-1 } - \sqrt[3]{ y }}{1 + \frac{x^2}{2} + \frac{y^2}{4}}$
2.	$b = x (\operatorname{arctg} z + e^{-(x+3)})$
3.	$a = \frac{3 + e^{y-1}}{1 + x^2 y - \operatorname{tg} z }$
4.	$b = 1 + y-x + \frac{(y-x)^2}{2} + \frac{ y-x }{3}$

5.	$a = (1 + y) \frac{x + y/(x^2 + 4)}{e^{-x-2} + 1/(x^2 + 4)}$
6.	$b = \frac{1 + \cos(y - 2)}{x^4/2 + \sin^2 z}$
7.	$a = \frac{1 + \sin^2(x + y)}{2 + x - 2x/(1 + x^2 y^2) } + x$
8.	$a = y + \frac{x}{y^2 + \left \frac{x^2}{y + x^3/3} \right }$
9.	$b = \left(1 + \operatorname{tg}^2 \frac{z}{2} \right)$
10.	$a = \frac{2 \cos(x - \pi/6)}{1/2 + \sin^2 y}$
11.	$b = 1 + \frac{z^2}{3 + z^2/5}$
12.	$a = \ln \left \left(y - \sqrt{ x } \right) \left(x - \frac{y}{z + x^2/4} \right) \right $
13.	$b = x - \frac{x^2}{3!} + \frac{x^5}{5!}$
14.	$b = \cos^2 \left(\operatorname{arctg} \frac{1}{z} \right)$
15.	$b = \frac{1 + \cos(y - 2)}{x^4/2 + \sin^2 z}$

«Условный оператор»

Задание 1.

№ вар.	Функция
1	$y = \begin{cases} x - 2 & x > 1.5 \\ 1 + x^2 & 0 \leq x \leq 2.5 \\ x \cdot \ln \cos(x) & x < 0 \end{cases}$
2	$y = \begin{cases} \sin(2.3x - 1) & x > 2.5 \\ 1 - 3 \ln 1 - x & 0 \leq x \leq 2.5 \\ \frac{x^2}{2 - x} & x < 0 \end{cases}$
3	$y = \begin{cases} \sqrt{\tan(x^2 - 1)} & x > 1 \\ -2x & 0 \leq x \leq 1 \\ e^{\cos(x)} & x < 0 \end{cases}$
4	$y = \begin{cases} x^2 - 3 + 2.5x^2 & x > 12.5 \\ e^x + 5 + \cos(0.001x) & 0 \leq x \leq 12.5 \\ x^2 & x < 0 \end{cases}$
5	$y = \begin{cases} 1 + \sqrt{ \cos(x) } & x > 1 \\ x + 1 & -0.5 \leq x \leq 1 \\ 1 - x^2 & x < -0.5 \end{cases}$
6	$y = \begin{cases} 2.5 \cdot x^3 + 6 \cdot x^2 - 30 & x > 1.5 \\ x + 1 & 0 \leq x \leq 1.5 \\ x & x < 0 \end{cases}$
7	$y = \begin{cases} 1 + x & x > 14.5 \\ e^{-x} & 3 \leq x \leq 14.5 \\ \cos(x) & x < 3 \end{cases}$
8	$y = \begin{cases} \ln 1 + x & x > 3.8 \\ e^{-x} & 2.8 \leq x \leq 3.8 \\ \cos(x) & x < 2.8 \end{cases}$

9	$y = \begin{cases} 1 + \sqrt{\cos(x)} & x > 4 \\ x + 1 & 0 \leq x \leq 4 \\ 1 - x^2 & x < 0 \end{cases}$
10	$y = \begin{cases} e^{-(x+8)} & x > 3.61 \\ 1 & 0 \leq x \leq 3.61 \\ \frac{x}{5} & x < 0 \end{cases}$
11	$y = \begin{cases} x & x > 1.5 \\ 2x^2 \sqrt{ \cos(2x) } & 0 \leq x \leq 1.5 \\ e^{-\cos(3x)} & x < 0 \end{cases}$
12	$y = \begin{cases} 1 - \sqrt{\cos(2x)} & x > 2.5 \\ x^2 - x & 1 \leq x \leq 2.5 \\ 1 + x^2 & x < 1 \end{cases}$
13	$y = \begin{cases} 2x & x > 4.5 \\ 1 - \ln 1 - x^2 & 0 \leq x \leq 4.5 \\ e^{-x} & x < 0 \end{cases}$
14	$y = \begin{cases} \sqrt{\ln x^2 - 1 } & x > 2 \\ -2x^3 & 0 \leq x \leq 2 \\ e^{\sin(x)} & x < 0 \end{cases}$
15	$y = \begin{cases} e^{-(x+8)} & x > 3.61 \\ 1 & 0 \leq x \leq 3.61 \\ \frac{x}{5} & x < 0 \end{cases}$

Задание 2.

- Даны положительные числа a , b , c и d . Выяснить, можно ли прямоугольник со сторонами a , b уместить внутри прямоугольника со сторонами c , d так, чтобы каждая из сторон одного прямоугольника была параллельна или перпендикулярна каждой стороне второго прямоугольника. Ответ получите в текстовой форме: 'можно' или 'нельзя'.
- Даны положительные числа a , b , c , x . Выяснить пройдет ли кирпич с ребрами a , b и c в квадратное отверстие со стороной x . Просовывать кирпич в отверстие разрешается только так, чтобы каждое из его ребер было параллельно или перпендикулярно каждой из сторон отверстия. Ответ получите в текстовой форме: 'можно' или 'нельзя'.

