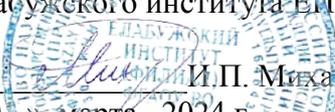


МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего
образования
"Казанский (Приволжский) федеральный университет"
Елабужский институт (филиал) КФУ



УТВЕРЖДАЮ

Заместитель директора по
образовательной деятельности
Елабужского института ЕИ КФУ

 И.П. Михайлова

« 01 » марта 2024 г.

МП

Программа дисциплины (модуля)

ОП.13 Программирование на языке высокого уровня

Направление подготовки/специальность: 09.02.07 «Информационные системы и
программирование»

Квалификация выпускника: Программист

Форма обучения: очная

Язык обучения: русский

Год начала обучения по образовательной программе: 2022

г.Елабуга, 2024

1. Цели освоения дисциплины

формирование знаний по:

- основным этапам разработки программного обеспечения;
- основным принципам технологии структурного программирования;
- основным принципам отладки и тестирования программных продуктов;
- языку программирования Python.

формирование умений по:

- осуществлению разработки кода программного модуля на языках низкого и высокого уровней;
- созданию программы на языке программирования Python по разработанному алгоритму;
- выполнению отладки и тестирования программы на языке программирования Python,

2. Место дисциплины в структуре ПССЗ

Учебная дисциплина ОП.13 «Программирование на языке высокого уровня» является общепрофессиональной дисциплиной профессионального цикла в соответствии с ФГОС по специальности СПО 09.02.07 «Информационные системы и программирование».

Осваивается четвертом курсе (8 семестр).

3. Компетенции обучающегося, формируемые в результате освоения дисциплины

В результате освоения дисциплины формируются компетенции:

| Индекс компетенции | Расшифровка приобретаемой компетенции |
|--------------------|--|
| ОК 09 | Использовать информационные технологии в профессиональной деятельности |
| ПК 1.1 | Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием |
| ПК 1.2 | Разрабатывать программные модули в соответствии с техническим заданием |

В результате освоения дисциплины обучающийся должен **знать:**

- основные этапы разработки программного обеспечения;
- основные принципы технологии структурного программирования;
- основным принципы отладки и тестирования программных продуктов;
- язык программирования Python.

В результате освоения дисциплины обучающийся должен **уметь:**

- осуществлять разработку кода программного модуля на языках низкого и высокого уровней;

- создавать программы на языке программирования Python по разработанному алгоритму;
- выполнять отладку и тестирование программ на языке программирования Python.

4. Структура и содержание дисциплины

4.1. Общая трудоемкость дисциплины в часах

Общая трудоемкость дисциплины составляет 108 часов.

Форма промежуточной аттестации по дисциплине: экзамен в 8 семестре.

| Разделы и темы дисциплины | | Семестр | Виды и часы аудиторной работы, их трудоемкость (в часах) | | | Самостоятельная работа | Текущие формы контроля |
|----------------------------|--|---------|--|----------------------|---------------------|------------------------|------------------------|
| | | | Лекции | Практические занятия | Лабораторные работы | | |
| Тема 1 | Разработка графических интерфейсов В программе на языке Python | 8 | 16 | 0 | 16 | 1 | Проверочная работа 1* |
| Тема 2 | Инструменты для создания графических Интерфейсов пользователя (GUI) | 8 | 16 | 0 | 16 | 1 | Проверочная работа 2* |
| Тема 3 | Элементы графического интерфейса | 8 | 16 | 0 | 16 | 2 | Проверочная работа 3* |
| Итого:100 | | | 48 | 0 | 48 | 4 | |
| Консультация | | | | | | 1 | |
| Экзамен | | | | | | 7 | |
| Всего по дисциплине | | | | | | 108 | |

* письменная Проверочная точка (выполняется за компьютером)

4.2. Содержание дисциплины

| Наименование разделов и тем | Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная работа обучающихся | Объем часов (лек/лаб.р./самост) | Уровень освоения* |
|---|---|---------------------------------|-------------------|
| 1 | 2 | 3 | 4 |
| Тема 1 Разработка графических интерфейсов | Содержание учебного материала | 33 (16/16/1) | |
| | 1. Событие. | 4 | 1 |
| | 2. Обработчик события. | 6 | 1 |
| | 3. Цикл обработки событий. | 6 | 1 |
| | Лабораторные работы: 1. Событие. 2. Обработчик события. 3. Цикл обработки событий. | 4 6 6 | 2 2 2 |
| | Самостоятельная работа обучающегося Подготовка к проверочной работе по темам: Событие. Обработчик события. Цикл обработки событий. Жуков, Р. А. Язык программирования Python. Практикум : учебное пособие / Р. А. Жуков. — Москва : ИНФРА-М, 2021. — 216 с. — (Среднее профессиональное образование). - ISBN 978-5-16-015638-5. - Текст : электронный. - URL: https://znanium.com/read?id=365208 (дата обращения: 04.08.2023). – Режим доступа: по подписке. стр. 180-182 | 2 | 3 |
| Тема 2. Инструменты для создания графических | Содержание учебного материала | 33 (16/16/1) | |
| | 1. Общие сведения о GUI Python. | 4 | 1 |
| | 2. Отслеживание событий. | 6 | 1 |
| | 3. Создание базового окна. | 6 | 1 |
| | Лабораторные работы 1. Общие сведения о GUI Python. 2. Отслеживание событий. 3. Создание базового окна. | 4 6 6 | 2 2 2 |
| | Самостоятельная работа обучающегося | 2 | 3 |

