



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Магнитогорский государственный технический университет им. Г.И. Носова»



УТВЕРЖДАЮ

Директор ИЭиАС

В.Р. Храмшин

04.02.2025 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

**ОСНОВЫ ПРОГРАММИРОВАНИЯ НА PYTHON**

Направление подготовки (специальность)

20.03.01 Техносферная безопасность

Направленность (профиль/специализация) программы

Управление экологической и промышленной безопасностью

Уровень высшего образования - бакалавриат

Форма обучения

очная

|                     |                                                 |
|---------------------|-------------------------------------------------|
| Институт/ факультет | Институт энергетики и автоматизированных систем |
| Кафедра             | Информатики и информационной безопасности       |
| Курс                | 2                                               |
| Семестр             | 4                                               |

Магнитогорск  
2025 год

Рабочая программа составлена на основе ФГОС ВО - бакалавриат по направлению подготовки 20.03.01 Техносферная безопасность (приказ Минобрнауки России от 25.05.2020 г. № 680)

Рабочая программа рассмотрена и одобрена на заседании кафедры Информатики и информационной безопасности

03.02.2025 г., протокол № 5

Зав. кафедрой

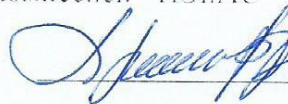


И.И. Баранкова

Рабочая программа одобрена методической комиссией ИЭиАС

04.02.2025 г., протокол № 3

Председатель



В.Р. Храмшин

Согласовано:

Зав. кафедрой Промышленной экологии и безопасности жизнедеятельности



Ю.В. Сомова

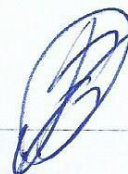
Рабочая программа составлена:  
ст. преподаватель кафедры ИиИБ



Ю.А. Мазнина

Рецензент:

Проректор по цифровизации, канд. техн. наук



К.А. Рубан

## Лист актуализации рабочей программы

---

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2026 - 2027 учебном году на заседании кафедры ПИЛОТЫ

Протокол от \_\_\_\_\_ 20\_\_ г. № \_\_\_\_  
Зав. кафедрой \_\_\_\_\_

---

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2027 - 2028 учебном году на заседании кафедры ПИЛОТЫ

Протокол от \_\_\_\_\_ 20\_\_ г. № \_\_\_\_  
Зав. кафедрой \_\_\_\_\_

---

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2028 - 2029 учебном году на заседании кафедры ПИЛОТЫ

Протокол от \_\_\_\_\_ 20\_\_ г. № \_\_\_\_  
Зав. кафедрой \_\_\_\_\_

---

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2029 - 2030 учебном году на заседании кафедры ПИЛОТЫ

Протокол от \_\_\_\_\_ 20\_\_ г. № \_\_\_\_  
Зав. кафедрой \_\_\_\_\_

### **1 Цели освоения дисциплины (модуля)**

Целью дисциплины «Основы программирования на Python» является изучение основ современных методов и средств программирования высокого уровня и формирование у обучающихся навыков их практического применения на примере языка программирования Python.

Дисциплина «Основы программирования на Python» предусматривает освоение методологий структурного, процедурного и функционального программирования, а также методов тестирования и отладки программ.

### **2 Место дисциплины (модуля) в структуре образовательной программы**

Дисциплина Основы программирования на Python входит в часть учебного плана формируемую участниками образовательных отношений образовательной программы.

Для изучения дисциплины необходимы знания (умения, владения), сформированные в результате изучения дисциплин/ практик:

Информатика

Программирование в инженерном деле

Цифровая грамотность

### **3 Компетенции обучающегося, формируемые в результате освоения дисциплины (модуля) и планируемые результаты обучения**

В результате освоения дисциплины (модуля) «Основы программирования на Python» обучающийся должен обладать следующими компетенциями:

| Код индикатора | Индикатор достижения компетенции                                                                  |
|----------------|---------------------------------------------------------------------------------------------------|
| ДПК-004-2      | Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения |
| ДПК-004-2.1    | Применяет основные алгоритмы к решению прикладных программ                                        |
| ДПК-004-2.2    | Использует системы программирования для разработки компьютерных программ                          |

#### 4. Структура, объём и содержание дисциплины (модуля)

Общая трудоемкость дисциплины составляет 3 зачетных единиц 108 акад. часов, в том числе:

- контактная работа – 36,1 акад. часов;
- аудиторная – 36 акад. часов;
- внеаудиторная – 0,1 акад. часов;
- самостоятельная работа – 71,9 акад. часов;
- в форме практической подготовки – 0 акад. час;

