

## Указания по выполнению курсовой работы по дисциплине «Языки программирования»

**Цель работы:** закрепить на практике навыки объектно-ориентированного проектирования программного обеспечения, а также использования различных средств языка Python для разработки прикладного программного обеспечения.

### **Исходные данные к работе**

1. Индивидуальное задание выбирается из списка, приведенного в данном файле.

2. Язык программирования Python.

3. Рекомендуемая среда разработки не регламентируется и выбирается самостоятельно.

4. Вид приложения – объектно-ориентированное консольное.

5. Хранение данных в памяти должно быть организовано посредством классов (объектно-ориентированная база данных).

6. Физически данные должны храниться в файлах. При запуске программы происходит загрузка данных из файла в массив, либо другую структуру данных, компонентами которой являются объекты. Перед завершением работы программы, либо при выборе соответствующего пункта меню происходит сохранение данных в текстовый файл (перезапись исходного файла данных).

7. Построение кода должно соответствовать правилам: PEP 8.

### **Требования к программной разработке**

В результате должен быть разработан комплекс программных модулей, позволяющих осуществлять обработку данных согласно варианту задания.

Программное обеспечение основано на работе с файлами в консольном режиме. Использование СУБД приветствуется!

Внизу программных форм должна выводиться информация о разработчике, год разработки.

Комплекс программных средств включает две прикладных программы: «Администратор» и «Пользователь».

**Программа «Администратор»** должна обеспечивать выполнение следующих функциональных возможностей:

1. Управление учетными записями пользователей:

- просмотреть все учетные записи;
- добавить учетную запись;
- отредактировать учетную запись;
- удалить учетную запись.

2. Работа с данными:

а) режим редактирования:

- просмотреть все данные;
- добавить запись;
- удалить запись (для этого необходимо ввести порядковый номер

конкретной записи);

- редактировать запись (для этого необходимо ввести порядковый номер конкретной записи).

**Программа «Пользователь»** должна обеспечивать выполнение следующих функциональных возможностей:

- просмотреть все данные;
- выполнить задачу (задачи), указанную в индивидуальном задании;
- выполнить поиск данных;
- выполнить сортировку по различным полям в алфавитном порядке / в порядке убывания.

После авторизации пользователю должно быть выведено меню, из которого, путем ввода номера соответствующего пункта меню, пользователь переходит к выполнению нужной ему операции. Примеры пользовательских интерфейсов приведены на рисунках 1-3. (Они примерные – можно предложить свои варианты)



Рисунок 1 – Пример интерфейса для авторизации и меню для пользователя

```
----- Поиск студента -----
ФИО искомого студента Петр
Информация о найденных студентах
-----
| № | ФИО | № гр. | Матем. | Физ | Инф. | Ср.б. | Актив. | Стипендия |
-----
| 1 | Петров | 12 | 4 | 5 | 5 | 4.67 | 1 | 7500.00 |
-----
| 2 | Петрова | 21 | 5 | 3 | 5 | 4.33 | 0 | 0.00 |
-----
```

Рисунок 2 – Пример интерфейса для функциональной возможности «выполнить поиск»

```

-----Меню сортировки -----
1. По алфавиту;
2. По среднему баллу;
3. По стипендии
0. Назад.
1
Сортированная информация
-----
| № | ФИО | № гр. | Матем. | Физ | Инф. | Ср.б. | Актив. | Стипендия |
-----
| 1 | Антонов | 21 | 3 | 5 | 5 | 4.33 | 1 | 5000.00 |
-----
| 2 | Иванов | 12 | 5 | 5 | 5 | 5.00 | 0 | 6250.00 |
-----
| 3 | Петров | 12 | 4 | 5 | 5 | 4.67 | 1 | 7500.00 |
-----
| 4 | Петрова | 21 | 5 | 3 | 5 | 4.33 | 0 | 0.00 |
-----
| 5 | Сидоров | 11 | 4 | 5 | 4 | 4.33 | 1 | 7500.00 |
-----
| 6 | Чернов | 23 | 4 | 4 | 5 | 4.33 | 0 | 6250.00 |
-----

```

Рисунок 3 – Пример интерфейса для функциональной возможности «выполнить сортировку»

### **Предусмотреть:**

- обработку исключительных ситуаций (файл с данными не найден, ввод некорректных данных) данный пункт может быть дополнен по согласованию с руководителем работы;

- в случае отсутствия результатов поиска должно быть выведено сообщение «По Вашему запросу ничего не найдено.».