3. Определить, войдет ли в конверт с внутренними размерами a и b мм прямоугольная открытка размером c и d мм. Для размещения открытки в конверте необходим зазор в 1 мм с каждой стороны.
4. Вася пытается высунуть голову в форточку размерами a и b см. Приняв условно, что его голова — круглая диаметром d см, определить, сможет ли Вася сделать это. Для прохождения головы в форточку необходим зазор в 1 см с каждой стороны.
5. Год является високосным, если его номер кратен 4, однако из кратных 100 високосными являются лишь кратные 400, например, 1700, 1800 и 1900 — не високосные года, 2000 — високосный. Дано натуральное число n . Определить, является ли високосным год с таким номером.
6. Дано целое число k ($1 \leq k \leq 365$). Определить, каким будет k -й день года: выходным (суббота и воскресенье) или рабочим, если 1 января — понедельник.
7. Даны четыре целых числа. Определить сумму тех из них, которые кратны трем. Оператор цикла не использовать.
8. В чемпионате по футболу команде за выигрыш дается 3 очка, за проигрыш — 0, за ничью — 1. Известно количество очков, полученных командой за игру. Определить словесный результат игры (выигрыш, проигрыш или ничья).
9. Даны две тройки вещественных чисел. В каждой тройке все числа различные. Найти среднее арифметическое средних чисел каждой тройки (средним назовем такое число в тройке, которое больше наименьшего из чисел данной тройки, но меньше наибольшего).
10. Вывести на экран номер четверти координатной плоскости, которой принадлежит точка с координатами (x, y) , при условии, что $x \neq 0$ и $y \neq 0$.
11. Определить, является ли заданное шестизначное число счастливым. (Счастливым называют такое шестизначное число, что сумма его первых трех цифр равна сумме его последних трех цифр.)
12. Определить, является ли число a делителем числа b или, наоборот, число b делителем числа a . Ответом должны служить сообщения "Да, одно из чисел является делителем другого" или "Нет, ни одно из чисел не является делителем другого".
13. Составить программу нахождения суммы двух наибольших из трех различных чисел.
14. Составить программу нахождения произведения двух наименьших из трех различных чисел.
15. Даны четыре вещественных числа. Определить, сколько из них отрицательных. Оператор цикла не использовать.

Оператор цикла с параметром

Часть А

1. Напечатать числа следующим образом:

21	19.2
20	18.2
.....	
10	8.2

2. Напечатать числа следующим образом:

45	44.5	44.2
44	43.5	43.2
...		
25	24.5	24.2

3. Напечатать числа следующим образом:

21	20.4
22	21.4
...	
35	34.4

4. Напечатать числа следующим образом:

16	15.5	16.8
17	16.5	17.8
...		
24	23.5	24.8

5. Напечатать числа следующим образом:

10	10.4
11	11.4
...	
25	25.4

6. Напечатать числа следующим образом:

25	25.5	24.8
26	26.5	25.8
...		
35	35.5	34.8

7. Одна штука некоторого товара стоит 20,4 руб. Напечатать таблицу стоимости 2, 3, ..., 20 штук этого товара.

8. Напечатать таблицу соответствия между весом в фунтах и весом в килограммах для значений 1, 2, ..., 10 фунтов (1 фунт = 453 г).

9. Напечатать таблицу перевода расстояний в дюймах в сантиметры для значений 10, 11, ..., 22 дюйма (1 дюйм = 25,4 мм).

10. Напечатать таблицу перевода 1, 2, ... 20 долларов США в рубли по текущему курсу (значение курса вводится с клавиатуры).

11. Напечатать таблицу умножения на 7:

$$1 \times 7 = 7$$

$$2 \times 7 = 14$$

...

$$9 \times 7 = 63$$

12. Напечатать таблицу умножения на 9:

$$9 \times 1 = 9$$

$$9 \times 2 = 18$$

...

$$9 \times 9 = 81$$

13. Напечатать таблицу умножения на число n (значение n вводится с клавиатуры; $1 \leq n \leq 9$).

14. Напечатать "столбиком" значения $\sin 2, \sin 3, \dots, \sin 20$.

15. Напечатать таблицу стоимости 50, 100, 150, ..., 1000 г сыра (стоимость 1 кг сыра вводится с клавиатуры).

Часть Б

1. Вычислить сумму $1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$.

2. Вычислить сумму $\frac{2}{3} + \frac{3}{4} + \frac{4}{5} + \dots + \frac{10}{11}$.

3. Вычислить сумму $1 + \frac{1}{3} + \frac{1}{3^2} + \dots + \frac{1}{3^8}$. Операцию возведения в степень не использовать.

4. Вычислить сумму $1 - \frac{1}{2} + \frac{1}{3} - \dots + (-1)^{n+1} \frac{1}{n}$. Условный оператор и операцию возведения в степень не использовать.

5. Вычислить сумму $x + \frac{x^3}{3} + \frac{x^5}{5} + \dots + \frac{x^{11}}{11}$. Значение x вводится с клавиатуры.

6. Вычислить сумму $1 - \frac{2}{3}x + \frac{3}{4}x^2 - \frac{4}{5}x^3 + \dots + \frac{11}{12}x^{10}$. Значение x вводится с клавиатуры.

7. Известны оценки двух учеников по четырем предметам. Определить сумму оценок каждого ученика. Массив не использовать.

8. Известны результаты двух спортсменов-пятиборцев в каждом из пяти видов спорта в баллах. Определить сумму баллов, полученных каждым спортсменом. Массив не использовать.

9. В области 12 районов. Известны количество жителей (в тысячах человек) и площадь (в км²) каждого района. Определить среднюю плотность населения по области в целом. Массив не использовать.

10. В области 12 районов. Известны количество жителей каждого района (в тысячах человек) и плотность населения в нем (тыс. чел./км²). Определить общую площадь территории области. Массив не использовать.

11. В ведомости указана зарплата, выплаченная каждому из сотрудников фирмы за месяц. Определить общую сумму выплаченных по ведомости денег. Массив не использовать.

12. Известна масса каждого предмета, загружаемого в автомобиль. Определить общую массу груза.

13. Известно сопротивление каждого из элементов электрической цепи. Все элементы соединены последовательно. Определить общее сопротивление цепи. Массив не использовать.

14. Известно сопротивление каждого из элементов электрической цепи. Все элементы соединены параллельно. Определить общее сопротивление цепи. Массив не использовать.

15. В области 10 районов. Заданы площади, засеваемые пшеницей (в гектарах), и средняя урожайность (в центнерах с гектара) в каждом районе. Определить количество пшеницы, собранное в области, и среднюю урожайность по области. Массив не использовать.