Форма аттестации - зачет

| Раздел/ тема дисциплины                                                                                                                                                              | Семестр | Аудиторная контактная работа (в акад. часах) |           |             | Самостоятельная работа студента | Вид самостоятельной работы                                                                                                           | Форма текущего контроля успеваемости и промежуточной аттестации | Код компетенции |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------|----------------------------------------------|-----------|-------------|---------------------------------|--------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------|-----------------|
|                                                                                                                                                                                      |         | Лек.                                         | лаб. зан. | практ. зан. |                                 |                                                                                                                                      |                                                                 |                 |
| 1. 1. Введение в Python                                                                                                                                                              |         |                                              |           |             |                                 |                                                                                                                                      |                                                                 |                 |
| 1.1 История развития языков программирования. Компиляторы и интерпретаторы. Инструментальные средства для разработки программного обеспечения. Понятие алгоритма, понятие блок-схемы | 4       |                                              |           | 2           | 4                               | Самостоятельное изучение учебной и научно литературы, работа с материалами образовательного портала и ЭБС. Подготовка к тестированию | Тестирование                                                    | ДПК-004-2.2     |
| 1.2 Знакомство с языком программирования Python: поддерживаемые парадигмы программирования, сферы применения. Интерпретатор Python. Знакомство со средой разработки                  |         |                                              |           | 1           | 2                               |                                                                                                                                      | Тестирование                                                    | ДПК-004-2.2     |
| 1.3 Литералы, переменные, типы данных, инструкции, приоритет операторов. Соглашения по оформлению кода pep8                                                                          |         |                                              |           | 1           | 2                               |                                                                                                                                      |                                                                 | ДПК-004-2.2     |
| 1.4 Работа со строками                                                                                                                                                               |         |                                              |           | 2           | 6                               |                                                                                                                                      |                                                                 | ДПК-004-2.2     |
| Итого по разделу                                                                                                                                                                     |         |                                              |           | 6           | 14                              |                                                                                                                                      |                                                                 |                 |
| 2. 2. Основные синтаксические конструкции Python                                                                                                                                     |         |                                              |           |             |                                 |                                                                                                                                      |                                                                 |                 |
| 2.1 Операторы ветвлений. Тернарный оператор.                                                                                                                                         | 4       |                                              |           | 2           | 4                               | Самостоятельное изучение                                                                                                             | Аудиторная контрольная                                          | ДПК-004-2.2     |

|                                                                |   |  |  |    |    |                                                                                                                              |                               |                          |
|----------------------------------------------------------------|---|--|--|----|----|------------------------------------------------------------------------------------------------------------------------------|-------------------------------|--------------------------|
|                                                                |   |  |  |    |    | учебной и научно литературы, работа с материалами образовательного портала и ЭБС. Подготовка к аудиторной контрольной работе | работа                        |                          |
| 2.2 Циклы                                                      | 4 |  |  | 2  | 4  | Самостоятельное изучение учебной и научно литературы, работа с материалами образовательного портала и ЭБС. Подготовка к АКР  | Аудиторная контрольная работа | ДПК-004-2.2              |
| 2.3 Функции                                                    |   |  |  | 4  | 8  | Самостоятельное изучение учебной и научно литературы, работа с материалами образовательного портала и ЭБС. Подготовка к АКР  | Аудиторная контрольная работа | ДПК-004-2.2              |
| 2.4 Стандартные библиотеки Python для вычислений: math, random |   |  |  | 2  | 4  | Самостоятельное изучение учебной и научно литературы, работа с материалами образовательного портала и ЭБС. Подготовка к АКР  | Аудиторная контрольная работа | ДПК-004-2.2              |
| Итого по разделу                                               |   |  |  | 10 | 20 |                                                                                                                              |                               |                          |
| 3. 3. Коллекции в Python: списки, кортежи, множества, словари  |   |  |  |    |    |                                                                                                                              |                               |                          |
| 3.1 Списки. Алгоритмы поиска и сортировки                      | 4 |  |  | 4  | 8  | Самостоятельное изучение учебной и научно литературы, работа с материалами образовательного портала и ЭБС.                   | Аудиторная контрольная работа | ДПК-004-2.1, ДПК-004-2.2 |

|                                                                                                        |   |  |  |    |    |                                                                                                                                                       |                               |                          |
|--------------------------------------------------------------------------------------------------------|---|--|--|----|----|-------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------|--------------------------|
|                                                                                                        |   |  |  |    |    | Подготовка к аудиторной контрольной работе                                                                                                            |                               |                          |
| 3.2 Кортежи                                                                                            |   |  |  | 1  | 2  | Самостоятельное изучение учебной и научно литературы, работа с материалами образовательного портала и ЭБС. Подготовка к аудиторной контрольной работе | Аудиторная контрольная работа | ДПК-004-2.1, ДПК-004-2.2 |
| 3.3 Множества                                                                                          | 4 |  |  | 1  | 2  | Самостоятельное изучение учебной и научно литературы, работа с материалами образовательного портала и ЭБС. Подготовка к аудиторной контрольной работе | Аудиторная контрольная работа | ДПК-004-2.1, ДПК-004-2.2 |
| 3.4 Словари                                                                                            |   |  |  | 4  | 8  | Самостоятельное изучение учебной и научно литературы, работа с материалами образовательного портала и ЭБС. Подготовка к аудиторной контрольной работе | Аудиторная контрольная работа | ДПК-004-2.1, ДПК-004-2.2 |
| Итого по разделу                                                                                       |   |  |  | 10 | 20 |                                                                                                                                                       |                               |                          |
| 4. 4. Работа с файлами в Python                                                                        |   |  |  |    |    |                                                                                                                                                       |                               |                          |
| 4.1 Принципы работы с текстовыми и бинарными файлами                                                   | 4 |  |  | 2  | 4  |                                                                                                                                                       |                               | ДПК-004-2.1, ДПК-004-2.2 |
| 4.2 Стандартные библиотеки Python для работы с файловой системой: os, sys                              |   |  |  | 2  | 4  |                                                                                                                                                       |                               | ДПК-004-2.1, ДПК-004-2.2 |
| 4.3 Стандартные библиотеки Python для работы с популярными форматами файлов (csv- и json-файлами, zip- |   |  |  | 2  | 4  |                                                                                                                                                       |                               | ДПК-004-2.1, ДПК-004-2.2 |