- запрос на подтверждение удаления вида «Вы действительно хотите удалить файл (запись)?»;

- вывод сообщения о результате выполнения каждой из операций (например, «Запись успешно создана»). Перечень выводимых сообщений должен быть согласован с руководителем работы.

### **Работа меню должна быть реализована в виде класса.**

Каждая курсовая работа, включая программную разработку, должен быть уникальной, т.к. подразумевает индивидуальную работу над своим заданием.

#### При выставлении отметки за курсовую работу учитывается:

- 1) Качество разработанной проектной, а также сопроводительной документации (пояснительная записка)
- 2) Объем реализованных функциональных требований
- 3) Качество работы представляемого ПО (наличие ошибок и неправильных действий во время работы)
- 4) Качество используемых алгоритмов
- 5) Качество оформления программного кода
- 6) Ответы на вопросы преподавателя на защите

## Структура пояснительной записки

Титульный лист

Содержание

Введение

1. Теоретическое исследование предметной области

2. Анализ и разработка программного обеспечения

2.1 Анализ предметной области и формирование функциональных требований

2.2 Разработка структуры и алгоритмов работы модулей программы

2.3. Описание программного кода

3. Демонстрация работы

3.1. Руководство пользователя

3.2. Тестирование программы

Заключение

Библиографический список

Приложения

А Листинг программного кода

**Титульный лист** является первой страницей курсовой работы (Приложение 1).

**Задание на курсовую работу** (Приложение 2).

**Содержание** включает в себя все заголовки курсовой работы с указанием страниц каждого раздела, подраздела.

Во **введении** отражается объект, предмет, цель и задачи, актуальность работы, обосновывается новизна, теоретическая и практическая значимость работы, приводятся сведения о состоянии изучаемой проблемы.

Цель должна быть сформулирована с точки зрения предметной области. Записать в качестве цели закрепление навыков программирования на языке Python **нельзя**.

**Теоретическое исследование предметной области.** Здесь дается полный ответ на теоретический вопрос (согласно Вашему варианту задания).

**Анализ предметной области и формулирование функциональных требований.** Здесь приводится полный текст Вашего варианта задания. Четко обозначается предметная область, кратко описываются правила и закономерности ее функционирования, на основании которых позднее формулируются функциональные требования к разрабатываемому ПО.

**Разработка структуры программы** подразумевает перечисление и краткое описание свойств и методов для классов. Код методов **НЕ ПРИВОДИТСЯ**, т.к. на этом этапе он еще не существует.

**Разработка алгоритмов работы модулей программы** включает блок-схемы алгоритмов с описанием их работы для **трех** пользовательских функций. Обратите внимание, что так как алгоритмы разрабатываются до непосредственного кодирования программы, то они **не могут содержать просто копии строчек кода**. Алгоритм может содержать словесные инструкции с

упоминанием имен структур, массивов/векторов, модулей, описанных выше.

**Описание программного кода** дается краткое словесное описание основных модулей, методов и т.д., разработанной программы.

**Руководство пользователя** подразумевает подробное пошаговое описание работы программы.

**Тестирование программы** здесь приводятся тестовые случаи работы программы, проиллюстрированные программными формами консоли. Допускается помещать программные формы в отдельное приложение.

**В заключении** прописываются выводы о проделанной работе. Что было конкретно сделано студентом. Далее можно перечислить основные функциональные возможности, область применения, а также слабые места и пути дальнейшего развития ПО. Текст должен быть составлен не общими фразами, а конкретно в привязке в вашему варианту задания. Объем 1 страница. **Писать, что работа выполнена в полном объеме в соответствии с требованиями нельзя. В качестве достоинств разработанного ПО нельзя писать про качественный интерфейс.**

**Библиографический список.** Приводится список печатных, а также интернет-ресурсов, используемых при выполнении работы. Оформляется в соответствии с требованиями стандарта ГОСТ Р 7.0.100–2018 «Библиографическая запись. Библиографическое описание. Общие требования и правила составления».