Решение практических задач сортировки массивов различными методами

Сложность А

1. Написать программу, сортирующую по возрастанию методом выбора главную диагональ двумерного массива целых чисел.
2. Написать программу, сортирующую по убыванию методом вставок главную диагональ двумерного массива целых чисел.
3. Написать программу, сортирующую по возрастанию методом пузырька главную диагональ двумерного массива целых чисел.
4. Написать программу, сортирующую по убыванию методом выбора побочную диагональ двумерного массива целых чисел.
5. Написать программу, сортирующую по возрастанию методом вставок побочную диагональ двумерного массива целых чисел.
6. Написать программу, сортирующую по убыванию методом пузырька побочную диагональ двумерного массива целых чисел.
7. Написать программу, сортирующую методом выбора двумерный массив целых чисел. При этом самое маленькое число должно оказаться в первом элементе первой строки, а самое большое в последнем элементе последней строки.
8. Написать программу, сортирующую методом вставок двумерный массив целых чисел. При этом самое маленькое число должно оказаться в первом элементе первой строки, а самое большое в последнем элементе последней строки.
9. Написать программу, сортирующую методом пузырька двумерный массив целых чисел. При этом самое маленькое число должно оказаться в первом элементе первой строки, а самое большое в последнем элементе последней строки.
10. Написать программу, сортирующую по возрастанию последний столбец двумерного массива целых чисел. Использовать сортировку выбором.
11. Написать программу, сортирующую по убыванию первый столбец двумерного массива целых чисел. Использовать сортировку вставкой.
12. Написать программу, сортирующую по возрастанию последнюю строку двумерного массива целых чисел. Использовать сортировку пузырьком.
13. Написать программу, сортирующую по убыванию первую строку двумерного массива целых чисел. Использовать сортировку вставкой.

Сложность Б

1. Написать программу, сортирующую строки двумерного массива целых чисел по возрастанию суммы элементов строк и использующую сортировку вставкой.
2. Написать программу, сортирующую столбцы двумерного массива чисел по убыванию суммы последних четырех чисел столбца и использующую сортировку пузырьком.

3. Написать программу, сортирующую строки двумерного массива чисел по возрастанию суммы первых трех чисел строки и использующую сортировку выбором.
4. Написать программу, сортирующую строки двумерного массива целых чисел по убыванию произведения элементов строк и использующую сортировку вставкой.
5. Написать программу, сортирующую строки двумерного массива целых чисел по убыванию суммы элементов строк и использующую сортировку пузырьком.
6. Написать программу, сортирующую строки двумерного массива целых чисел по убыванию суммы элементов строк, расположенных выше главной диагонали, и использующую сортировку вставкой.
7. Написать программу, сортирующую столбцы двумерного массива целых чисел по возрастанию суммы элементов столбцов и использующую сортировку вставкой.
8. Написать программу, сортирующую строки двумерного массива чисел по возрастанию суммы последних трех чисел строки и использующую сортировку пузырьком.
9. Написать программу, сортирующую строки двумерного массива чисел по возрастанию суммы первых пяти чисел строки и использующую сортировку вставкой.
10. Написать программу, сортирующую строки двумерного массива целых чисел по убыванию суммы элементов строк, расположенных ниже главной диагонали, и использующую сортировку пузырьком.
11. Написать программу, сортирующую столбцы двумерного массива целых чисел по убыванию произведения элементов столбца и использующую сортировку выбором.
12. Написать программу, сортирующую строки двумерного массива целых чисел по убыванию произведения первых трех элементов строк и использующую сортировку вставкой.
13. Написать программу, сортирующую строки двумерного массива целых чисел по убыванию суммы элементов строк, расположенных ниже главной диагонали, и использующую сортировку пузырьком.

Пример задания матриц в Python

```
m=10
n=5

matrix = []
for i in range(m):
    matrix.append([])

    for j in range(n):
        a = int(input())
        matrix[i].append(a)
```

Пример вывода матрицы на экран

```
for i in range(len(matrix)):
    print(matrix[i])
```

Повторы в массиве – левый и правый бинарный поиск

Линейный поиск

На вход подается массив, состоящий из целочисленных элементов, упорядоченных по возрастанию. Также подается целое число x , известно, что в массиве нет элементов, равных по значению x . С помощью линейного поиска найдите позицию (индекс), на которую нужно вставить элемент x (подвинув элементы, стоящие справа), чтобы сохранить упорядоченность массива.

Тесты

<i>Входные данные</i>	<i>Выходные данные</i>
1 2 4 8 0	0
34 106 208 301	3
1 3 5 7 9 11 13 12	6
1 40 75 89 302 302 699 701 899 1000 700	7
1 2 2 3 4 6 7 88 5	5

Поиск позиции

На вход подается массив, состоящий из целочисленных элементов, упорядоченных по возрастанию. Также подается целое число x , известно, что в массиве нет элементов, равных по значению x . С помощью бинарного поиска найдите позицию (индекс), на которую нужно вставить элемент x (подвинув элементы, стоящие справа), чтобы сохранить упорядоченность массива.

<i>Входные данные</i>	<i>Выходные данные</i>
1 2 4 8 0	0

34 106 208 301	3
1 3 5 7 9 11 13 12	6
1 40 75 89 302 302 699 701 899 1000 700	7
1 2 2 3 4 6 7 88 5	5

Единственное пропущенное число

Дан упорядоченный целочисленный массив длины N, содержащий уникальные значения в диапазоне от 0 до N включительно. С помощью бинарного поиска найдите единственное пропущенное число из диапазона 0-N.

Тесты

<i>Входные данные</i>	<i>Выходные данные</i>
1 2 3 4 5 6	0
0 1 2 3 4 6	5
0 1 2 3 4 5	6
1 2 3 4 5 6 7 8 9 10	0
0 1 2 3 4 5 7 8 9 10	6

Индекс последнего элемента в массиве

В программе задан массив, состоящий из некоторого количества целочисленных элементов. На вход программы передается целое число x. Используя линейный поиск (полный обход массива) найдите индекс последнего элемента в массиве, который больше x и является четным. Выведите найденный индекс на экран.

Если индекс не найден, верните -1.

Тесты

arr = [1, 12, 4, 7, 11, 4, 8]

<i>Входные данные</i>	<i>Выходные данные</i>
6	6
11	1
100	-1

Ресторан

Вы работаете над проектом для местного ресторана в управлении запасами ингредиентов. Владелец ресторана хочет знать, есть ли у него на складе определенный ингредиент. Вам дается список ингредиентов, и владелец ресторана сообщает вам название ингредиента, который он ищет. Напишите функцию Python, которая будет возвращать, есть ли ингредиент на складе или нет, используя бинарный поиск.