|                                                                                        |   |  |  |    |      |                                                                                            |       |                          |
|----------------------------------------------------------------------------------------|---|--|--|----|------|--------------------------------------------------------------------------------------------|-------|--------------------------|
| архивами)                                                                              |   |  |  |    |      |                                                                                            |       |                          |
| Итого по разделу                                                                       |   |  |  | 6  | 12   |                                                                                            |       |                          |
| 5. 5. Прочие возможности Python                                                        |   |  |  |    |      |                                                                                            |       |                          |
| 5.1 Стандартные библиотеки Python для работы с датами и временем: datetime, time       | 4 |  |  | 2  | 2    |                                                                                            |       | ДПК-004-2.1, ДПК-004-2.2 |
| 5.2 Стандартные библиотеки Python для работы с сервисами операционной системы: os, sys |   |  |  | 2  | 1    |                                                                                            |       | ДПК-004-2.1, ДПК-004-2.2 |
| Итого по разделу                                                                       |   |  |  | 4  | 3    |                                                                                            |       |                          |
| 6. 6. Подготовка к зачету                                                              |   |  |  |    |      |                                                                                            |       |                          |
| 6.1 Подготовка к промежуточной аттестации                                              | 4 |  |  |    | 2,9  | Изучение учебной и научной литературы, работа с материалами образовательного портала и ЭБС | зачет | ДПК-004-2.1, ДПК-004-2.2 |
| Итого по разделу                                                                       |   |  |  |    | 2,9  |                                                                                            |       |                          |
| Итого за семестр                                                                       |   |  |  | 36 | 71,9 |                                                                                            | зачёт |                          |
| Итого по дисциплине                                                                    |   |  |  | 36 | 71,9 |                                                                                            | зачет |                          |

## 5 Образовательные технологии

Для реализации предусмотренных видов учебной работы в качестве образовательных технологий в преподавании дисциплины «Основы программирования на Python» используются традиционная и модульно-компетентностная технологии.

Реализация компетентностного подхода предусматривает использование в учебном процессе активных и интерактивных форм проведения занятий в сочетании с внеаудиторной работой с целью формирования и развития профессиональных навыков обучающихся.

При проведении учебных занятий преподаватель обеспечивает развитие у обучающихся навыков командной работы, межличностной коммуникации, принятия решений, лидерских качеств посредством проведения интерактивных лекций, групповых дискуссий, ролевых игр, тренингов, анализа ситуаций, учета особенностей профессиональной деятельности выпускников и потребностей работодателей.

Формы учебных занятий с использованием традиционных технологий:

- обзорные лекции – для рассмотрения общих вопросов информатики и информационных технологий, для систематизации и закрепления знаний;
- информационные – для ознакомления с техническими средствами реализации информационных процессов, со стандартами организации сетей, основными приемами защиты информации, и другой справочной информацией;
- лекции-визуализации – для наглядного представления способов решения алгоритмических и функциональных задач, визуализации результатов решения задач;
- семинар – практическое занятие, посвященное освоению конкретных умений и навыков по предложенному алгоритму.

Формы учебных занятий с использованием технологий проблемного обучения:

- проблемная лекция – изложение материала, предполагающее постановку проблемных и дискуссионных вопросов, освещение различных научных подходов, авторские комментарии, связанные с различными моделями интерпретации изучаемого материала (для развития исследовательских навыков и изучения способов решения задач);
- лекции с заранее запланированными ошибками – направленные на поиск обучающимися синтаксических и алгоритмических ошибок при решении алгоритмических и функциональных задач, с последующей диагностикой слушателей и разбором сделанных ошибок;
- практическое занятие в форме практикума – организация учебной работы, направленная на решение комплексной учебно-познавательной задачи, требующей от обучающегося применения как научно-теоретических знаний, так и практических навыков;
- практическое занятие на основе кейс-метода – обучение в контексте моделируемой ситуации, воспроизводящей реальные условия научной, производственной, общественной деятельности; обучающиеся должны проанализировать ситуацию, разобраться в сути проблем, предложить возможные решения и выбрать лучшее из них; кейсы базируются на реальном фактическом материале или же приближены к реальной ситуации;
- подготовка тематических рефератов, содержащих разделы, частично или полностью выносимые на самостоятельное изучение.

Формы учебных занятий с использованием игровых технологий:

- учебная игра – форма воссоздания предметного и социального содержания будущей профессиональной деятельности специалиста, моделирования таких систем

отношений, которые характерны для этой деятельности как целого;

– деловая игра – моделирование различных ситуаций, связанных с выработкой и принятием совместных решений, обсуждением вопросов в режиме «мозгового штурма», реконструкцией функционального взаимодействия в коллективе и т.п.

Технологии проектного обучения:

– творческий проект – учебно-познавательная деятельность обучающихся осуществляется в рамках рамочного задания, подчиняясь логике и интересам участников проекта, жанру конечного результата (газета, фильм, праздник, издание, экскурсия, подготовка заданий конкурсов и т.п.);

– информационный проект – учебно-познавательная деятельность с ярко выраженной эвристической направленностью (поиск, отбор и систематизация информации о каком-то объекте, ознакомление участников проекта с этой информацией, ее анализ и обобщение для презентации более широкой аудитории).

## **6 Учебно-методическое обеспечение самостоятельной работы обучающихся**

Представлено в приложении 1.

## **7 Оценочные средства для проведения промежуточной аттестации**

Представлены в приложении 2.

## **8 Учебно-методическое и информационное обеспечение дисциплины**

### **а) Основная литература:**

1. Гниденко, И. Г. Технологии и методы программирования : учебник для вузов / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 241 с. — (Высшее образование). — ISBN 978-5-534-18130-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/581329> (дата обращения: 08.05.2025).

2. Карякин, М. И. Технологии программирования и компьютерный практикум на языке Python : учебное пособие / М. И. Карякин, К. А. Ватульян, Р. М. Мнухин ; Южный федеральный университет. - Ростов-на-Дону ; Таганрог : Издательство Южного федерального университета, 2022. - 242 с. - ISBN 978-5-9275-4108-9. - Текст : электронный. - URL: <https://znanium.com/catalog/product/2057604> (дата обращения: 08.05.2025).

3. Лаврищева, Е. М. Программная инженерия и технологии программирования сложных систем : учебник для вузов / Е. М. Лаврищева. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2025. — 432 с. — (Высшее образование). — ISBN 978-5-534-07604-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/561885> (дата обращения: 08.05.2025).