| | | | |
|--|---|-------------------------------|-------------|
| | <p>Подготовка к проверочной работе по темам: Общие сведения о GUI Python. Отслеживание событий. Создание базового окна.</p> <p>Жуков, Р. А. Язык программирования Python. Практикум: учебное пособие / Р. А. Жуков. — Москва : ИНФРА-М, 2021. — 216 с. — (Среднее профессиональное образование). - ISBN 978-5-16-015638-5. - Текст : электронный. - URL: https://znanium.com/read?id=365208 (дата обращения: 04.08.2023). – Режим доступа: по подписке. Стр. 182-185</p> | | |
| Тема 3. Элементы графического интерфейса | Содержание учебного материала | 34 (16/16/2) | |
| | 1. Создание и конфигурирование виджета. Менеджер размещения. | 4 | 1 |
| | 2. Использование элементов Button (Кнопка). Label (Надпись). Entry (Поле ввода). | 6 | 1 |
| | 3. Checkbutton (Флажок). Radiobutton (Переключатель). Другие виджеты. Метод bind. Canvas. | 6 | 1 |
| | Лабораторные работы | | |
| | 1. Создание и конфигурирование виджета. Менеджер размещения. 2. Использование элементов Button (Кнопка). Label (Надпись). Entry (Поле ввода). 3. Checkbutton (Флажок). Radiobutton (Переключатель). Другие виджеты. Метод bind. Canvas. | 4 6 6 | 2 2 2 |
| Самостоятельная работа обучающегося | 2 | 3 | |
| <p>Подготовка к проверочной работе по темам: Создание и конфигурирование виджета. Менеджер размещения. Использование элементов Button (Кнопка). Label (Надпись). Entry (Поле ввода). Checkbutton (Флажок). Radiobutton (Переключатель). Другие виджеты. Метод bind. Canvas.</p> <p>Жуков, Р. А. Язык программирования Python. Практикум : учебное пособие / Р. А. Жуков. — Москва : ИНФРА-М, 2021. — 216 с. — (Среднее профессиональное образование). - ISBN 978-5-16-015638-5. - Текст : электронный. - URL: https://znanium.com/read?id=365208 (дата обращения: 04.08.2023). – Режим доступа: по подписке. Стр. 187-205</p> | | | |
| Примерная тематика курсовой работы (проекта) | не предусмотрен | о | |
| Итого: | 100 (48/48/4) | | |

| | | |
|---------------------|------------|--|
| Консультация | 1 | |
| Экзамен | 7 | |
| Всего | 108 | |

*Для характеристики уровня освоения учебного материала используются следующие обозначения:

1. – ознакомительный (узнавание ранее изученных объектов, свойств);
2. – репродуктивный (выполнение деятельности по образцу, инструкции или под руководством)
3. – продуктивный (планирование и самостоятельное выполнение деятельности, решение проблемных задач)

4.3. Структура и содержание самостоятельной работы студентов

| Темы дисциплины | | Виды самостоятельной работы | Трудоемкость (в часах) | Формы контроля самостоятельной работы |
|-----------------|--------------------------------------|---|------------------------|---------------------------------------|
| 1 | 2 | 3 | 4 | 5 |
| Тема 1 | Разработка графических интерфейсов | <p>Самостоятельная работа обучающегося</p> <p>Подготовка к проверочной работе по темам: Событие. Обработчик события. Цикл обработки событий.</p> <p>Жуков, Р. А. Язык программирования Python. Практикум : учебное пособие / Р. А. Жуков. — Москва : ИНФРА-М, 2021. — 216 с. — (Среднее профессиональное образование). - ISBN 978-5-16-015638-5. - Текст : электронный. - URL: https://znanium.com/read?id=365208 (датаобращения: 04.08.2023). – Режим доступа: по подписке. стр. 180-182</p> | 1 | Проверочная работа 1 |
| Тема 2 | Инструменты для создания графических | <p>Самостоятельная работа обучающегося</p> <p>Подготовка к проверочной работе по темам: Общие сведения о GUI Python. Отслеживание событий. Создание базового окна.</p> <p>Жуков, Р. А. Язык программирования Python. Практикум : учебное пособие / Р. А. Жуков. — Москва : ИНФРА-М, 2021. — 216 с. — (Среднее профессиональное образование). - ISBN 978-5-16-015638-5. - Текст : электронный. - URL: https://znanium.com/read?id=365208 (датаобращения: 04.08.2023). – Режим доступа: по подписке. Стр. 182-185</p> | 1 | Проверочная работа 2* |
| Тема 3 | Элементы графического интерфейса | <p>Самостоятельная работа обучающегося</p> <p>Подготовка к проверочной работе по темам: Создание и конфигурирование виджета. Менеджер размещения. Использование элементов Button (Кнопка). Label (Надпись). Entry (Поле ввода). Checkbutton (Флажок). Radiobutton (Переключатель). Другие виджеты. Метод bind. Canvas.</p> <p>Жуков, Р. А. Язык программирования Python. Практикум : учебное пособие / Р. А. Жуков. — Москва : ИНФРА-М, 2021. — 216 с. — (Среднее профессиональное образование). - ISBN 978-5-16-015638-5. - Текст: электронный. - URL:</p> | 2 | Проверочная работа 3 |

| | | | | |
|----------------------------|--|--|----------|--|
| | | https://znanium.com/read?id=365208 (дата обращения: 04.08.2023). – Режим доступа: по подписке. Стр. 187-205 | | |
| <i>Всего по дисциплине</i> | | | 4 | |

5. Образовательные технологии

Освоение дисциплины «Программирование на языке высокого уровня» предполагает использование как традиционных (лекции, практические занятия с использованием методических материалов), так использованием в учебном процессе активных и интерактивных форм проведения занятий:

работа в малых группах, решение кейсов (анализ реальных проблемных ситуаций, имевших место в соответствующей области профессиональной деятельности, и поиск вариантов лучших решений), проблемное обучение (стимулирование студентов к самостоятельному приобретению знаний, необходимых для решения конкретной проблемы). Выполнение заданий требует использования не только учебников и пособий, но и информации, содержащейся в Интернете.

На лекциях и практических занятиях используются:

- информационная и презентационная лекция;
- тематические опросы;
- решение задач;
- коллективное выполнение заданий в подгруппах для обобщения тематического теоретического материала в схемах, таблицах;
- выполнение практических заданий с использованием среды программирования разработки PyCharm Community Edition.

Занятия, проводимые в активной и интерактивной формах

| Номер темы | Наименование темы | Форма проведения занятия | Объем в часах |
|---------------------|--------------------------------------|---------------------------------|---------------|
| Тема 1 | Разработка графических интерфейсов | Информационно-проблемная лекция | 16 |
| Тема 2 | Инструменты для создания графических | Информационно-проблемная лекция | 16 |
| Тема 3 | Элементы графического интерфейса | Информационно-проблемная лекция | 16 |
| Всего по дисциплине | | | 16 |

6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины

Фонд оценочных средств по дисциплине (модулю) включает оценочные материалы, направленные на проверку освоения компетенций, в том числе знаний, умений и навыков. Фонд оценочных средств включает оценочные средства текущего контроля и оценочные средства промежуточной аттестации.

Фонд оценочных средств по дисциплине находится в Приложении 1 к программе дисциплины (модулю).