Тесты

<i>Входные данные</i>	<i>Выходные данные</i>
apple banana carrot lettuce onion tomato lettuce	True
flour sugar butter eggs milk eggs	True
carrot lettuce onion tomato broccoli	False
apple banana orange pineapple pear	False
beef chicken fish pork fish	True

Динамическое программирование для поиска путей

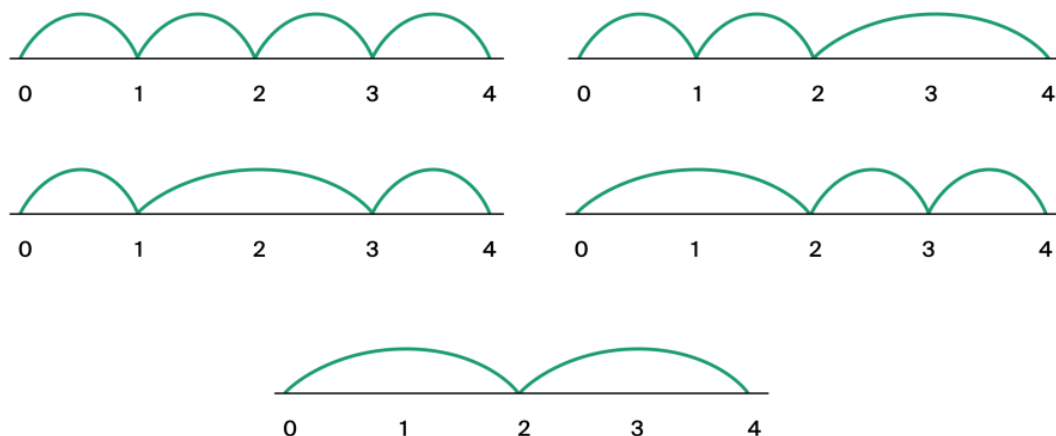
Задача о кузнечике

Рассмотрим следующую задачу.

Пример 1

На числовой прямой сидит кузнечик, который может прыгать вправо на одну или две единицы. Первоначально кузнечик находится в точке с координатой 0. Сколько существует способов для кузнечика добраться до координаты 4?

Попробуем сначала перебрать все варианты.



Как видно, кузнечик может пятью способами допрыгать до координаты 4.

Повторим подсчет, немного его структурируя.

1. Изначально кузнечик находится в нулевой координате. Сколько у него способов добраться до координаты 1? Всего 1.
2. Сколько у него способов добраться до координаты 2? В координату 2 он может попасть как из 1, так и из 0. То есть 2 способа.
3. Сколько у него способов добраться до 3? Аналогично предыдущему, попасть в 3 можно из 1 и 2. Но мы помним, что в координату 2 можно попасть двумя способами. В итоге, получаем $1 + 2 = 3$ способа.
4. Наконец, сколько существует способов добраться до 4? Туда попадаем либо из 3, либо из 2. Причем до 2 можно добраться двумя способами, а до 3 — тремя. Итого, получаем ответ к нашей задаче: $2 + 3 = 5$.

То есть число способов попасть кузнечнику в конкретную точку — это сумма способов попасть в координату, меньшую на 1, и в координату, меньшую на 2.

Попробуем теперь решить эту же задачу для общего случая, когда кузнечнику нужно добраться до координаты $n > 0$. Обозначим количество маршрутов кузнечика, ведущих в точку с координатой n , через $K(n)$. Тогда предыдущие соображения можно записать следующим образом.

1. $K(1) = 1$

2. $K(2) = 2$
3. $K(3) = K(2) + K(1) = 3$
4. $K(4) = K(3) + K(2) = 5$

Или в общем виде:

$$K(i) = K(i - 1) + K(i - 2), i = 2, 3, \dots, n.$$

Последовательно вычисляя значения $K(i)$, мы в конечном итоге найдем нужное значение $K(n)$. В этом и состоит основная идея динамического программирования.

Примечание

Полученное выражение для $K(i)$ называется **рекуррентным соотношением**, потому что оно связывает число в последовательности с предыдущими членами. Чтобы найти любой член в такой последовательности, нужно задать начальные данные. В нашем случае достаточно задать $K(0)=1$ (вырожденный случай, в точку 0 единственным образом попадаем из нее же) и $K(1)=1$.

Приведем реализацию нашего алгоритма.

Код

```
n = 4
K = [0] * (n + 1)
K[0] = 1
K[1] = 1
for i in range(2, n + 1):
    K[i] = K[i - 1] + K[i - 2]
print(K)
print(K[n])
```

Вывод

```
[1, 1, 2, 3, 5]
5
```

Сложность такого алгоритма $O(n)$, поскольку он сводит решение задачи к заполнению одномерного массива длины $n+1$.

Пример 2

Немного модифицируем предыдущую задачу. Пусть теперь кузнечик способен

прыгать вправо на расстояние 1, 2 и 3. Мы хотим определить количество различных маршрутов кузнечика, приводящих его в определенную точку на прямой.

Несложно догадаться, что есть один ключевой отличительный момент — теперь в конкретную точку i можно попасть из точек с координатами $i - 1$, $i - 2$, $i - 3$. То есть для данной задачи наше рекуррентное соотношение на количество $T(n)$ маршрутов кузнечика, ведущих в точку с координатой n , будет иметь вид

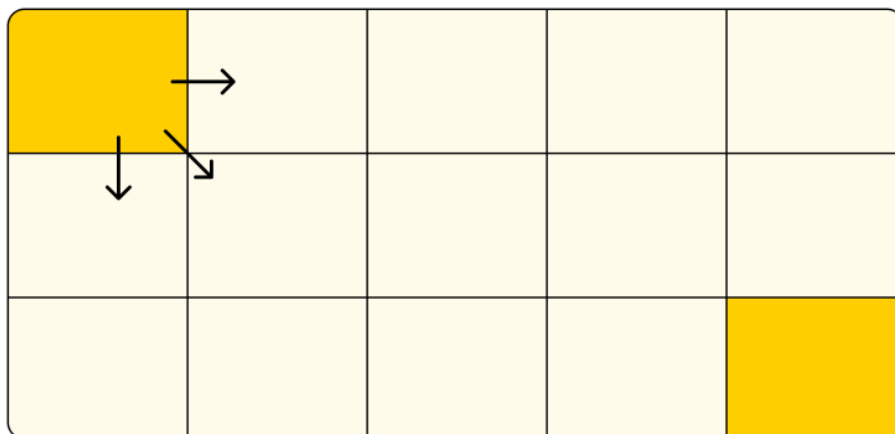
$$T(i) = T(i - 1) + T(i - 2) + T(i - 3), i = 3, 4, \dots$$

Задача о работе

Только что на примере кузнечика мы рассмотрели принцип одномерного динамического программирования. Однако часто возникает потребность решать задачи о нахождении количества путей в двумерном случае. Рассмотрим следующую задачу.