4. Маркин, А. В. Программирование на SQL : учебник и практикум для вузов / А. В. Маркин. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 805 с. — (Высшее образование). — ISBN 978-5-534-18371-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/568900> (дата обращения: 08.05.2025).

5. Федоров, Д. Ю. Программирование на python : учебное пособие для вузов / Д. Ю. Федоров. — 6-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 187 с. — (Высшее образование). — ISBN 978-5-534-19666-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/556864> (дата обращения: 08.05.2025).

6. Черткова, Е. А. Программная инженерия. Визуальное моделирование программных систем : учебник для вузов / Е. А. Черткова. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 146 с. — (Высшее образование). — ISBN 978-5-534-18197-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/562413> (дата обращения: 08.05.2025).

7. Шелудько, В. М. Основы программирования на языке высокого уровня Python : учебное пособие / В. М. Шелудько ; Южный федеральный университет. - Ростов-на-Дону ; Таганрог : Издательство Южного федерального университета, 2017. - 146 с. - ISBN 978-5-9275-2649-9. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1021662> (дата обращения: 08.05.2025).

#### **б) Дополнительная литература:**

1. Гуриков, С. Р. Основы алгоритмизации и программирования на Python : учебное пособие / С.Р. Гуриков. — Москва : ИНФРА-М, 2025. — 343 с. — (Высшее образование). - ISBN 978-5-16-020255-6. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/2166199> (дата обращения: 08.05.2025). – Режим доступа: по подписке.

2. Жуков, Р. А. Язык программирования Python: практикум : учебное пособие / Р.А. Жуков. — Москва : ИНФРА-М, 2024. — 216 с. + Доп. материалы [Электронный ресурс]. — (Высшее образование). — DOI 10.12737/textbook\_5cb5ca35aaa7f5.89424805. - ISBN 978-5-16-018516-3. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/2139862> (дата обращения: 08.05.2025). – Режим доступа: по подписке.

3. Зыков, С. В. Объектно-ориентированное программирование : учебник и практикум для вузов / С. В. Зыков. — 2-е изд. — Москва : Издательство Юрайт, 2025. — 151 с. — (Высшее образование). — ISBN 978-5-534-16941-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/561434> (дата обращения: 08.05.2025).

4. Зыков, С. В. Программирование : учебник и практикум для вузов / С. В. Зыков. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 285 с. — (Высшее образование). — ISBN 978-5-534-16031-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/560815> (дата обращения: 08.05.2025).

5. Зыков, С. В. Программирование. Функциональный подход : учебник и практикум для вузов / С. В. Зыков. — 2-е изд. — Москва : Издательство Юрайт, 2025. — 150 с. — (Высшее образование). — ISBN 978-5-534-16942-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/561744> (дата обращения: 08.05.2025).

6. Конструирование программного обеспечения : учебное пособие / под ред. Л.Г. Гагариной. — Москва : ИНФРА-М, 2024. — 319 с. — (Высшее образование). — DOI 10.12737/1893880. - ISBN 978-5-16-017861-5. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1893880> (дата обращения: 08.05.2025). – Режим доступа: по подписке.

7. Нестеров, С. А. Базы данных : учебник и практикум для вузов / С. А. Нестеров. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 258 с. — (Высшее образование). — ISBN 978-5-534-18107-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/536687> (дата обращения: 08.05.2025).

#### **в) Методические указания:**

#### **г) Программное обеспечение и Интернет-ресурсы:**

### Программное обеспечение

| Наименование ПО                         | № договора                   | Срок действия лицензии |
|-----------------------------------------|------------------------------|------------------------|
| MS Office 2007 Professional             | № 135 от 17.09.2007          | бессрочно              |
| 7Zip                                    | свободно распространяемое ПО | бессрочно              |
| Anaconda Python                         | свободно распространяемое ПО | бессрочно              |
| MS SQL Server Management Studio         | свободно распространяемое ПО | бессрочно              |
| MS Visual Studio 2017 Community Edition | свободно распространяемое ПО | бессрочно              |
| Adobe Reader                            | свободно распространяемое ПО | бессрочно              |
| Браузер Mozilla Firefox                 | свободно распространяемое ПО | бессрочно              |
| Браузер Yandex                          | свободно распространяемое ПО | бессрочно              |
| MS Office 2003 Professional             | № 135 от 17.09.2007          | бессрочно              |
| Git                                     | свободно распространяемое ПО | бессрочно              |
| MS Visual Studio Code                   | свободно распространяемое ПО | бессрочно              |
| JetBrains PyCharm Community Edition     | свободно распространяемое ПО | бессрочно              |
| NotePad++                               | свободно распространяемое ПО | бессрочно              |
| FAR Manager                             | свободно распространяемое ПО | бессрочно              |
| Double Commander                        | свободно распространяемое ПО | бессрочно              |
| Atom Editor                             | свободно распространяемое ПО | бессрочно              |

### Профессиональные базы данных и информационные справочные системы

| Название курса                                                                                       | Ссылка                                                                                              |
|------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------|
| Национальная информационно-аналитическая система – Российский индекс научного цитирования (РИНЦ)     | URL:<br><a href="https://elibrary.ru/project_risc.asp">https://elibrary.ru/project_risc.asp</a>     |
| Международная реферативная и полнотекстовая справочная база данных научных изданий «Springer Nature» | <a href="https://www.nature.com/siteindex">https://www.nature.com/siteindex</a>                     |
| Электронные ресурсы библиотеки МГТУ им. Г.И. Носова                                                  | <a href="https://host.megaprolib.net/M/P0109/Web">https://host.megaprolib.net/M/P0109/Web</a>       |
| Российская Государственная библиотека. Каталоги                                                      | <a href="https://www.rsl.ru/ru/4readers/catalogues/">https://www.rsl.ru/ru/4readers/catalogues/</a> |
| Электронная база периодических изданий East View Information Services, ООО «ИБИС»                    | <a href="https://dlib.eastview.com/">https://dlib.eastview.com/</a>                                 |

### 9 Материально-техническое обеспечение дисциплины (модуля)

Материально-техническое обеспечение дисциплины включает:

Лекционные аудитории:

- мультимедийные средства хранения, передачи и представления информации.