*Пример:*

1. Абдрахманов, М. И. Язык программирования Python : учебное пособие / М. И. Абдрахманов. – Москва : Ай Пи Ар Медиа, 2023. – 171 с.

2. Сузи, Р. А. Язык программирования Python : учебное пособие / Р. А. Сузи. – 4-е изд. – Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2024. – 350 с. – Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. – URL: <https://www.iprbookshop.ru/142310.html> (дата обращения: 12.08.2024).

**Приложения.** Приводится программный код.

**Требования по оформлению:**

шрифт Times New Roman, размер шрифта – 14 кегль;

межстрочный интервал – 1,5;

до и после абзаца дополнительный интервал не устанавливается;

абзацный отступ – 1,25 см;

поля: левое – 30 мм, правое – 15 мм, верхнее – 20 мм, нижнее – 20 мм;

выравнивание – по ширине страницы.

Переносы слов в текстовом документе не допускаются.

Страницы документа нумеруются арабскими цифрами (нумерация сквозная по всему тексту). Номер страницы ставится в центре нижнего поля листа без точки. Тип шрифта – Times New Roman. Размер шрифта (кегель) – 14. Титульный лист включается в общую нумерацию, номер страницы на нем не ставится. Номера страниц проставляются начиная с ВВЕДЕНИЯ.

## Варианты заданий к курсовой работе

### Вариант 1

1. Библиотеки для создания нейронных сетей на Python: общее описание библиотек Theano, TensorFlow, Keras, Lasagne, Neon, Kayak

### 2. Разработка программы учета больных вирусом COVID-19

Программа предоставляет сведения о больных COVID-19, которые находятся на карантине: ФИО больного; количество дней, находящихся на карантине; назначенное лечение; дата заболевания.

Вывести в алфавитном порядке список больных COVID-19, находящиеся на больничном более x дней (x вводится с клавиатуры).

### Вариант 2

1. Графики. Модуль matplotlib: Простые графики, Заголовок, подписи, сетка, легенда, Несколько графиков на одном полотне, Гистограммы, диаграммы-столбцы, Круговые и контурные диаграммы, Трёхмерные графики.

### 2. Разработка программы расселения детей в лагере

Для получения места формируется список детей, который включает: Ф.И.О. ребенка; номер группы (шесть цифр); средний балл; участие в общественной деятельности; доход на одного члена семьи.

Вывести список детей, проживающих в определённой комнате (номер комнаты вводится с клавиатуры).

### Вариант 3

1. Дата и время в Python: получение текущей даты и времени, форматирование даты и времени, модуль calendar, функция sleep, измерение времени выполнения фрагментов кода, модуль datetime.

### 2. Разработка программы расписания движения самолетов

В аэропорту хранится расписание самолётов. Для каждого рейса указаны: номер рейса; пункт назначения; время отправления; время прибытия на конечный пункт.

Вывести информацию о всех рейсах, которыми можно воспользоваться для прибытия в пункт назначения не позднее чем за 12 часов до заданного времени (интересующее время прибытия вводится с клавиатуры).

### Вариант 4

1. Документирование проекта: рекомендации относительно написания технической документации, строки документации в python, языки разметки для документации, популярные генераторы документации (использование sphinx, использование mkdocs).

### 2. Разработка программы продажи билетов на самолёт

Аэропорт осуществляет продажу билетов на различные рейсы. Для каждого рейса указаны: номер рейса; пункт назначения; дату отправления; время отправления; время прибытия на конечный пункт; стоимость одного билета; количество оставшихся для продажи билетов; количество проданных билетов.

Обеспечить функциональную возможность покупки билетов на конкретный рейс в требуемом количестве (при этом количество оставшихся для продажи билетов и количество проданных билетов обновляется).

### Вариант 5

1. Итераторы и генераторы: ручное использование итератора, делегирование итерации, создание нового шаблона итерации с помощью генераторов, реализация протокола итератора, итерация в обратном направлении, экстрасостояние функции-генератора, пропуск первой части итерируемого, итерирование по всем возможным комбинациям или перестановкам.

## **2. Разработка программы учета переговоров работников такси**

Оператор сотовой связи хранит информацию о разговорах своих абонентов: номер абонента; ФИО абонента; указание принадлежности вызова к исходящему или входящему; номер исходящего или входящего вызова; дата звонка; время звонка; продолжительность разговора.