7 Таблица соответствия компетенций, критериев оценки их освоения и оценочных средств

| Шифр компетенции | Планируемые результаты обучения | Оценочные средства | Критерии оценивания результатов обучения (баллы) | | | |
|------------------|---------------------------------|--------------------|--|---|---|---|
| | | | 2 | 3 | 4 | 5 |

| | | | | | | |
|--------|--|---|---|---|--|---|
| ОК 09 | <p>Знать основные этапы разработки программного обеспечения; основные принципы технологии структурного программирования; основным принципами отладки и тестирования программных продуктов; язык программирования Python</p> | <p>Проверочные работы 1-3 Вопросы</p> | <p>Не знает, допускает грубые ошибки</p> | <p>Демонстрирует частичные знания без грубых ошибок</p> | <p>Знает достаточно в базовом объеме</p> | <p>Демонстрирует высокий уровень знаний</p> |
| | <p>Уметь осуществлять разработку кода программного модуля на языках низкого и высокого уровней; создавать программы на языке программирования Python по разработанному алгоритму; выполнять отладку и тестирование программ на языке программирования Python</p> | <p>Проверочные работы 1-3 Вопросы экзамену 1-80 Практические варианты</p> | <p>Не умеет, демонстрирует частичные умения, допуская грубые ошибки</p> | <p>Демонстрирует частичные умения без грубых ошибок</p> | <p>Умеет применять знания на практике в базовом объеме</p> | <p>Демонстрирует высокий уровень умений</p> |
| ПК 1.1 | <p>Знать: основные этапы разработки программного обеспечения; основные принципы технологии структурного и объектно-ориентированного</p> | <p>Проверочные работы 1-3 Вопросы варианты</p> | <p>Не знает, допускает грубые ошибки</p> | <p>Демонстрирует частичные знания без грубых ошибок</p> | <p>Знает достаточно в базовом объеме</p> | <p>Демонстрирует высокий уровень знаний</p> |

| | | | | | | |
|--|--|--|---|---|--|---|
| | <p>программирования; способы оптимизации и приемы рефакторинга; основные принципы отладки и тестирования программных продуктов</p> | | | | | |
| | <p>Уметь: осуществлять разработку кода программного модуля на языках низкого и высокого уровней; создавать программу по разработанному алгоритму как отдельный модуль; выполнять отладку и тестирование программы на уровне модуля; осуществлять разработку кода программного модуля на современных языках программирования; уметь выполнять оптимизацию и рефакторинг программного кода; оформлять документацию на программные средства</p> | <p>Проверочные работы 1-3 Вопросы варианты</p> | <p>Не умеет, демонстрирует частичные умения, допуская грубые ошибки</p> | <p>Демонстрирует частичные умения без грубых ошибок</p> | <p>Умеет применять знания на практике в базовом объеме</p> | <p>Демонстрирует высокий уровень умений</p> |

| | | | | | | |
|--------|--|---|--|--|---|--------------------------------------|
| ПК 1.2 | Знать основные этапы разработки программного обеспечения; основные принципы технологии структурного программирования; основным принципами отладки и тестирования программных продуктов; язык программирования Python | Проверочные работы 1-3 Вопросы к экзамену 1-80 Практические варианты к экзамену 1-2 | Не знает, допускает грубые ошибки | Демонстрирует частичные знания без грубых ошибок | Знает достаточно в базовом объеме | Демонстрирует высокий уровень знаний |
| | Уметь осуществлять разработку кода программного модуля на языках низкого и высокого уровней; создавать программы на языке программирования Python по разработанно | Проверочные работы 1-3 Вопросы к экзамену 1-80 | Не умеет, демонстрирует частичные умения, допуская грубые ошибки | Демонстрирует частичные умения без грубых ошибок | Умеет применять знания на практике в базовом объеме | Демонстрирует высокий уровень умений |

8. Методические указания для обучающихся при освоении дисциплины

Работа на лабораторных занятиях предполагает активное участие в дискуссиях и решении задач. Для подготовки к занятиям рекомендуется выделять в материале проблемные вопросы, затрагиваемые преподавателем в лекции, и группировать информацию вокруг них.

При работе с терминами необходимо обращаться к словарям, в том числе доступным в Интернете, например на сайте <http://dic.academic.ru>.

При подготовке обучающихся по темам используются конспекты лекций и источники основной и дополнительной литературы. Подготовка докладов осуществляется с использованием нормативно-правовых документов и учебников.

Устный опрос по этой теме проводится в форме беседы. Работа на практических занятиях предполагает активное участие в дискуссиях и решении задач. Для подготовки к занятиям рекомендуется выделять в материале проблемные вопросы, затрагиваемые преподавателем в лекции, и группировать информацию вокруг них.

Решение задач проводится в группе с обсуждением хода решения, применяемых, схем, способов, проверкой результатов и проведением работы над ошибками.

Тестирование проводится после ознакомления с материалом темы. Обучающийся выполняет тестирование, рассчитанное по времени на 40-50 минут, на бумажном носителе. Тест включает в себя задания разного типа: на выбор одного или нескольких правильных ответов, на соответствие, краткий и числовой ответ. Для прохождения теста дается одна попытка. Далее сверяются и обсуждаются результаты с определением правильных ответов.

Промежуточная аттестация по этой дисциплине проводится в форме экзамена. При подготовке к экзамену необходимо опираться, прежде всего, на источники, которые разбирались на лекционных занятиях и на материалы практических занятий. В каждом билете экзамена содержится два вопроса.

9. Учебно-методическое и информационное обеспечение дисциплины

Основная литература:

Залогова, Л. А. Основы объектно-ориентированного программирования на базе языка C# / Л. А. Залогова. — 4-е изд., стер. — Санкт-Петербург : Лань, 2023. — 192 с. — ISBN 978-5-507-48276-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/345992> (дата обращения: 02.11.2023). — Режим доступа: для авториз. пользователей.

Жуков, Р. А. Язык программирования Python. Практикум : учебное пособие / Р.А. Жуков. — Москва : ИНФРА-М, 2023. — 216 с. + Доп. материалы [Электронный ресурс]. — (Среднее профессиональное образование). - ISBN 978-5-16-015638-5. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1916202> (дата обращения: 02.11.2023). – Режим доступа: по подписке.