Учебные аудитории для проведения практических занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации:

- персональные компьютеры с ПО, выходом в Интернет и с доступом в электронную информационно-образовательную среду университета;
- мультимедийные средства хранения, передачи и представления информации;
- комплекс тестовых заданий для проведения промежуточных и рубежных контролей.

Помещения для самостоятельной работы обучающихся:

- персональные компьютеры с ПО, выходом в Интернет и с доступом в электронную информационно-образовательную среду университета.

Помещения для хранения и профилактического обслуживания учебного оборудования:

- шкафы для хранения учебно-методической документации, учебного оборудования и учебно-наглядных пособий.

### Учебно-методическое обеспечение самостоятельной работы обучающихся

Аудиторная самостоятельная работа обучающихся на практических занятиях осуществляется под контролем преподавателя в виде решения задач и выполнения упражнений, которые определяет преподаватель для обучающегося с использованием методов ИТ.

Внеаудиторная самостоятельная работа обучающихся осуществляется в виде чтения литературы по соответствующему разделу с проработкой материала и выполнения домашних заданий с консультациями преподавателя, а так же с применением кейс-технологий.

#### 1. Контрольные вопросы и задания для проведения текущего контроля

1. Особенности языка программирования Python. Соглашения по оформлению кода реп8, организация кода в модели и пакеты.
2. Язык программирования Python: числовые, булевы и строковые литералы, основные операции над простыми типами.
3. Особенности целочисленного деления в языке Python.
4. Язык программирования Python: синтаксические конструкции, условия, циклы, функция range().
5. Тернарный оператор условия, его преимущества и недостатки.
6. Язык программирования Python: системные библиотеки Python: math, time, random, os.
7. Коллекции в языке программирования Python: строки.
8. Отрицательная индексация при работе со строками.
9. Коллекции в языке программирования Python: списки, кортежи, множества.
10. Методы списков и операции со списками. Срезы списков. Присваивание в срез. Копирование списка.

11. В чем преимущество встроенного типа кортеж перед встроенным типом список? Примеры применения кортежей в языке программирования Python.

12. Коллекции в языке программирования Python: словари
13. Функции в языке программирования Python. Область видимости переменных.
14. Именованные параметры функций. Значения параметров по умолчанию.
15. Функции с переменным числом аргументов.
16. Рекурсия.
17. Принципы работы с текстовыми и бинарными файлами.
18. Стандартные библиотеки Python для работы с файловой системой: os, sys.
19. Работа с файловой системой и популярными форматами файлов: csv формат.
20. Работа с файловой системой и популярными форматами файлов: json формат.
21. Работа с файловой системой и популярными форматами файлов: zip-архивы.
22. Стандартные библиотеки Python для работы с датами и временем: datetime.
23. Стандартные библиотеки Python для работы с датами и временем: time.
24. Стандартные библиотеки Python для работы с сервисами операционной системы: os, sys.

#### 2. Примеры индивидуальных домашних заданий

1. Имя файла задается в консоли. Проведите анализ текста из файла: подсчитайте общее количество символов в тексте, количество символов без пробелов, количество слов и количество предложений в тексте. Предложения могут заканчиваться точкой, вопросительным, восклицательным знаком или троеточием. Также возможен случай, что предложение оборвалось с концом файла без заключительного знака препинания. Если файл отсутствует, выведите сообщение «Ошибка при открытии файла».

2. Дан файл с экспортом сданных решений в системе ejudge. Контекст проводился в формате личной олимпиады. Вам понадобятся следующие поля:

- User\_Id – уникальный числовой идентификатор пользователя;
- User\_Login – уникальный логин пользователя (также можно использовать, как уникальный числовой идентификатор);
- User\_Inv – у скрытого пользователя в этом поле записан символ "I";
- Lang – идентификатор языка программирования.
- Score – Количество баллов, полученных за задачу.

Посчитайте общее число участников конкурса, а также то, сколько участников использовало тот или иной язык программирования. Считается, что участник использует язык программирования, если он сдал хотя бы одно решение, набравшее больше 0 баллов. Скрытых пользователей учитывать не нужно.

Выведите общее число пользователей (Total) в первой строке, считаются только участники, сдавшие хотя бы одну задачу на ненулевой балл. После слова Total или названия языка программирования через двоеточие и пробел выведите число участников.

В последующих строках выведите все использованные языки программирования, Список упорядочите по убыванию числа участников, использующих данный язык программирования, при равных значениях – в лексикографическом порядке названия языка программирования. Пример результата работы программы:

```
Total: 19  
python3: 19  
C++: 5
```

### **3. Примеры аудиторных контрольных работ:**

1. Программа получает на вход в одной строке элементы списка – целые числа, разделенные пробелом. Количество элементов произвольное. На следующей строке вводится одно число *g* – значение поиска. Реализуйте линейный алгоритм поиска введенного значения *g*. В случае успеха выведите порядковый номер (индекс) первого найденного элемента в списке при условии, что индексация начинается с единицы. Если данный элемент отсутствует, необходимо вывести строку `ErrorValue`.

2. Реализуйте алгоритм Евклида для нахождения наибольшего общего делителя с помощью рекурсивной функции. Функция должна иметь два аргумента.