Вывести по каждому сотруднику такси за требуемый период времени: перечень входящих и исходящих вызовов, общее время входящих вызовов, общее время исходящих вызовов, общую сумму на исходящие вызовы (требуемый период времени вводится с клавиатуры).

### **Вариант 6**

1. Клиент-серверное программирование в python: протоколы, сокет, порты и адреса, лексика, модуль socket, подключение сервера, общие методы.

### **2. Разработка программы расчета заработной платы сотрудников Яндекс Такси**

Сведения о сотрудниках Яндекс Такси содержат: Ф.И.О. сотрудника; табельный номер; год; месяц; количество проработанных часов за неделю; количество выполненных заказов.

Вычеты складываются из:

Комиссии сервиса – 23% от суммы заказа.

Комиссии таксопарка – варьирует в пределах 3-30%.

Определить объем выплат конкретному сотруднику за требуемый период времени (требуемый период времени вводится с клавиатуры).

### **Вариант 7**

1. Контроль кода: введение в системы контроля версиями, знакомство с GIT, установка GIT, основы работы с GIT, основные понятия GIT, управление файлами

### **2. Разработка программы учета проживающих в общежитии**

Для проживающих в общежитии, задаются: ФИО; год заселения; комната; замечания; год выселения; причина выселения, отметка о проживании текущий момент (если еще проживает в общежитии, если не проживает в общежитии указать, что выселен в таком-то году).

Вывести список комнат с ФИО в алфавитном порядке, заселенных после заданного года (год вводится с клавиатуры). Вывести список проживающих, в текущий момент.

### **Вариант 8**

1. Кортежи: понятие кортежа, создание кортежей, методы кортежей, перебор элементов кортежа, кортеж как условие. функция len() и оператор in, неизменность кортежей и слияния, модуль itertools, распаковка кортежа в отдельные переменные, списки vs кортежи.

### **2. Разработка программы учета выполненных заказов Яндекс такси**

Сведения о выполненных заказах включают: дату; номер заказа; точка отправления; точка прибытия; количество выполненных заказов; ФИО водителя в данный день.

Для заданного водителя Яндекс такси необходимо вывести количество выполненных заказов за требуемый период времени (требуемый период времени вводится с клавиатуры).

### **Вариант 9**

1. Массивы. Модуль numpy: Создание и индексация массивов, Арифметические операции и функции с массивами, Двумерные массивы, форма массивов, срезы.

### **2. Разработка программы учета научной деятельности сотрудников**

Информация о сотрудниках содержит: Ф.И.О. сотрудника; ученую степень; название кафедры; должность; дату начала работы; количество публикаций.

Вывести список сотрудников занимающиеся активной научной деятельностью. Вывести список сотрудников в порядке убывания количества публикаций.

### **Вариант 10**

1. **Метапрограммирование:** введение в метапрограммирование, декораторы, метаклассы (введение в метаклассы, пользовательские метаклассы, использование метаклассов вместо функций), генерация кода.

#### **2. Разработка программы автоматизации учета книг библиотеки**

Сведения о книгах содержат следующую информацию: автор книги, наименование, издательство, город издания, год издания, код ББК, код ISBN, фамилия, имя отчество читателя, дата получения книги и дата возврата.

Вывести список книг (наименование, автор, фамилия и имя читателя, дата возврата) заданного автора, полученных читателями за 2023 год.

### **Вариант 11**

1. **Многозадачность в Python:** многопоточность, кейсы, подходящие для использования многопоточности, модуль `thread`, модуль `threading`, синхронизация потоков, многопоточная приоритетная очередь.

#### **2. Разработка программы учета клиентов салона красоты**

Информация о клиентах: Ф.И.О. клиента, дата посещения, номер телефона, мастер Ф.И.О., дата следующей записи, процедура.

Вывести список и подсчитать общее количество постоянных клиентов определенного мастера, Ф.И.О., которого будет введена с клавиатуры.

### **Вариант 12**

1. **Множества и словари:** понятие словаря, различные операции над словарями, доступ к элементу, добавление и удаление элементов словаря, перебор элементов словаря, сортировка словаря, методы `keys()`, `values()`, понятие множества, операции над множествами, методы множеств.