Пример 1

Пусть у нас есть робот, стартующий в левом верхнем углу сетки. Данный робот запрограммирован двигаться только вправо, вниз или по диагонали вправо-вниз. Мы хотим исследовать, сколькими способами робот может дойти от точки $(0, 0)$ до точки (x, y) .



На помощь вновь приходит динамическое программирование. Идея та же: оказаться в конкретной клетке можно всего тремя способами — перейдя из клетки слева, из клетки сверху и из клетки по диагонали слева-сверху. Поэтому, если обозначить количество маршрутов до клетки с координатами (i, j) через $P(i, j)$, снова получим рекуррентное соотношение

$$P(i, j) = P(i - 1, j) + P(i, j - 1) + P(i - 1, j - 1).$$

То есть отличие от задачи с кузнечиком заключается в том, что заполняется не одномерный массив, а таблица. В качестве начальных данных здесь нужно задать значения P для первой строки и первого столбца, то есть $P(i, 0) = P(0, j) = 1$, где $i = 0, 1, \dots$, x и $j = 0, 1, \dots, y$.

Напишем реализацию нашего алгоритма.

Код

```
x = 3
y = 4
P = [[0] * y for i in range(x)]
for i in range(x):
    P[i][0] = 1
for j in range(y):
    P[0][j] = 1
for i in range(1, x):
    for j in range(1, y):
        P[i][j] = P[i][j - 1] + P[i - 1][j] + P[i - 1][j - 1]
for line in P:
    print(line)
print(P[x - 1][y - 1])
```

Вывод

```
[1, 1, 1, 1]
[1, 3, 5, 7]
[1, 5, 13, 25]
25
```

Данный алгоритм по сути заключается в заполнении таблицы размером x на y . В коде это реализуется с помощью двух вложенных циклов. Поэтому сложность данного алгоритма $O(xy)$.

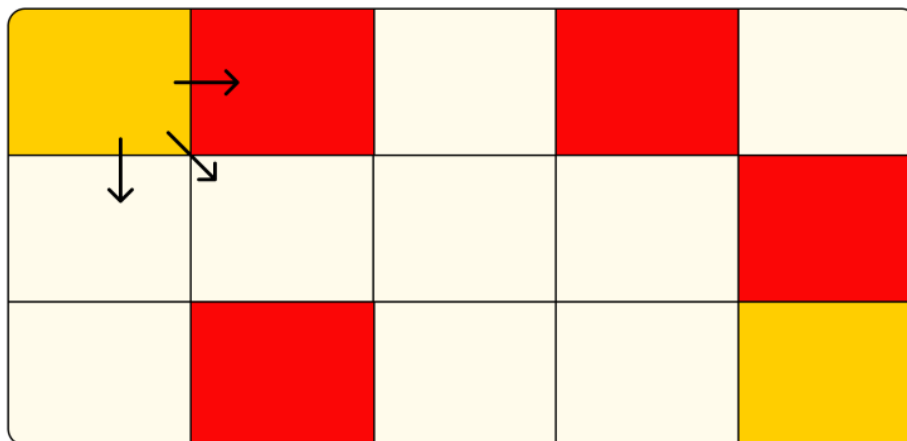
Задача о роботе с препятствиями

Модифицируем задачу из предыдущей секции, введя запрет на посещение некоторых

клеток.

Пример 1

Пусть теперь робот не может попасть в красные клетки.



Храниться информация о непреодолимых препятствиях будет в отдельном двумерном массиве `Obstacle[x][y]`:

- `Obstacle[i][j] = 0`, если клетку (i, j) разрешено посещать;
- `Obstacle[i][j] = 1`, если клетка (i, j) — это препятствие.

Тогда в алгоритм нужно добавить проверку, является ли клетка красной. Иными словами, если нам встретилась подобная клетка, мы точно знаем, что в ней робот оказаться не мог, и количество способов попасть в красную клетку равно 0. Также при задании начальных данных нужно учитывать, что если при заполнении первой строки/столбца таблицы встретилось препятствие, то далее по строке/столбцу роботу никак не пройти и оставшуюся часть нужно будет заполнить 0.

Задачи для самостоятельного решения

Маршрут с наименьшей стоимостью

В программе задана целочисленная матрица произвольного размера. На вход переданы два целых числа x и y такие, что оба они укладываются в размерность матрицы, то есть элемент матрицы с координатами (x, y) достижим.

Допустим, что в каждой клетке матрицы приписано некоторое число — стоимость прохождения через данную клетку. Двигаться можно только с увеличением координат по прямой или по диагонали. Нас интересует конкретный маршрут из клетки $(0, 0)$ в клетку (x, y) имеющий наименьшую стоимость. Выведите на экран сумму значений элементов по этому маршруту.

Cost = [[1, 1, 1, 1, 1],
 [100, 100, 100, 100, 1],
 [1, 1, 1, 100, 1],
 [1, 100, 100, 100, 1],
 [1, 1, 1, 1, 1]]

Тесты

Входные данные	Выходные данные
4 4	8
3 3	202
0 0	1

Минимальная скорость кузнечика

Вспомните задачу о кузнечике, прыгающем на 1 и 2. Модифицируем данную задачу, введя стоимости для каждого прыжка. Пусть стоимости будут различны для каждой точки и хранятся в массиве **Cost**, таким образом стоимость прыжка в точку **i** определяется значением **Cost[i]**. Найдите минимальную стоимость, за которую кузнечик может добраться из точки 0 до n, где n — длина массива **Cost**.

Тесты

Входные данные	Выходные данные
1 1	1
1 2 3 4	6
0 0 0 0	0
1 4 2 5 3 0 8 2	7

Матрица препятствий

Реализуйте алгоритм для работа с заданной матрицей препятствий.

Obstacle = [[0, 1, 0, 0],
 [0, 0, 1, 0],
 [1, 0, 0, 0]]

Практикум решения задач на нахождение НОД и НОК

Деление отрезка

На вход программы подаются два различных двузначных числа: начало и конец отрезка. Необходимо вывести сумму всех натуральных чисел из этого отрезка (включая начало и конец), которые делятся на каждую свою цифру, не считая нулей.

Тесты

Входные данные	Выходные данные
10 99	1080
10 20	68
20 21	20
98 99	99
30 90	867

Наибольший общий делитель

Реализуйте функцию, которая на вход принимает массив целых чисел и печатает их наибольший общий делитель.