## Оценочные средства для проведения промежуточной аттестации

## а) Планируемые результаты обучения и оценочные средства для проведения промежуточной аттестации

| Код индикатора                                                                                                                                                                                                                                                                                                                            | Индикатор достижения компетенции                           | Оценочные средства                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ДПК-004-2: Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения <ul style="list-style-type: none"> <li>– ДПК-004-2.1: Применяет основные алгоритмы к решению прикладных программ</li> <li>– ДПК-004-2.2: Использует системы программирования для разработки компьютерных программ</li> </ul> |                                                            |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| ДПК-004-2.1                                                                                                                                                                                                                                                                                                                               | Применяет основные алгоритмы к решению прикладных программ | <p>Примеры практических заданий для зачета:</p> <ol style="list-style-type: none"> <li>1. Электричка отправляется в h1:m1 и едет h2:m2. Выведите время прибытия электрички на электронных часах в формате hh:mm.</li> <li>2. Напишите программу, которая находит рекордное количество вхождений (не обязательно подряд) символа в строку.</li> <li>3. На языке программирования Python реализуйте вычисление факториала через лямбда-функцию.</li> </ol> <p>Теоретические вопросы к зачету:</p> <ol style="list-style-type: none"> <li>1. Особенности языка программирования Python. Соглашения по оформлению кода pep8, организация кода в модели и пакеты.</li> <li>1. Язык программирования Python: числовые, булевы и строковые литералы, основные операции над простыми типами.</li> <li>2. Особенности целочисленного деления в языке Python.</li> <li>3. Язык программирования Python: синтаксические конструкции, условия, циклы, функция range().</li> <li>4. Тернарный оператор условия, его преимущества и недостатки.</li> <li>2. Язык программирования Python: системные библиотеки Python: math, time, random, os.</li> <li>3. Коллекции в языке программирования Python: строки.</li> <li>5. Отрицательная индексация при работе со строками.</li> <li>6. Коллекции в языке программирования Python: списки, кортежи, множества.</li> <li>7. Методы списков и операции со списками. Срезы списков. Присваивание в срез. Копирование списка.</li> <li>8. В чем преимущество встроенного типа кортеж перед встроенным типом список? Примеры применения кортежей в языке программирования Python.</li> <li>9. Коллекции в языке программирования Python: словари</li> <li>10. Функции в языке программирования Python.</li> </ol> |

| Код индикатора | Индикатор достижения компетенции                                         | Оценочные средства                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|----------------|--------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                |                                                                          | <p>Область видимости переменных.</p> <p>11. Именованные параметры функций. Значения параметров по умолчанию.</p> <p>12. Функции с переменным числом аргументов.</p> <p>13. Рекурсия.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| ДПК-004-2.2    | Использует системы программирования для разработки компьютерных программ | <p>Примеры практических заданий для зачета:</p> <p>1. Создайте текстовый файл, в котором на каждой строке написана дата в формате dd/mm/yyyy. Используя язык Python, в другой файл выведите даты в порядке возрастания в формате yyyy/mm/dd.</p> <p>2. В csv-файле даны даны рост и вес нескольких человек:<br/>164;71.5<br/>181;78.3<br/>151;52.8</p> <p>Вычислите средний рост и вес и сохраните результаты вычислений в файл формата json.</p> <p>Теоретические вопросы к зачету:</p> <p>1. Принципы работы с текстовыми и бинарными файлами.</p> <p>2. Стандартные библиотеки Python для работы с файловой системой: os, sys.</p> <p>3. Работа с файловой системой и популярными форматами файлов: csv-формат.</p> <p>4. Работа с файловой системой и популярными форматами файлов: json-формат.</p> <p>5. Работа с файловой системой и популярными форматами файлов: zip-архивы.</p> <p>6. Стандартные библиотеки Python для работы с датами и временем: datetime.</p> <p>7. Стандартные библиотеки Python для работы с датами и временем: time.</p> <p>8. Стандартные библиотеки Python для работы с сервисами операционной системы: os, sys.</p> |

#### б) Примерный перечень индивидуальных заданий

1. Имя файла задается в консоли. Проведите анализ текста из файла: подсчитайте общее количество символов в тексте, количество символов без пробелов, количество слов и количество предложений в тексте. Предложения могут заканчиваться точкой, вопросительным, восклицательным знаком или троеточием. Также возможен случай, что предложение оборвалось с концом файла без заключительного знака препинания. Если файл отсутствует, выведите сообщение «Ошибка при открытии файла».

2. Дан файл с экспортом сданных решений в системе ejudge. Контест проводился в формате личной олимпиады. Вам понадобятся следующие поля:

- User\_Id – уникальный числовой идентификатор пользователя;
- User\_Login – уникальный логин пользователя (также можно использовать, как уникальный числовой идентификатор);
- User\_Inv – у скрытого пользователя в этом поле записан символ "I";
- Lang – идентификатор языка программирования.

- Score – Количество баллов, полученных за задачу.

Посчитайте общее число участников конкурса, а также то, сколько участников использовало тот или иной язык программирования. Считается, что участник использует язык программирования, если он сдал хотя бы одно решение, набравшее больше 0 баллов. Скрытых пользователей учитывать не нужно.

Выведите общее число пользователей (Total) в первой строке, считаются только участники, сдавшие хотя бы одну задачу на ненулевой балл. После слова Total или названия языка программирования через двоеточие и пробел выведите число участников.