#### **2. Разработка программы ассортимента машин в таксопарке**

Сведения об ассортименте в таксопарке включают: название марки машины, цена, изготовитель, количество, год выпуска, пробег машины.

Вывести список машин, которые подходят по цене, где  $x$  цена ( $x$  вводится с клавиатуры).

### **Вариант 13**

1. **Модули и пакеты:** понятие модуля, инструкция `import`, инструкция `from`, путь поиска модулей, повторная загрузка модулей, EGG-файлы, разделение модуля на несколько файлов, создание отдельных каталогов импорта кода под общим пространством имен, перезагрузка модулей, добавление каталогов в `sys.path`, распространение пакетов.

#### **2. Разработка программы подбора материалов для ремонта квартиры**

Сведения включают: артикул, наименование, цена, изготовитель, размер, количество.

Вывести список материалов артикула  $x$  и размера  $y$  ( $x$ ,  $y$  вводятся с клавиатуры). Осуществить расчет приобретения необходимого количества материалов для ремонта.

### **Вариант 14**

1. **Обработка исключений:** понятие исключений, типы исключений, инструкция `try..except..else..finally`, инструкция `with .. as`, генерирование исключений.

#### **2. Разработка программы учета выполнения работы лаборанта**

На кафедре хранится информация обо всех заявках: наименование ремонтируемого изделия (компьютеры и т.д.), инвентарный номер изделия, кабинет, Ф.И.О. заявителя, дата написания заявки, дата выполнения, статус (выполнен или нет).

Вывести список заявок, невыполненных на текущий момент (сначала – просроченные, затем – ожидающие выполнения в плановом порядке). Вывести общий объем выполненных заявок от даты  $x$  до даты  $y$  ( $x$ ,  $y$  вводятся с клавиатуры).



### **Вариант 15**

1. ООП и Python: основы объектно-ориентированного программирования, определение класса и создание объекта, конструктор и деструктор, наследование, инкапсуляция, полиморфизм.

#### **2. Разработка программы учета слушателей переподготовки**

Сведения об успеваемости слушателя содержат следующую информацию: номер группы (шесть цифр), Ф.И.О. слушателя, сведения о зачетах (зачет/незачет), оценки по экзаменам, рефератам, ОКР.

Вывести всех слушателей в порядке убывания количества задолженностей (количество задолженностей по каждому слушателю необходимо указать); средний балл, полученный каждым слушателем группы  $x$  (вводится с клавиатуры), и всей группой в целом.

### **Вариант 16**

1. ООП и Python: специальные методы, статические методы, абстрактные методы, перегрузка операторов, свойства класса, декораторы класса, композиция.

#### **2. Разработка программы учета сведений о музыкальном конкурсе**

Информация о конкурсе включает: Ф.И.О участника; год рождения; название страны; наименование музыкального инструмента (гитара, фортепиано, скрипка, виолончель и др.); занятое место по результатам конкурса.

По каждому классу музыкальных инструментов вывести первые три места с указанием возраста победителей. Вывести список самых молодых (до 12 лет) победителей конкурса в порядке увеличения возраста.

### **Вариант 17**

1. Операторы в Python: математические операторы и работа с числами, операторы для работы с последовательностями, операторы присваивания, двоичные операторы, приоритет выполнения операторов.

#### **2. Разработка программы учета сведений о пациентах медицинского центра**

Сведения о пациентах медицинского центра содержат: Ф.И.О. пациента, пол, дату рождения, место проживания (город), контактный телефон, диагноз.

Вывести иногородних пациентов. Вывести список пациентов старше  $x$  лет, у которых диагноз  $y$  ( $x, y$  вводятся с клавиатуры).

### **Вариант 18**

1. Переменные и типы данных в Python: имена переменных, типы данных, присваивание значений, проверка типа данных и приведение типов, удаление переменной.

#### **2. Разработка программы продажи железнодорожных билетов**

Железнодорожный вокзал осуществляет продажу билетов на поезда дальнего следования. Для каждого поезда указаны: номер поезда; пункт назначения; дату отправления; время отправления; время прибытия на конечный пункт; стоимость одного билета; количество оставшихся для продажи билетов; количество проданных билетов.

Обеспечить функциональную возможность покупки билетов на конкретный рейс в требуемом количестве (при этом количество оставшихся для продажи билетов и количество проданных билетов обновляется). Вывести номер, время отправления и наличие билетов для поездов, прибывающих в город  $x$  в интервале от  $a$  до  $b$  ( $x, a, b$  вводятся с клавиатуры).