Колдаев, В. Д. Основы алгоритмизации и программирования : учебное пособие / В.Д. Колдаев ; под ред. проф. Л.Г. Гагариной. — Москва : ФОРУМ : ИНФРА-М, 2022. — 414 с. — (Среднее профессиональное образование). - ISBN 978-5-8199-0733-7. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1735805> (дата обращения: 02.11.2023). – Режим доступа: по подписке.

| |
|---|
| <p>Гуриков, С. Р. Основы алгоритмизации и программирования на Python : учебное пособие / С.Р. Гуриков. — Москва : ИНФРА-М, 2023. — 343 с. — (Высшее образование: Бакалавриат). - ISBN 978-5-16-017142-5. - Текст : электронный. - URL: https://znanium.com/catalog/product/1913856 (дата обращения: 02.11.2023). – Режим доступа: по подписке.</p> |
| <p>Дополнительная литература:</p> |
| <p>Ульянова, Н. Д. Основные принципы алгоритмизации : учебно-методическое пособие / Н. Д. Ульянова. — Брянск : Брянский ГАУ, 2020. — 56 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/172114 (дата обращения: 02.11.2023). — Режим доступа: для авториз. пользователей.</p> |
| <p>Ночка, Е. И. Основы алгоритмизации и программирования. Ответы на контрольные вопросы.: Учебник / Ночка Е.И. - Москва :КУРС, НИЦ ИНФРА-М, 2017. - 59 с.: ISBN 978-5-906818-82-9. - Текст : электронный. - URL: https://znanium.com/catalog/product/772548 (дата обращения: 02.11.2023). – Режим доступа: по подписке.</p> |
| <p>Голицына, О. Л. Основы алгоритмизации и программирования : учебное пособие / О.Л. Голицына, И.И. Попов. — 4-е изд., испр. и доп. — Москва : ФОРУМ : ИНФРА-М, 2021. — 431 с. — (Среднее профессиональное образование). - ISBN 978-5-00091-570-7. - Текст : электронный. - URL: https://znanium.com/catalog/product/1150328 (дата обращения: 02.11.2023). – Режим доступа: по подписке.</p> |
| <p>Шакин, В. Н. Объектно-ориентированное программирование на Visual Basic в среде Visual Studio.NET : учебное пособие / В.Н. Шакин, А.В. Загвоздкина, Г.К. Сосновиков. — Москва : ФОРУМ : ИНФРА-М, 2024. — 398 с. — (Среднее профессиональное образование). - ISBN 978-5-00091-551-6. - Текст : электронный. - URL: https://znanium.com/catalog/product/2078382 (дата обращения: 02.11.2023). – Режим доступа: по подписке.</p> |
| <p>Программирование на языке высокого уровня : учебное пособие / составитель Е. И. Николаев. — Ставрополь : СКФУ, 2015. — 211 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/155245 (дата обращения: 01.11.2023). — Режим доступа: для авториз. пользователей.</p> |
| <p>Бедердинова, О. И. Основы алгоритмизации и структурного программирования : учебное пособие / О. И. Бедердинова. — Архангельск : САФУ, 2017. — 88 с. — ISBN 978-5-261-01227-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/161718 (дата обращения: 01.11.2023). — Режим доступа: для авториз. пользователей.</p> |

10. Материально-техническое и программное обеспечение дисциплины

Освоение дисциплины ОП.13 «Программирование на языке высокого уровня» предполагает использование следующего материально-технического обеспечения:

| Наименование дисциплины | Наименование кабинета, перечень оборудования |
|--|---|
| ОП.13 Программирование на языке высокого уровня | Автоматизированные рабочие места обучающихся: системный блок (процессор Intel Core i7, оперативная память 16 ГБ), автоматизированное рабочее место преподавателя: системный блок (процессор Intel Core i7, оперативная память 16 ГБ), комплект мебели (посадочных мест), комплект мебели (посадочных мест) для преподавателя, проектор и экран (интерактивная доска), трибуна, кондиционер, настенные полки, шкаф двухстворчатый с полками, веб-камера, маркерная доска, выход в Интернет, внутривузовская компьютерная сеть, доступ в электронную информационно-образовательную среду. Программное обеспечение общего и профессионального назначения: Office Professional Plus 2010, GIMP, Inkscape, Notepad ++, Python, Lazarus |

Учебно-методическая литература для данной дисциплины имеется в наличии в электронно-библиотечной системе "ZNANIUM.COM", доступ к которой предоставлен обучающимся. ЭБС "ZNANIUM.COM" содержит произведения крупнейших российских учёных, руководителей государственных органов, преподавателей ведущих вузов страны, высококвалифицированных специалистов в различных сферах бизнеса. Фонд

12

библиотеки сформирован с учетом всех изменений образовательных стандартов и включает учебники, учебные пособия, монографии, авторефераты, диссертации, энциклопедии, словари и справочники, законодательно-нормативные документы, специальные периодические издания и издания, выпускаемые издательствами вузов. В настоящее время ЭБС ZNANIUM.COM соответствует всем требованиям федеральных государственных образовательных стандартов среднего профессионального образования нового поколения.

Учебно-методическая литература для данной дисциплины имеется в наличии в электронно-библиотечной системе Издательства "Лань", доступ к которой предоставлен обучающимся. ЭБС Издательства "Лань" включает в себя электронные версии книг издательства "Лань" и других ведущих издательств учебной литературы, а также электронные версии периодических изданий по естественным, техническим и гуманитарным наукам. ЭБС Издательства "Лань" обеспечивает доступ к научной, учебной литературе и научным периодическим изданиям.

11. Методы обучения для обучающихся инвалидов и лиц с ограниченными возможностями здоровья

В образовательном процессе используются социально-активные и рефлексивные методы обучения, технологии социокультурной реабилитации с целью оказания помощи в установлении полноценных межличностных отношений с другими обучающимися, создании комфортного психологического климата в студенческой группе.

Условия обучения инвалидов и лиц с ограниченными возможностями здоровья:
- учебные аудитории, в которых проводятся занятия со студентами с нарушениями слуха, оборудованы мультимедийной системой (ПК и проектор), компьютерные тифлотехнологии базируются на комплексе аппаратных и программных средств,

обеспечивающих преобразование компьютерной информации доступные для слабовидящих формы (укрупненный текст);

- в образовательном процессе используются социально-активные и рефлексивные методы обучения: кейс-метод, метод проектов, исследовательский метод, дискуссии в форме круглого стола, конференции, метод мозгового штурма.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

Елабужский институт (филиал) федерального государственного автономного
образовательного учреждения высшего образования
«Казанский (Приволжский) федеральный университет»

УТВЕРЖДАЮ

Заместитель директора по
образовательной деятельности
Елабужского института ЕИ КФУ
И.П. Михайлова
« 01 » марта 2024 г.



ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

ОП.13 Программирование на языке высокого уровня
(наименование дисциплины)

09.02.07 Информационные системы и программирование
(код и наименование специальности)

Программист
(квалификация выпускника)

г.Елабуга, 2024

**Паспорт
фонда оценочных средств по дисциплине
ОП.13 Программирование на языке высокого уровня**

| Индекс компетенции | Расшифровка компетенции | Показатель формирования компетенции для данной дисциплины | Оценочные средства |
|---------------------------|---|--|---|
| 1 | 2 | 3 | 4 |
| ОК 09 | Использовать информационные технологии профессиональной деятельности в | Знать основные этапы разработки программного обеспечения; основные принципы технологии структурного программирования; основным принципы отладки и тестирования программных продуктов; язык программирования Python. Уметь осуществлять разработку кода программного модуля на языках низкого и высокого уровней; создавать программы на языке программирования Python по разработанному алгоритму; выполнять отладку и тестирование программ на языке программирования Python | Проверочные работы 1-3 Вопросы к экзамену 1-30 Практические варианты к экзамену 1-2 |
| ПК 1.1 | Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием | Знать: основные этапы разработки программного обеспечения; основные принципы технологии структурного и объектно-ориентированного программирования; способы оптимизации и приемы рефакторинга; основные принципы отладки и тестирования программных продуктов. | Проверочные работы 1-3 Вопросы к экзамену 1-30 Практические варианты к экзамену 1-2 |

| | | | |
|--------|--|---|--|
| | | <p>Уметь: осуществлять разработку кода программного модуля на языках низкого и высокого уровней; создавать программу по разработанному алгоритму как отдельный модуль; выполнять отладку и тестирование программы на уровне модуля; осуществлять разработку кода программного модуля на современных языках программирования; уметь выполнять оптимизацию и рефакторинг программного кода; оформлять документацию на программные средства</p> | |
| ПК 1.2 | Разрабатывать программные модули в соответствии с техническим заданием | <p>Знать основные этапы разработки программного обеспечения; основные принципы технологии структурного программирования; основным принципы отладки и тестирования программных продуктов; язык программирования Python.</p> <p>Уметь осуществлять разработку кода программного модуля на языках низкого и высокого уровней; создавать программы на языке программирования Python по разработанному алгоритму; выполнять отладку и тестирование программ на языке программирования Python</p> | <p>Проверочные работы 1-3 Вопросы к экзамену 1-30 Практические варианты к экзамену 1-2</p> |

Тема 1 Разработка графических интерфейсов

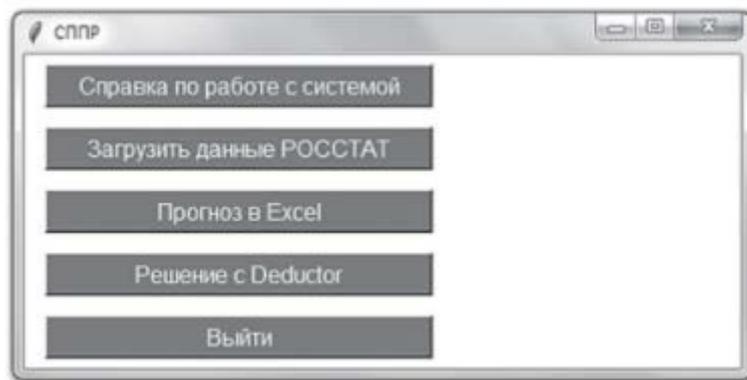
Комплект заданий для проверочной работы 1 ОП.13 «Программирование на языке высокого уровня» (ОК 09, ПК 1.1, ПК 1.2)

Вариант 1

- 1 Укажите возможные события (не менее семи), которые могут возникнуть при диалоге пользователя с программой. (ОК 09, ПК 1.1, ПК 1.2)
2. Основываясь на GUI, представленном на рисунке, предложите, какие события должна отслеживать программа. (ОК 09, ПК 1.1, ПК 1.2)
3. Составьте алгоритм и представьте его в виде блок-схемы обработчика запроса на загрузку интернет-страницы в браузер, предполагая, что модель взаимодействия основана на клиент-серверной архитектуре. (ОК 09, ПК 1.1, ПК 1.2)

Вариант 2

- 1 Основываясь на GUI, представленном на рисунке, предложите свой вариант интерфейса СППР (система поддержки принятия решений). Как можно задействовать радиокнопки? (ОК 09, ПК 1.1, ПК 1.2)
2. Составьте концептуальную модель (описательная модель на естественном языке) обработчика нажатия ЛКМ на кнопку панели инструментов «Сохранить» в MS Office. (ОК 09, ПК 1.1, ПК 1.2)
3. Основываясь на GUI, представленном на рисунке, предложите, какие события должна отслеживать программа. (ОК 09, ПК 1.1, ПК 1.2)



Тема 2. Инструменты для создания графических интерфейсов

Комплект заданий для проверочной работы 2 ОП.13 «Программирование на языке высокого уровня» (ОК 09, ПК 1.1, ПК 1.2)

Вариант 1

Создайте окно с названием «Студенты», где разместите кнопки «Вывести список», «Ввести данные», «Создать запрос», «Выйти» тремя рассмотренными способами. Сравните результат и прокомментируйте преимущества и недостатки использования данных методов. (ОК 09, ПК 1.1, ПК 1.2)

Вариант 2

Создайте окно с названием «Покупатели», где разместите кнопки «Вывести список», «Ввести данные», «Создать запрос», «Выйти» тремя рассмотренными способами. Сравните результат и прокомментируйте преимущества и недостатки использования данных методов. (ОК 09, ПК 1.1, ПК 1.2)

Тема 3. Элементы графического интерфейса

Комплект заданий для проверочной работы 3

ОП.13 «Программирование на языке высокого уровня»

Защита программного приложения (ОК 09, ПК 1.1, ПК 1.2)