В последующих строках выведите все использованные языки программирования, Список упорядочите по убыванию числа участников, использующих данный язык программирования, при равных значениях – в лексикографическом порядке названия языка программирования. Пример результата работы программы:

```
Total: 19  
python3: 19  
C++: 5
```

#### **в) Порядок проведения промежуточной аттестации, показатели и критерии оценивания**

##### **Показатели и критерии оценивания зачета:**

- на оценку **«отлично»** – обучающийся должен показать высокий уровень знаний не только на уровне воспроизведения и объяснения информации, но и интеллектуальные навыки решения проблем и задач, нахождения уникальных ответов к проблемам, оценки и вынесения критических суждений;

- на оценку **«хорошо»** – обучающийся должен показать средний уровень знаний не только на уровне воспроизведения и объяснения информации, но и интеллектуальные навыки решения проблем и задач;

- на оценку **«удовлетворительно»** – обучающийся должен показать пороговый уровень знаний на уровне воспроизведения и объяснения информации, навыки решения типовых задач;

- на оценку **«неудовлетворительно»** – обучающийся не может показать знания на уровне воспроизведения и объяснения информации, не может показать навыки решения типовых задач.

### Учебно-методическое обеспечение самостоятельной работы обучающихся

Аудиторная самостоятельная работа обучающихся на практических занятиях осуществляется под контролем преподавателя в виде решения задач и выполнения упражнений, которые определяет преподаватель для обучающегося с использованием методов ИТ.

Внеаудиторная самостоятельная работа обучающихся осуществляется в виде чтения литературы по соответствующему разделу с проработкой материала и выполнения домашних заданий с консультациями преподавателя, а так же с применением кейс-технологий.

#### 1. Контрольные вопросы и задания для проведения текущего контроля

1. Особенности языка программирования Python. Соглашения по оформлению кода `per8`, организация кода в модели и пакеты.
2. Язык программирования Python: числовые, булевы и строковые литералы, основные операции над простыми типами.
3. Особенности целочисленного деления в языке Python.
4. Язык программирования Python: синтаксические конструкции, условия, циклы, функция `range()`.
5. Тернарный оператор условия, его преимущества и недостатки.
6. Язык программирования Python: системные библиотеки Python: `math`, `time`, `random`, `os`.
7. Коллекции в языке программирования Python: строки.
8. Отрицательная индексация при работе со строками.
9. Коллекции в языке программирования Python: списки, кортежи, множества.
10. Методы списков и операции со списками. Срезы списков. Присваивание в срез.

Копирование списка.

11. В чем преимущество встроенного типа кортеж перед встроенным типом список? Примеры применения кортежей в языке программирования Python.

12. Коллекции в языке программирования Python: словари
13. Функции в языке программирования Python. Область видимости переменных.
14. Именованные параметры функций. Значения параметров по умолчанию.
15. Функции с переменным числом аргументов.
16. Рекурсия.
17. Принципы работы с текстовыми и бинарными файлами.
18. Стандартные библиотеки Python для работы с файловой системой: `os`, `sys`.
19. Работа с файловой системой и популярными форматами файлов: `csv` формат.
20. Работа с файловой системой и популярными форматами файлов: `json` формат.
21. Работа с файловой системой и популярными форматами файлов: `zip`-архивы.
22. Стандартные библиотеки Python для работы с датами и временем: `datetime`.
23. Стандартные библиотеки Python для работы с датами и временем: `time`.
24. Стандартные библиотеки Python для работы с сервисами операционной системы: `os`, `sys`.

#### 2. Примеры индивидуальных домашних заданий

1. Имя файла задается в консоли. Проведите анализ текста из файла: подсчитайте общее количество символов в тексте, количество символов без пробелов, количество слов и количество предложений в тексте. Предложения могут заканчиваться точкой, вопросительным, восклицательным знаком или троеточием. Также возможен случай, что предложение оборвалось с концом файла без заключительного знака препинания. Если файл отсутствует, выведите сообщение «Ошибка при открытии файла».

2. Дан файл с экспортом сданных решений в системе ejudge. Контекст проводился в формате личной олимпиады. Вам понадобятся следующие поля:

- User\_Id – уникальный числовой идентификатор пользователя;
- User\_Login – уникальный логин пользователя (также можно использовать, как уникальный числовой идентификатор);
- User\_Inv – у скрытого пользователя в этом поле записан символ "I";
- Lang – идентификатор языка программирования.
- Score – Количество баллов, полученных за задачу.

Посчитайте общее число участников конкурса, а также то, сколько участников использовало тот или иной язык программирования. Считается, что участник использует язык программирования, если он сдал хотя бы одно решение, набравшее больше 0 баллов. Скрытых пользователей учитывать не нужно.

Выведите общее число пользователей (Total) в первой строке, считаются только участники, сдавшие хотя бы одну задачу на ненулевой балл. После слова Total или названия языка программирования через двоеточие и пробел выведите число участников.

В последующих строках выведите все использованные языки программирования, Список упорядочите по убыванию числа участников, использующих данный язык программирования, при равных значениях – в лексикографическом порядке названия языка программирования. Пример результата работы программы:

```
Total: 19  
python3: 19  
C++: 5
```

### **3. Примеры аудиторных контрольных работ**

1. Программа получает на вход в одной строке элементы списка – целые числа, разделенные пробелом. Количество элементов произвольное. На следующей строке вводится одно число *g* – значение поиска. Реализуйте линейный алгоритм поиска введенного значения *g*. В случае успеха выведите порядковый номер (индекс) первого найденного элемента в списке при условии, что индексация начинается с единицы. Если данный элемент отсутствует, необходимо вывести строку `ErrorValue`.

2. Реализуйте алгоритм Евклида для нахождения наибольшего общего делителя с помощью рекурсивной функции. Функция должна иметь два аргумента.