### **Вариант 19**

1. Пользовательские функции: объявление функции, необязательные параметры функции, переменное число параметров, анонимные функции, функции-генераторы, декораторы, рекурсия, глобальные и локальные переменные, документирование функций, возвращаем несколько значений, именованные аргументы.

## **2. Разработка программы планирования факультативных учебных дисциплин для студентов**

Для формирования факультативных занятий необходимо обработать информацию следующего вида: Ф.И.О. студента, номер группы, средний балл успеваемости, пять возможных для факультативного посещения дисциплин. Выбираемая дисциплина помечается 1, не выбираемая – 0. Перечень предлагаемых дисциплин: математика, физика, программирование, английский язык, базы данных.

Вывести список и общее количество студентов, желающих прослушать дисциплину х. Если число желающих больше 15, то отобразить 15 студентов с более высоким баллом успеваемости. Вывести предлагаемые дисциплины в порядке убывания популярности с указанием общего числа записавшихся на каждую из них.

### **Вариант 20**

1. Программная реализация элементов нейронной сети на Python: элементы искусственного интеллекта (основные понятия и определения искусственного интеллекта, искусственный нейрон как основа нейронных сетей, функция единичного скачка, сигмоидальная функция активации, гиперболический тангенс), нейронные сети (однослойные нейронные сети, многослойные нейронные сети, обучение нейронных сетей), создание нейронных блоков, пример работы с нейронами в python, сбор нейронов в нейросеть, прямое распространение feedforward.

### **2. Разработка программы учета сведений об игроках хоккейной команды**

Сведения об игроках хоккейной команды включают: Ф.И.О. игрока, дату рождения, количество сыгранных матчей, число заброшенных шайб, количество голевых передач, количество штрафных минут.

Вывести 6 лучших игроков (голы+передачи) с указанием их результативности.

### **Вариант 21**

1. Работа с интернетом: разбор URL-адреса, декодирование строки запроса, разбор HTML-эквивалентов, преобразование относительных ссылок, определение кодировки, реализация http-клиента.

### **2. Разработка программы продажи авиабилетов**

Авиакомпания осуществляет продажу билетов на самолеты с указанием: номера рейса; типа самолета; пункта назначения; дату вылета; время вылета; время прилета; вместимости самолета; количества оставшихся билетов бизнес-класса; стоимости билета бизнес-класса; количества оставшихся билетов эконом-класса; стоимости билета эконом-класса.

Обеспечить функциональную возможность покупки билетов на конкретный рейс в требуемом количестве (при этом количество оставшихся для продажи билетов обновляется). Если на интересующий рейс нет билетов требуемого класса, то при наличии билетов другого класса на этот рейс – вывести соответствующее информационное сообщение (например, «Билетов эконом-класса на данный рейс нет в наличии, но имеются билеты бизнес-класса в количестве 10.»).

### **Вариант 22**

1. Работа с файлами в Python: открытие файла на чтение и запись в Python, чтение файла, запись в файл на Python, управление указателем, работа с файловой системой.

### **2. Разработка программы учета автомобилей автопарка**

Таксопарк содержит информацию об имеющихся автомобилях: марка автомобиля, год выпуска, расход топлива, стоимость автомобиля, количество автомобилей данной марки.

Подсчитать общую стоимость таксопарка. Подсчитать общую стоимость автомобилей каждого вида.

### **Вариант 23**

1. Разработка web-приложений на Python: основы динамических веб-сайтов (взаимодействие: HTTP, URL, запросы, ответы, хранилища данных: SQL и реляционные базы данных, представление: шаблоны отображения в разметку HTML и в другие форматы), понимание моделей, представлений и шаблонов, общий обзор архитектуры Django.

#### **2. Разработка программы расчета стипендии**

Для расчета стипендии необходимо обработать информацию следующего вида: номер группы, Ф.И.О. студента, форма обучения (платная/бюджетная), количество пропусков (их должно быть не больше 10 часов), зачеты по пяти предметам (зачет/незачет), оценки по 4 предметам по 10 бальной системе.