1. Разработайте диалоговую программу на тему: Российские кинофильмы. (ОК 09, ПК 1.1, ПК 1.2)
2. Разработайте диалоговую программу на тему: Спорт. (ОК 09, ПК 1.1, ПК 1.2)
3. Разработайте диалоговую программу на тему: Автомобили. (ОК 09, ПК 1.1, ПК 1.2)
4. Разработайте диалоговую программу на тему: Российская эстрада. (ОК 09, ПК 1.1, ПК 1.2)
5. Разработайте диалоговую программу на тему: Программирование. (ОК 09, ПК 1.1, ПК 1.2)
6. Разработайте диалоговую программу на тему: Животные. (ОК 09, ПК 1.1, ПК 1.2)
7. Разработайте диалоговую программу на тему: Города России. (ОК 09, ПК 1.1, ПК 1.2)
8. Разработайте диалоговую программу на тему: Достопримечательности моего города. (ОК 09, ПК 1.1, ПК 1.2)
9. Разработайте диалоговую программу на тему: Кулинария. (ОК 09, ПК 1.1, ПК 1.2)
10. Разработайте диалоговую программу на тему: Мотоциклы. (ОК 09, ПК 1.1, ПК 1.2)
11. Разработайте диалоговую программу на тему: Живопись. (ОК 09, ПК 1.1, ПК 1.2)
12. Разработайте диалоговую программу на тему: Книги. (ОК 09, ПК 1.1, ПК 1.2)
13. Разработайте диалоговую программу на тему: Мультфильмы. (ОК 09, ПК 1.1, ПК 1.2)
14. Разработайте диалоговую программу на тему: Зарубежное кино. (ОК 09, ПК 1.1, ПК 1.2)
15. Разработайте диалоговую программу на тему: Зарубежная эстрада. (ОК 09, ПК 1.1, ПК 1.2)
16. Разработайте диалоговую программу на тему: Военная техника. (ОК 09, ПК 1.1, ПК 1.2)
17. Разработайте диалоговую программу на тему: Классическая музыка. (ОК 09, ПК 1.1, ПК 1.2)
18. Разработайте диалоговую программу на тему: Домашние животные. (ОК 09, ПК 1.1, ПК 1.2)
19. Разработайте диалоговую программу на тему: Информатика. (ОК 09, ПК 1.1, ПК 1.2)
20. Разработайте диалоговую программу на тему: Моря и океаны. (ОК 09, ПК 1.1, ПК 1.2)
21. Разработайте диалоговую программу на тему: Великие пустыни мира. (ОК 09, ПК 1.1, ПК 1.2)
22. Разработайте диалоговую программу на тему: Герои олимпиад. (ОК 09, ПК 1.1, ПК 1.2)
23. Разработайте диалоговую программу на тему: История России. (ОК 09, ПК 1.1, ПК 1.2)
24. Разработайте диалоговую программу на тему: Русские поэты. (ОК 09, ПК 1.1, ПК 1.2)
25. Разработайте диалоговую программу на тему: Дикие животные. (ОК 09, ПК 1.1, ПК 1.2)
26. Разработайте диалоговую программу на тему: Российские киноактеры. (ОК 09, ПК 1.1, ПК 1.2)
27. Разработайте диалоговую программу на тему: Интернет. (ОК 01, ОК 02, ОК 04, ОК 05, ОК 08, ОК 09, ПК 1.2, ПК 2.4)
28. Разработайте диалоговую программу на тему: Русские народные сказки. (ОК 09, ПК 1.1, ПК 1.2)
29. Разработайте диалоговую программу на тему: Легенды российского балета. (ОК 09, ПК 1.1, ПК 1.2)
30. Разработайте диалоговую программу на тему: Оргтехника. (ОК 09, ПК 1.1, ПК 1.2)

Критерии оценки:

| Компетенции | Планируемые результаты обучения | Критерии оценивания результатов обучения (баллы) | | | |
|-------------|--|--|---|---|---|
| | | «неудовлетворительно» | «удовлетворительно» | «хорошо» | «отлично» |
| | | [критерии выставления оценки «неудовлетворительно»] | [критерии выставления оценки «удовлетворительно»] | [критерии выставления оценки «хорошо»] | [критерии выставления оценки «отлично»] |
| ОК 09 | Знать основные этапы разработки программного обеспечения; основные принципы технологии структурного программирования; основным принципы отладки и тестирования программных продуктов; язык программирования Python | Не знает, допускает грубые ошибки | Демонстрирует частичные знания без грубых ошибок | Знает достаточное базовое объёме | Демонстрирует высокий уровень знаний |
| | Уметь осуществлять разработку кода программного модуля на языках низкого и высокого уровней; создавать программы на языке программирования Python по разработанному алгоритму; выполнять отладку и тестирование программ на языке программирования Python | Не умеет, демонстрирует частичные умения, допуская грубые ошибки | Демонстрирует частичные умения без грубых ошибок | Умеет применять знания на практике в базовом объёме | Демонстрирует высокий уровень умений |
| ПК 1.1 | Знать: основные этапы разработки программного обеспечения; основные принципы технологии структурного и объектно-ориентированного программирования; способы оптимизации и приемы рефакторинга; основные принципы отладки и тестирования программных продуктов | Не знает, допускает грубые ошибки | Демонстрирует частичные знания без грубых ошибок | Знает достаточное базовое объёме | Демонстрирует высокий уровень знаний |

| | | | | | |
|--------|---|---|---|--|---|
| | <p>Уметь: осуществлять разработку кода программного модуля на языках низ-кого и высокого уровней; создавать программу по разработанному алгоритму как отдельный модуль; выполнять отладку и тестирование программы на уровне модуля; осуществлять разработку кода программного модуля на современных языках программирования; уметь выполнять оптимизацию и рефакторинг программного кода; оформлять документацию на программные средства</p> | <p>Не умеет, демонстрирует частичные умения, допуская грубые ошибки</p> | <p>Демонстрирует частичные умения без грубых ошибок</p> | <p>Умеет применять знания на практике в базовом объеме</p> | <p>Демонстрирует высокий уровень умений</p> |
| ПК 1.2 | <p>Знать основные этапы разработки программного обеспечения; основные принципы технологии структурного программирования ; основным принципом отладки и тестирования программных продуктов; язык программирования Python</p> | <p>Не знает, допускает грубые ошибки</p> | <p>Демонстрирует частичные знания без грубых ошибок</p> | <p>Знает достаточное количество базового объема</p> | <p>Демонстрирует высокий уровень знаний</p> |
| | <p>Уметь осуществлять разработку кода программного модуля на языках низкого и высокого уровней; создавать программы на языке программирования Python по разработанному алгоритму; выполнять отладку</p> | <p>Не умеет, демонстрирует частичные умения, допуская грубые ошибки</p> | <p>Демонстрирует частичные умения без грубых ошибок</p> | <p>Умеет применять знания на практике в базовом объеме</p> | <p>Демонстрирует высокий уровень умений</p> |

| | | | | | |
|--|---|--|--|--|--|
| | итестирование программ на языке программирования Python | | | | |
|--|---|--|--|--|--|

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Елабужский институт (филиал) федерального государственного автономного
образовательного учреждения высшего образования
«Казанский (Приволжский) федеральный университет»

Вопросы к экзамену (ОК 09, ПК 1.1, ПК 1.2)