## Оценочные средства для проведения промежуточной аттестации

## а) Планируемые результаты обучения и оценочные средства для проведения промежуточной аттестации

| Код индикатора                                                                                                                                                                                                                                                                                                                            | Индикатор достижения компетенции                           | Оценочные средства                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ДПК-004-2: Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения <ul style="list-style-type: none"> <li>– ДПК-004-2.1: Применяет основные алгоритмы к решению прикладных программ</li> <li>– ДПК-004-2.2: Использует системы программирования для разработки компьютерных программ</li> </ul> |                                                            |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| ДПК-004-2.1                                                                                                                                                                                                                                                                                                                               | Применяет основные алгоритмы к решению прикладных программ | <p>Примеры практических заданий для зачета:</p> <ol style="list-style-type: none"> <li>1. Электричка отправляется в h1:m1 и едет h2:m2. Выведите время прибытия электрички на электронных часах в формате hh:mm.</li> <li>2. Напишите программу, которая находит рекордное количество вхождений (не обязательно подряд) символа в строку.</li> <li>3. На языке программирования Python реализуйте вычисление факториала через лямбда-функцию.</li> </ol> <p>Теоретические вопросы к зачету:</p> <ol style="list-style-type: none"> <li>4. Особенности языка программирования Python. Соглашения по оформлению кода pep8, организация кода в модели и пакеты.</li> <li>14. Язык программирования Python: числовые, булевы и строковые литералы, основные операции над простыми типами.</li> <li>15. Особенности целочисленного деления в языке Python.</li> <li>16. Язык программирования Python: синтаксические конструкции, условия, циклы, функция range().</li> <li>17. Тернарный оператор условия, его преимущества и недостатки.</li> <li>5. Язык программирования Python: системные библиотеки Python: math, time, random, os.</li> <li>6. Коллекции в языке программирования Python: строки.</li> <li>18. Отрицательная индексация при работе со строками.</li> <li>19. Коллекции в языке программирования Python: списки, кортежи, множества.</li> <li>20. Методы списков и операции со списками. Срезы списков. Присваивание в срез. Копирование списка.</li> <li>21. В чем преимущество встроенного типа кортеж перед встроенным типом список? Примеры применения кортежей в языке программирования Python.</li> <li>22. Коллекции в языке программирования Python: словари</li> <li>23. Функции в языке программирования Python.</li> </ol> |

| Код индикатора | Индикатор достижения компетенции                                         | Оценочные средства                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|--------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                |                                                                          | <p>Область видимости переменных.</p> <p>24. Именованные параметры функций. Значения параметров по умолчанию.</p> <p>25. Функции с переменным числом аргументов.</p> <p>26. Рекурсия.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| ДПК-004-2.2    | Использует системы программирования для разработки компьютерных программ | <p>Примеры практических заданий для зачета:</p> <p>3. Создайте текстовый файл, в котором на каждой строке написана дата в формате dd/mm/yyyy. Используя язык Python, в другой файл выведите даты в порядке возрастания в формате yyyy/mm/dd.</p> <p>4. В csv-файле даны даны рост и вес нескольких человек:</p> <p>164;71.5<br/>181;78.3<br/>151;52.8</p> <p>Вычислите средний рост и вес и сохраните результаты вычислений в файл формата json.</p> <p>Теоретические вопросы к зачету:</p> <p>9. Принципы работы с текстовыми и бинарными файлами.</p> <p>10. Стандартные библиотеки Python для работы с файловой системой: os, sys.</p> <p>11. Работа с файловой системой и популярными форматами файлов: csv-формат.</p> <p>12. Работа с файловой системой и популярными форматами файлов: json-формат.</p> <p>13. Работа с файловой системой и популярными форматами файлов: zip-архивы.</p> <p>14. Стандартные библиотеки Python для работы с датами и временем: datetime.</p> <p>15. Стандартные библиотеки Python для работы с датами и временем: time.</p> <p>16. Стандартные библиотеки Python для работы с сервисами операционной системы: os, sys.</p> |

#### б) Примерный перечень индивидуальных заданий

1. Имя файла задается в консоли. Проведите анализ текста из файла: подсчитайте общее количество символов в тексте, количество символов без пробелов, количество слов и количество предложений в тексте. Предложения могут заканчиваться точкой, вопросительным, восклицательным знаком или троеточием. Также возможен случай, что предложение оборвалось с концом файла без заключительного знака препинания. Если файл отсутствует, выведите сообщение «Ошибка при открытии файла».

2. Дан файл с экспортом сданных решений в системе ejudge. Контекст проводился в формате личной олимпиады. Вам понадобятся следующие поля:

- User\_Id – уникальный числовой идентификатор пользователя;
- User\_Login – уникальный логин пользователя (также можно использовать, как уникальный числовой идентификатор);
- User\_Inv – у скрытого пользователя в этом поле записан символ "I";
- Lang – идентификатор языка программирования.

- Score – Количество баллов, полученных за задачу.

Посчитайте общее число участников конкурса, а также то, сколько участников использовало тот или иной язык программирования. Считается, что участник использует язык программирования, если он сдал хотя бы одно решение, набравшее больше 0 баллов. Скрытых пользователей учитывать не нужно.

Выведите общее число пользователей (Total) в первой строке, считаются только участники, сдавшие хотя бы одну задачу на ненулевой балл. После слова Total или названия языка программирования через двоеточие и пробел выведите число участников.

В последующих строках выведите все использованные языки программирования, Список упорядочите по убыванию числа участников, использующих данный язык программирования, при равных значениях – в лексикографическом порядке названия языка программирования. Пример результата работы программы:

```
Total: 19  
python3: 19  
C++: 5
```

#### **в) Порядок проведения промежуточной аттестации, показатели и критерии оценивания**

##### **Показатели и критерии оценивания зачета:**

- «зачтено» – обучающийся должен показать пороговый уровень знаний на уровне воспроизведения и объяснения информации, навыки решения типовых задач;
- «не зачтено» – обучающийся демонстрирует знания не более 20% теоретического материала, допускает существенные ошибки, не может показать интеллектуальные навыки решения простых задач.