Рассчитать стипендию для студентов-бюджетников. При этом:

1500 руб. – если средний балл успеваемости от 5,0 по 5,9;

1700 руб. – если средний балл успеваемости от 6,0 по 7,9;

2000 руб. – если средний балл успеваемости от 8,0 по 8,9;

2500 руб. – если средний балл успеваемости от 9,0 по 10,0.

При вводе номера группы появляется вся информация о стипендиях студентов и о студентах.

### **Вариант 24**

1. Разработка приложений с графическим интерфейсом: создание окна в tkinter, разметка виджетов в tkinter — pack, grid и place, виджеты button, checkbutton, label, scale и listbox, меню, подменю и панель инструментов в tkinter, диалоговые окна в tkinter.

#### **2. Разработка программы учета выполнения работы мастера по работе с волосами**

Необходимо обработать следующую информацию: ФИО мастера, тип работы, номер клиента, стоимость, дата.

Вывести лучшего сотрудника за рабочий день, рабочий день вводится с клавиатуры.

### **Вариант 25**

1. Регулярные выражения: введение в регулярные выражения, функция compile() и основы регулярных выражений, методы match() и search(), метод findall(), metoasub()..

#### **2. Разработка программы планирования бюджета при ремонте машины**

Необходимо обработать информацию следующего вида: номер машины, марка машины, год машины, Ф.И.О. сотрудника, перечень работ, количество часов, стоимость одного часа.

Для каждого мастера вывести его итоговую стоимость, перечень видов работ с указанием общего количества задействованных специалистов и стоимости данного этапа.

### **Вариант 26**

1. Списки: понятие списка, операции над списками, многомерные списки, проход по элементам списка, поиск элемента в списке, добавление и удаление элементов в списке, перемешивание элементов и выбор случайного элемента, сортировка списка, преобразование списка в строку, вычисления с большими числовыми массивами

#### **2. Разработка программы расчета оплаты лечения**

Сведения о клиентах имеют следующий вид: Ф.И.О.; год; месяц; наименование услуги.

Для месяца x года у вывести список клиентов. Вывести сумму, которая была заработана, интересующий месяц. X, y вводятся с клавиатуры.

### **Вариант 27**

1. Строки и строковые функции: понятие строки, выбор кавычек, создание строки, тройные кавычки, специальные символы, действия над строками (обращение к элементу по индексу, срез строки, конкатенация строк, проверка на вхождение, повтор, функция len()), форматирование строки и метод format(), функции и методы для работы со строками, настройка локали, поиск и замена в строке, шифрование строк.

## **2. Разработка программы подбора санатория**

Компания содержит сведения о предлагаемых путевках: тип путевки (отдых, лечение), количество дней, питание (завтраки/все включено), стоимость.

Вывести все путевки требуемого типа (вводится с клавиатуры) стоимостью менее  $x$  в порядке убывания стоимости.

### **Вариант 28**

1. Структуры: понятие структуры? Отличие структуры от массива, объявление нового типа данных для хранения структур в Python, обращение к полю структуры, точечная запись, сортировка структуры.

### **2. Разработка программы учета продаж билетов в аквапарк**

Сведения о продажах содержат следующую информацию: дата, наименование аквапарка, уровень доступа, количество проданных билетов.

Вывести список самых популярных аквапарков в порядке убывания в указанный месяц. Вывести список самых популярных аквапарков в порядке убывания в указанный месяц. Месяц вводится с клавиатуры.

### **Вариант 29**

1. Файловый ввод/вывод: работа с файлами (открытие файла. методы для работы с файлами, функции для манипулирования файлами, работа с каталогами), работа с файлами в разных форматах (работа с CSV, чтение и запись JSON-данных, парсинг XML-файлов, преобразование словаря в XML, модификация и перезапись XML-кода, декодирование и кодирование шестнадцатеричных чисел, кодирование/декодирование base64).

### **2. Разработка программы учета продаж квартир**

Сведения о продажах содержат следующую информацию: год продажи, месяц продажи, количество проданных квартир, стоимость квартиры; характеристика объекта; номер объекта.

Определить общий доход от продажи за определенный месяц. Вывести список самых востребованных объектов в порядке убывания в указанный месяц. Месяц вводится с клавиатуры.