1. Что такое Python? (ОК 09, ПК 1.1, ПК 1.2)
Ответ: это язык программирования общего назначения, который широко применяется в различных областях: от создания банальных веб-страниц до систем управления роутерами на других планетах.
2. Каковы преимущества использования Python? (ОК 09, ПК 1.1, ПК 1.2)
Ответ: Он интерпретируемый, а по тому и работа с ним будет более интерактивной. У Python, в отличие от C++, есть пакетный менеджер Python синтаксический гораздо проще, чем C++
В python встроен сборщик мусора, так что при изучении не придётся отвлекаться на ручное управление памятью.
У Python динамическая типизация, что также идёт на пользу интерактивности.
3. Как Python интерпретируется как язык? (ОК 09, ПК 1.1, ПК 1.2)
Ответ: Сначала Python проверяет синтаксис программы
Компилирует и преобразует его в байт-код, а непосредственно байт-код загружается в системную память.
Затем скомпилированный байт-код интерпретируется из памяти для его выполнения.
4. Как в Python управляется память? (ОК 09, ПК 1.1, ПК 1.2)
Ответ: В Python памятью управляет менеджер Python, который определяет, куда поместить данные приложения в память.
5. Как писать комментарии на Python? (ОК 09, ПК 1.1, ПК 1.2)
Ответ: В Python комментарии создаются с помощью знака #, за которым следует какой-либо описательный текст.
6. Что такое строки документации в Python? (ОК 09, ПК 1.1, ПК 1.2)
Ответ: это строковые литералы, которые пишутся сразу после определения функции, метода, класса или модуля.
7. Что такое функция в Python? (ОК 09, ПК 1.1, ПК 1.2)
Ответ: Функции в языке программирования Python технически представляют собой блоки повторяемого кода, доступ к которым осуществляется по имени функции с передачей аргументов.
8. Что такое индексы? (ОК 09, ПК 1.1, ПК 1.2)
Ответ: функция `index()` – элемент, который вы хотите найти, и последовательность, в которой он находится, – и возвращает значение его индекса.
9. Что такое отрицательные индексы и почему они используются? (ОК 09, ПК 1.1, ПК 1.2)
В Python отрицательные индексы позволяют обращаться к элементам последовательности (например, строке или списку) с конца, начиная с -1. Таким образом, последний элемент последовательности имеет индекс -1, предпоследний -2 и так далее.
10. Что такое словарь в Python? (ОК 09, ПК 1.1, ПК 1.2)
Ответ: Словари в Python – это изменяемые отображения ссылок на объекты, доступные по ключу.
11. Как работает `break`? (ОК 09, ПК 1.1, ПК 1.2)
Ответ: `Break` - это ключевое слово в python, которое используется для вывода управления программой из цикла.
12. Что такое функция `map` в Python? (ОК 09, ПК 1.1, ПК 1.2)

Ответ: Python map() — это встроенная функция, которая позволяет обрабатывать и преобразовывать все элементы в итерируемом объекте без использования явного цикла for, методом, широко известным как сопоставление (mapping).

13. Как добавить новое значение в объект списка? (ОК 09, ПК 1.1, ПК 1.2)

Ответ: Для добавления нового значения в объект списка в Python можно воспользоваться методом append().

14. Что такое поверхностная копия? Что такое глубокая копия? (ОК 09, ПК 1.1, ПК 1.2)

Ответ: Поверхностное копирование создает отдельный новый объект или список, но вместо копирования дочерних элементов в новый объект, оно просто копирует ссылки на их адреса памяти. Следовательно, если вы сделаете изменение в исходном объекте, оно будет отражено в скопированном объекте, и наоборот. Короче говоря, обе копии зависят друг от друга.

Глубокая копия создает новую и отдельную копию всего объекта или списка со своим уникальным адресом памяти.

15. Как создать пустой класс в Python? (ОК 09, ПК 1.1, ПК 1.2)

Ответ: В Python для записи пустого класса используется оператор pass. pass - это специальный оператор в Python, который ничего не делает. Он работает только как фиктивный оператор. Однако объекты пустого класса также могут быть созданы.

16. Что такое операторы тождественности в Python? (ОК 09, ПК 1.1, ПК 1.2)

Ответ: В Python существует два оператора тождественности: is и is not. Эти операторы сравнивают расположение двух объектов в памяти. Операторы тождественности используются не часто, но очень интересны с точки зрения того, как Python размещает объекты в памяти. Оператор "is". Возвращает True, если оба операнда указывают на один и тот же объект, иначе возвращает False.

17. Как обрабатывать входные данные в Python? (ОК 09, ПК 1.1, ПК 1.2)

Ответ: В Python входные данные можно обрабатывать с использованием различных методов и функций, в зависимости от типа данных и требуемой обработки.

Чтение входных данных

Обработка входных данных

18. Что такое PYTHONPATH? (ОК 09, ПК 1.1, ПК 1.2)

Ответ: Переменная среды PYTHONPATH используется в Python для указания списка каталогов, из которых можно импортировать модули.

19. Назовите пример режимов обработки файлов с помощью Python? (ОК 09, ПК 1.1, ПК 1.2)

Ответ: r - открыть файл для чтения. (по умолчанию)

w - открыть файл для записи. Если файл уже существует, его данные будут удалены перед открытием. В противном случае будет создан новый файл

x - открыт для исключительного создания, если файл уже существует

a - открыт для записи, добавляя к концу файла, если он существует.

b - двоичный режим

t - текстовый режим (по умолчанию)

+ r - открыть файл для обновления (чтение и запись)

20. Что такое pickling и unpickling? (ОК 09, ПК 1.1, ПК 1.2)

Ответ: Модуль pickle принимает любой объект Python, преобразует его в строковое представление и сохраняет в файл с помощью функции dump, такой процесс называется pickling. Процесс извлечения исходных объектов Python из сохраненного строкового представления называется unpickling.

21. Как получить текущий рабочий каталог с помощью Python? (ОК 09, ПК 1.1, ПК 1.2)

Ответ:

```
import os
```

```
dir_path = os.path.dirname(os.path.realpath(__file__))
```

22. Что такое приведение в Python? (ОК 09, ПК 1.1, ПК 1.2)

Преобразование типов - это преобразование объекта из одного типа данных в другой тип данных.

Неявное преобразование типов автоматически выполняется интерпретатором Python. Python позволяет избежать потери данных в неявном преобразовании типов.

Явное преобразование типов также называется приведением типов, типы данных объекта преобразуются с использованием predefined функции.

При приведении типов может произойти потеря данных, поскольку мы приводим объект к определенному типу данных.

23. Как можно удалить переменные в Python? (ОК 09, ПК 1.1, ПК 1.2)

Ответ: В Python удаление переменной осуществляется с помощью оператора del.

24. Что такое pandas в Python? (ОК 09, ПК 1.1, ПК 1.2)

Ответ: Pandas (Python Data Analysis Library) - это библиотека для языка программирования Python, которая предоставляет удобные инструменты для обработки и анализа данных.

25. Что означает одинарное и двойное подчеркивание перед именем объекта? (ОК 09, ПК 1.1, ПК 1.2)

Ответ: Имена в классе с начальным подчеркиванием (_) предназначены для указания другим пользователям, что атрибут или функция предназначены для использования в программе как частные.