Тесты

Входные данные	Выходные данные
748 1232 4400 1364	44
1 2 3	1
2 4 6 8	2
2 4 6 -8 3	1
-2 -4 -6 -8	2

Наименьшее общее кратное

Напишите функцию, которая принимает на вход 2 числа и возвращает наименьшее общее кратное этих чисел, используя связь между НОК и НОД.

Программа должна принимать на вход два целых числа и выводить на экран результат работы вашей функции — наименьшее общее кратное переданных чисел.

Входные данные	Выходные данные
126 12	252
1 1	1
7 14	14

345 234	26910
------------	-------

Практикум решения задач на поиск простых чисел

Замечание. При решении всех задач использовать решето Эратосфена

1. Найти простые числа в заданном интервале.
2. Найти первое простое число, сумма цифр которого больше 8, а произведение цифр меньше 10.
3. Найти все простые двузначные числа, заканчивающиеся на цифру 3.

Практикум решения задач с числами Фибоначчи

Вывод чисел Фибоначчи

Согласно определению последовательности Фибоначчи, чтобы найти n-е число, нам потребуется лишь два предыдущих: с номерами n-1 и n-2. Таким образом, нам достаточно хранить лишь их. Используя это соображение, напишите функцию, которая выводит n-е число Фибоначчи, используя лишь две переменные (a и b) и две арифметические операции: сложение и вычитание.

Для данного примера примем соглашение, что первые два числа из последовательности Фибоначчи равны 1.

Формула Бине

Рекуррентные соотношения можно превращать в явные формулы. Не будем вдаваться в детали, как это происходит. Скажем лишь, что для чисел Фибоначчи с помощью некоторых рассуждений можно вывести следующую формулу, известную как формула Бине:

$$F_n = \frac{\left(\frac{1 + \sqrt{5}}{2}\right)^n - \left(\frac{1 - \sqrt{5}}{2}\right)^n}{\sqrt{5}}$$

Напишите программу с использованием модуля math, которая получает на вход число n и печатает n-е число Фибоначчи, вычисляя его по формуле Бине.

Отметим, что формула Бине точна математически, но компьютер оперирует лишь дробями конечной точности, и при действиях с ними может накопиться ошибка, поэтому результат стоит округлять.

Номер числа Фибоначчи

Напишите программу, которая вводит натуральное число N, и, если N – число Фибоначчи, выводит его номер в последовательности, а если нет – выводит число -1.

Последняя цифра числа Фибоначчи

Напишите программу, которая вводит натуральное число N и выводит на экран последнюю цифру числа Фибоначчи. Для нахождения чисел Фибоначчи реализовать рекурсивный алгоритм

Проверка выполнения самостоятельной работы. Самостоятельная работа

направлена на самостоятельное освоение и закрепление обучающимися практических умений и знаний, овладение профессиональными компетенциями.

Самостоятельная подготовка обучающихся по дисциплине предполагает следующие виды и формы работы:

- Систематическая проработка конспектов занятий, учебной и специальной технической литературы.
- Самостоятельное изучение материала и конспектирование лекций по учебной и специальной технической литературе.
- Выполнение расчетных заданий.
- Оформление отчетов по практическим работам, и подготовка к их защите.

Проверка выполнения контрольных работ. Контрольная работа проводится с целью контроля усвоенных умений и знаний и последующего анализа типичных ошибок и затруднений обучающихся в конце изучения темы или раздела. Согласно календарно-тематическому плану дисциплины предусмотрено проведение следующих контрольных работ:

Задание для контрольной работы №1

Вариант 1

1. Описать структуру с именем AEROFLOT, содержащую следующие поля:
 - название пункта назначения рейса;
 - номер рейса;
 - тип самолета.
2. Написать программу, выполняющую следующие действия:
 - ввод с клавиатуры данных в массив, состоящий из семи элементов типа AEROFLOT; записи должны быть упорядочены по возрастанию номера рейса;
 - вывод на экран номеров рейсов и типов самолетов, вылетающих в пункт назначения, название которого совпало с названием, введенным с клавиатуры;
 - если таких рейсов нет, выдать на дисплей соответствующее сообщение.

Вариант 2

1. Описать структуру с именем WORKER, содержащую следующие поля:
 - фамилия и инициалы работника;
 - название занимаемой должности;
 - год поступления на работу.
2. Написать программу, выполняющую следующие действия:
 - ввод с клавиатуры данных в массив, состоящий из десяти структур типа WORKER; записи должны быть размещены по алфавиту.
 - вывод на дисплей фамилий работников, чей стаж работы в организации превышает значение, введенное с клавиатуры;
 - если таких работников нет, вывести на дисплей соответствующее сообщение.

Вариант 3

1. Описать структуру с именем TRAIN, содержащую следующие поля:
 - название пункта назначения;
 - номер поезда;
 - время отправления.
2. Написать программу, выполняющую следующие действия:

- ввод с клавиатуры данных в массив, состоящий из восьми элементов типа TRAIN; записи должны быть размещены в алфавитном порядке по названиям пунктов назначения;
- вывод на экран информации о поездах, отправляющихся после введенного с клавиатуры времени;
- если таких поездов нет, выдать на дисплей соответствующее сообщение.

Вариант 4

1. Описать структуру с именем MARSH, содержащую следующие поля:

- название начального пункта маршрута;
- название конечного пункта маршрута;
- номер маршрута.

2. Написать программу, выполняющую следующие действия:

- ввод с клавиатуры данных в массив, состоящий из восьми элементов типа MARSH; записи должны быть упорядочены по номерам маршрутов;
- вывод на экран информации о маршруте, номер которого введен с клавиатуры;
- если таких маршрутов нет, выдать на дисплей соответствующее сообщение.

Задание для контрольной работы №2

Вариант 1

Создать запись с именем Volonter, содержащую следующие поля:

- ФИО;
- день рождения;
- пол;
- заслуги;

Написать программу, выполняющую следующие действия:

- 1) Производить ввод с клавиатуры данных в массив, состоящий из N структур типа Volonter;
- 2) Упорядочить введенные данные по дню рождения;
- 3) Поиск по одному из полей; Если данных нет, то вывести сообщение об этом и попросить произвести ввод;
- 4) Выход из программы.

+ организовать программу с выводом меню на экран

+ описать вложенную запись (добрые дела: номер, дата, описание).