### **Вариант 30**

1. Циклы и условные операторы: условные операторы (логические значения, операторы сравнения, оператор if.else, блоки кода и отступы), циклы (цикл for, цикл while, операторы break и continue, функция range()), бесконечные циклы (бесконечный цикл по ошибке, намеренный бесконечный цикл), истинные и ложные значения.

### **2. Разработка программы автоматизации учета кадров на предприятии**

Сведения о сотрудниках компании содержат следующую информацию: Ф.И.О. сотрудника; год, когда был принят на работу; месяц, когда был принят на работу; когда заканчивается контракт; должность.

Вывести список сотрудников контракт которых заканчивается (с месяца  $x$  по месяц  $y$ ).

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

ФАКУЛЬТЕТ

КАФЕДРА (ВЫПУСКАЮЩАЯ)

**КУРСОВОЙ ПРОЕКТ (РАБОТА)**

по дисциплине « \_\_\_\_\_ »

на тему: « \_\_\_\_\_ »

**Выполнил (-а):**

\_\_\_\_\_ Ф.И.О.  
студент(ка) \_\_\_ курса, группы \_\_\_\_\_  
направления подготовки /специальность

\_\_\_\_\_

\_\_\_\_\_ формы обучения

\_\_\_\_\_

(подпись)

**Руководитель проекта (работы):**

\_\_\_\_\_

(Ф.И.О., должность, кафедра)

Проект (работа) допущен(а) к защите \_\_\_\_\_ (дата)  
\_\_\_\_\_ (подпись руководителя)

Проект (работа) выполнен(а)  
и защищен(а) с оценкой \_\_\_\_\_ (оценка) \_\_\_\_\_ (дата)

Члены комиссии:

\_\_\_\_\_ (должность) \_\_\_\_\_ (подпись) \_\_\_\_\_ (Ф.И.О.)

\_\_\_\_\_ (должность) \_\_\_\_\_ (подпись) \_\_\_\_\_ (Ф.И.О.)

\_\_\_\_\_ (должность) \_\_\_\_\_ (подпись) \_\_\_\_\_ (Ф.И.О.)

Воронеж 20\_\_

УТВЕРЖДАЮ  
Заведующий кафедрой

\_\_\_\_\_  
(наименование кафедры)

\_\_\_\_\_  
(Ф.И.О.)

Факультет \_\_\_\_\_

Кафедра \_\_\_\_\_

Направление подготовки (специальность) \_\_\_\_\_

направленность (профиль) /специализация \_\_\_\_\_

Задание  
на курсовую работу (проект)

Студента \_\_\_\_\_  
(Ф.И.О.)

по дисциплине \_\_\_\_\_

1. Тема работы (проекта) \_\_\_\_\_

\_\_\_\_\_

2. Цель \_\_\_\_\_

\_\_\_\_\_

3. Задачи \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

4. Перечень подлежащих разработке вопросов: \_\_\_\_\_

а) по теоретической части \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

б) по аналитической части \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

5. Исходные данные: \_\_\_\_\_

а) по литературным источникам \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

б) по вариантам, разработанным преподавателем \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

в) иное \_\_\_\_\_

\_\_\_\_\_

6. Список рекомендуемых источников \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

7. Контрольные сроки представления отдельных разделов курсовой работы (проекта):

25% – \_\_\_\_\_ «\_\_» \_\_\_\_\_ 20\_\_ г.

50% – \_\_\_\_\_ «\_\_» \_\_\_\_\_ 20\_\_ г.

75% – \_\_\_\_\_ «\_\_» \_\_\_\_\_ 20\_\_ г.

100% – \_\_\_\_\_ «\_\_» \_\_\_\_\_ 20\_\_ г.

8. Срок защиты студентом курсовой работы (проекта) «\_\_» \_\_\_\_\_ 20\_\_ г.

Дата выдачи задания «\_\_» \_\_\_\_\_ 20\_\_ г.

Руководитель курсовой работы (проекта)

\_\_\_\_\_ (ученая степень, звание) \_\_\_\_\_ (подпись) \_\_\_\_\_ (Ф.И.О.)

Задание принял(а) к исполнению студент(ка) \_\_\_\_\_

формы обучения \_\_\_\_\_ курса \_\_\_\_ группы \_\_\_\_\_

\_\_\_\_\_ (подпись) \_\_\_\_\_ (Ф.И.О.)