Вариант 2

Создать запись с именем Starosta, содержащую следующие поля:

- ФИО;
- № группы;
- заслуги;

Написать программу, выполняющую следующие действия:

- 1) Производить ввод с клавиатуры данных в массив, состоящий из N структур типа Starosta;
- 2) Упорядочить введенные данные по № группы;
- 3) Поиск по одному из полей. Если данных нет, то вывести сообщение об этом и попросить произвести ввод;
- 4) Выход из программы.

- + организовать программу с выводом меню на экран;
- + описать вложенную запись (информация о группе: кол-во человек, средний бал, название специальности).

Вариант 3

Создать запись с именем Pioneer, содержащую следующие поля:

- ФИО;
- день рождения;
- дата вступления в пионеры;
- заслуги;

Написать программу, выполняющую следующие действия:

- 1) Производить ввод с клавиатуры данных в массив, состоящий из N структур типа Pioneer (процедура);
- 2) Упорядочить введенные данные по дате вступления в пионеры;
- 3) Поиск по одному из полей. Если данных нет, то вывести сообщение об этом и попросить произвести ввод;
- 4) Выход из программы.

+ Организовать программу с выводом меню на экран

+ Описать вложенную запись (добрые дела: номер, дата, описание).

Вариант 4

Создать запись с именем Peкар, содержащую следующие поля:

- ФИО;
- день рождения;
- пол;
- квалификация;
- стаж работы;

Написать программу, выполняющую следующие действия:

- 1) Производить ввод с клавиатуры данных в массив, состоящий из N структур типа Peкар (процедура);
- 2) Упорядочить введенные данные по стажу работы;
- 3) Поиск по одному из полей. Если данных нет, то вывести сообщение об этом и попросить произвести ввод;
- 4) Выход из программы.

+ Организовать программу с выводом меню на экран;

+ Описать вложенную запись (виды хлеба и пирожных, которые умеет печь пекарь).

Задание для контрольной работы №3

Вариант 1

Написать программу с описанием класса «Student». Класс должен содержать следующие поля: фамилия, имя, отчество, пол, день, месяц и год рождения, год поступления на учебу. Класс должен содержать следующие методы: методы ввода и вывода данных о студенте, метод вычисления года окончания обучения. Предусмотреть конструктор. В основной программе продемонстрировать работу класса.

Вариант 2

Написать программу с описанием класса «Pokupatel». Класс должен содержать следующие поля: фамилия, имя, отчество, день, месяц и год рождения, общая сумма покупок. Класс должен содержать следующие методы: методы ввода и вывода данных о покупателе, метод вычисления скидки для покупателя, которая зависит от общей суммы покупок. Предусмотреть конструктор. В основной программе продемонстрировать работу

класса.

Вариант 3

Написать программу с описанием класса «Samolet». Класс должен содержать следующие поля: инвентарный номер, марку, месяц и год выпуска, рекомендуемый срок эксплуатации. Класс должен содержать следующие методы: методы ввода и вывода данных о самолете, метод вычисления даты окончания срока эксплуатации. Предусмотреть конструктор. В основной программе продемонстрировать работу класса.

Вариант 4

Написать программу с описанием класса «Rabotnik». Класс должен содержать следующие поля: фамилия, имя, отчество, день, месяц и год рождения, год поступления на работу, количество лет (на сколько был заключен договор о приеме на работу). Класс должен содержать следующие методы: методы ввода и вывода данных о работнике, метод вычисления даты окончания договора. Предусмотреть конструктор. В основной программе продемонстрировать работу класса.

Спецификации контрольных работ приведены ниже в данном комплекте ФОС.

Сводная таблица по применяемым формам и методам текущего контроля и оценки результатов обучения

Результаты обучения (освоенные умения, усвоенные знания)	Формы и методы контроля и оценки результатов обучения
Освоенные умения:	
распознавать задачу и/или проблему в профессиональном и/или социальном контексте;	Оценка правильности выполнения самостоятельной работы Решение задач во время занятия Контрольная работа №1, № 2, № 3
анализировать задачу и/или проблему и выделять её составные части;	Оценка правильности выполнения самостоятельной работы Решение задач во время занятия Контрольная работа №1, №2, № 3
определять этапы решения задачи;	Оценка правильности выполнения самостоятельной работы Решение задач во время занятия Контрольная работа №1, № 2, № 3
выявлять и эффективно искать информацию, необходимую для решения задачи и/или проблемы;	Оценка правильности выполнения самостоятельной работы Решение задач во время занятия Контрольная работа №1, № 2, № 3
составить план действия; определить необходимые ресурсы;	Оценка правильности выполнения самостоятельной работы Решение задач во время занятия Контрольная работа №1, № 2, № 3
владеть актуальными методами работы в профессиональной и смежных сферах;	Оценка правильности выполнения самостоятельной работы Решение задач во время занятия Контрольная работа №1, № 2, № 3
реализовать составленный план;	Оценка правильности выполнения самостоятельной работы Решение задач во время занятия

	Контрольная работа №1, № 2, № 3
оценивать результат и последствия своих действий (самостоятельно или с помощью наставника)	Оценка правильности выполнения самостоятельной работы Решение задач во время занятия Контрольная работа №1, № 2, № 3
определять задачи для поиска информации; определять необходимые источники информации; планировать процесс поиска;	Оценка правильности выполнения самостоятельной работы Решение задач во время занятия Контрольная работа №1, № 2, № 3
структурировать получаемую информацию;	Оценка правильности выполнения самостоятельной работы Решение задач во время занятия Контрольная работа №1, № 2, № 3
выделять наиболее значимое в перечне информации; оценивать практическую значимость результатов поиска; оформлять результаты поиска.	Оценка правильности выполнения самостоятельной работы Решение задач во время занятия Контрольная работа №1, № 2, № 3
Усвоенные знания:	
актуальный профессиональный и социальный контекст, в котором приходится работать и жить	Контрольная работа №1, №2, № 3 Устный опрос и выполнение практических заданий во время занятия
основные источники информации и ресурсы для решения задач и проблем в профессиональном и/или социальном контексте	Контрольная работа №1, №2, № 3 Устный опрос и выполнение практических заданий во время занятия
алгоритмы выполнения работ в профессиональной и смежных областях	Контрольная работа №1, №2, № 3 Устный опрос и выполнение практических заданий во время занятия
методы работы в профессиональной и смежных сферах	Контрольная работа №1, №2, № 3 Устный опрос и выполнение практических заданий во время занятия
структуру плана для решения задач	Контрольная работа №1, №2, № 3 Устный опрос и выполнение практических заданий во время занятия
порядок оценки результатов решения задач профессиональной деятельности	Контрольная работа №1, №2, № 3 Устный опрос и выполнение практических заданий во время занятия
номенклатуры информационных источников, применяемых в профессиональной деятельности;	Контрольная работа №1, №2, № 3 Устный опрос и выполнение практических заданий во время занятия
приемы структурирования информации;	Контрольная работа №1, №2, № 3 Устный опрос и выполнение практических заданий во время занятия
формат оформления результатов поиска информации.	Контрольная работа №1, №2, № 3 Устный опрос и выполнение практических заданий во время занятия

3.2 Форма промежуточной аттестации

Промежуточная аттестация по дисциплине в форме итоговой контрольной работы в 3 и 4 семестрах и в форме экзамена в 5 семестре, спецификация которого содержится в данном комплекте ФОС.

Обучающиеся допускаются к сдаче экзамена при выполнении всех видов практических и контрольных работ, предусмотренных рабочей программой и календарно-тематическим планом дисциплины.

Итоговая контрольная работа проводится за счет времени отведенного на изучение дисциплины. При условии своевременного и качественного выполнения обучающимся всех видов работ, предусмотренных рабочей программой дисциплины.

Перечень вопросов к экзамену

1. Загрузка и установка Python
2. Введение в Python. Переменные, константы. Типы данных.
3. Операторы, выражения. Ввод-вывод данных.
4. Условные выражения: операции сравнения, логические операции, оператор `in`
5. Ветвления в Python
6. Циклы в Python.
7. Строки. Базовые операции. Функции и методы строк.
8. Списки. Базовые операции. Функции и методы списков.
9. Списки. Генераторы списков. Вложенные списки.
10. Кортежи. Базовые операции.
11. Словари. Создание, базовые операции. Методы словарей.
12. Множества. Базовые операции. Генераторы множеств.
13. Функции. Передача параметров в функцию. Области видимости переменных в Python.
14. Функциональные возможности Python (лямбда-функции и т.д.)
15. Модули. Подключение функциональности модуля в глобальное пространство имен. Установка псевдонимов
16. Модули `math` и `random`
17. Модули `locale` и `decimal`
18. Работа с файлами в Python.
19. Работа с файлами формата CSV
20. Обработка значений даты и времени

4 Система оценивания комплекта ФОС текущего контроля и промежуточной аттестации

Критерии оценивания опорных конспектов.

«5» (отлично) – аккуратность выполнения, читаемость текста, грамотность (терминологическая и орфографическая), полное раскрытие темы конспекта.

«4» (хорошо) – тема конспекта раскрыта, однако материал изложен недостаточно логично; аккуратность выполнения, читаемость конспекта, грамотность (терминологическая и орфографическая).

«3» (удовлетворительно) – материал изложен недостаточно логично, неаккуратное выполнение, читаемость конспекта, грамотность (терминологическая и орфографическая), тема конспекта раскрыта не в полной мере.

«2» (неудовлетворительно) – материал изложен нелогично, допущены терминологические и орфографические ошибки, неразборчивый почерк, тема конспекта не раскрыта.

Критерии оценивания заданий практических работ.

Практическая работа оценивается максимально оценкой «5» (отлично).

Каждое задание оценивается максимально оценкой «5» (отлично).

По результатам оценивания всех заданий оценка соответствует средней.

Критерии оценивания решений задач.

«5» (отлично) – составлен правильный алгоритм решения задачи, в логическом рассуждении, в выборе нормативных источников и решении нет ошибок, получен верный ответ, задача решена рациональным способом.

«4» (хорошо) – составлен правильный алгоритм решения задачи, в логическом рассуждении и решении нет существенных ошибок; правильно сделан выбор нормативных источников; есть объяснение решения, но задача решена нерациональным способом или допущено не более двух несущественных ошибок, получен верный ответ.

«3» (удовлетворительно) – задание выполнено, в логическом рассуждении нет существенных ошибок, но допущены существенные ошибки в выборе нормативных источников; задача решена не полностью или в общем виде.

«2» (неудовлетворительно) – задача решена неправильно.

Критерии оценивания контрольной работы

Задание к контрольной работе состоит из четырех задач. Критерий оценивания следующий:

Решены все 4 задачи «отлично»,
Решены любые 3 задачи «хорошо»,
Решена любые 2 задачи «удовлетворительно»,
Решено менее 2 задач «неудовлетворительно».

Критерии оценивания ответов по экзаменационным билетам.

Экзаменационный билет оценивается максимально оценкой «5» (отлично).

В экзаменационном билете два вопроса.

Первый вопрос максимально оценивается оценкой «5» (отлично).

Второй вопрос максимально оценивается оценкой «5» (отлично).

По результатам оценивания двух вопросов оценка соответствует средней.

При оценивании практической и самостоятельной работы студента учитывается следующее:

- качество выполнения практической части работы;
- качество оформления отчета по работе;
- качество устных ответов на контрольные вопросы при защите работы.

Каждый вид работы оценивается по пяти бальной шкале.

«5» (отлично) – за глубокое и полное овладение содержанием учебного материала, в котором обучающийся свободно и уверенно ориентируется; за умение практически применять теоретические знания, высказывать и обосновывать свои суждения. Оценка «5» (отлично) предполагает грамотное и логичное изложение ответа.

«4» (хорошо) – если обучающийся полно освоил учебный материал, владеет научно-понятийным аппаратом, ориентируется в изученном материале, осознанно применяет теоретические знания на практике, грамотно излагает ответ, но содержание и форма ответа имеют отдельные неточности.

«3» (удовлетворительно) – если обучающийся обнаруживает знание и понимание основных положений учебного материала, но излагает его неполно, непоследовательно, допускает неточности, в применении теоретических знаний при ответе на практико-ориентированные вопросы; не умеет доказательно обосновать собственные суждения.

«2» (неудовлетворительно) – если обучающийся имеет разрозненные, бессистемные знания, допускает ошибки в определении базовых понятий, искажает их смысл; не может практически применять теоретические знания.

Тест оценивается по пяти бальной шкале следующим образом: стоимость каждого вопроса 1 балл. За правильный ответ студент получает 1 балл. За неверный ответ или его

отсутствие баллы не начисляются.

Оценка «5» соответствует 86% – 100% правильных ответов.

Оценка «4» соответствует 73% – 85% правильных ответов.

Оценка «3» соответствует 53% – 72% правильных ответов.

Оценка «2» соответствует 0% – 52% правильных ответов.