Пользователям рекомендуется использовать одинарное подчеркивание (_) для локальных переменных и двойное подчеркивание (__) для полностью приватных переменных.

26. Что такое генератор? (ОК 09, ПК 1.1, ПК 1.2)

Ответ: генераторы в Python – это функции, которые возвращают итератор.

27. Каковы типы наследования в Python? (ОК 09, ПК 1.1, ПК 1.2)

Ответ: В Python существует два основных типа наследования: одиночное наследование и множественное наследование.

Одиночное наследование (single inheritance) - класс наследуется только от одного родительского класса.

Множественное наследование (multiple inheritance) - класс наследуется от нескольких родительских классов.

28. Что такое распаковка кортежа? (ОК 09, ПК 1.1, ПК 1.2)

Ответ: Распаковка (unpacking, также называемая Деструктуризация) представляет разложение коллекции (кортежа, списка и т.д.) на отдельные значения.

29. Освобождается ли вся память при выходе из Python? (ОК 09, ПК 1.1, ПК 1.2)

Ответ: Python обычно освобождает большинство объектов, как только их число ссылок достигает нуля. В случае циклических ссылок это никогда не происходит, поэтому сборщик мусора периодически ходит по памяти и освобождает объекты с циркулярной ссылкой.

30. Является ли функция допустимой, если она не имеет оператора return? (ОК 09, ПК 1.1, ПК 1.2)

Ответ: В Python функция может быть допустимой даже без оператора return, если она не должна возвращать какое-либо значение. В таком случае она может выполнять только определенные действия или изменять состояние объектов, но не возвращать результат.

Тестовые задания к экзамену (ОК 09, ПК 1.1, ПК 1.2)

1. Что представляет собой Python-список? (ОК 09, ПК 1.1, ПК 1.2)

- 1) упорядоченный массив
- 2) многомерный массив
- 3) разновидность FIFO-очереди

- 4) позиционно упорядоченную коллекцию с произвольными типами элементов

Правильный ответ: 4

2. Какой метод отвечает за добавление элемента в конец списка? (ОК 09, ПК 1.1, ПК 1.2)

- 1) include()
- 2) add()
- 3) extend()
- 4) append()

Правильный ответ: 4

3. Какой максимальный размер у списка в Python? (ОК 09, ПК 1.1, ПК 1.2)

- 1) 1 млн. элементов
- 2) 10 тыс. элементов
- 3) 100 тыс. элементов
- 4) фиксированного размера нет

Правильный ответ: 4

4. За что отвечает следующий фрагмент кода: (ОК 09, ПК 1.1, ПК 1.2)

```
[123, 'hello', 1.23]  
insert(1, 'inserted')
```

- 1) удаление указанного элемента
- 2) проверка наличия указанного элемента с указанной позицией
- 3) замена элемента с индексом 1 на указанный элемент
- 4) добавление элемента с индексом 1

Правильный ответ: 4

5. Для каких типов элементов списка метод sort() работает без ошибок? (ОК 09, ПК 1.1, ПК 1.2)

- 1) только для целочисленных
- 2) только для строковых
- 3) для однотипных элементов одного списка
- 4) для любых типов элементов

Правильный ответ: 3

6. Выберите одно верное утверждение про списки: (ОК 09, ПК 1.1, ПК 1.2)

- 1) списки имеют фиксированный размер
- 2) списки представляют собой массивы
- 3) списки в Python являются изменяемыми
- 4) списки невозможно сортировать

Правильный ответ: 3

7. Какая функция отвечает за вывод длины списка? (ОК 09, ПК 1.1, ПК 1.2)

- 1) getLen()
- 2) size()
- 3) length()
- 4) len()

Правильный ответ: 4

8. Какой метод отвечает за удаление элемента с указанной позицией? (ОК 09, ПК 1.1, ПК 1.2)

- 1) delete()
- 2) pop()
- 3) remove()
- 4) clear()

Правильный ответ: 2

9. Какой метод отвечает за разворачивание списка? (ОК 09, ПК 1.1, ПК 1.2)

- 1) rebuild()
- 2) reverse()
- 3) rotate()
- 4) reorder()

Правильный ответ: 2

10. Какой метод отвечает за удаление указанного значения элемента? (ОК 09, ПК 1.1, ПК 1.2)

- 1) clear()

- 2) remove()
- 3) delete()
- 4) pop()

Правильный ответ: 2

11. Выберите верные утверждения о языке программирования Python. (ОК 09, ПК 1.1, ПК 1.2))

- 1) высокоуровневый
- 2) низкоуровневый
- 3) интерпретируемый
- 4) компилируемый.

Правильный ответ: 1,3

12. Обязательно ли в Пайтоне объявлять тип переменной? (ОК 09, ПК 1.1, ПК 1.2)

- 1) 1.по желанию программиста
- 2) обязательно
- 3) зависит от ситуации
- 4) объявлять тип не нужно

Правильный ответ: 4

13. Какие из представленных фрагментов кода являются литералами? (ОК 09, ПК 1.1, ПК 1.2)

- 1) '#'
- 2) 0.77
- 3) 5**2
- 4) b = [1, 2].

Правильный ответ: 1,2

14. Какие из предлагаемых имен допустимы в Python? (ОК 09, ПК 1.1, ПК 1.2)

- 1) s_5
- 2) _5s
- 3) 5_s
- 4) Ц5

Правильный ответ: 1,2,4

15. Как в Пайтоне создать пустой список? (ОК 09, ПК 1.1, ПК 1.2)

- 1) li = [],
- 2) li = None,
- 3) li = 0
- 4) li = list().

Правильный ответ: 1,4

16. Какие из предложенных вариантов объектов относятся к итерируемым? (ОК 09, ПК 1.1, ПК 1.2)

- 1.2)
- 1) 'abc',
- 2) (25,),
- 3) 321
- 4) {'a': 1}.

Правильный ответ: 1,2,4

17. Какая инструкция используется в Python для определения функций? (ОК 09, ПК 1.1, ПК 1.2)

- 1) df,
- 2) def
- 3) dfn
- 4) define

Правильный ответ: 2

18. Что возвращает функция, в которой отсутствует инструкция return? (ОК 09, ПК 1.1, ПК 1.2)

- 1) объект функции
- 2) None
- 3) False
- 4) ничего

Правильный ответ: 2

19. Выберите верные утверждения о списках. (ОК 09, ПК 1.1, ПК 1.2)

- 1) неизменяемые
- 2) упорядоченные
- 3) последовательности
- 4) изменяемые

Правильный ответ: 2,3,4

20. Какие из представленных литералов относятся к типу list? (ОК 09, ПК 1.1, ПК 1.2)

- 1) ['1', '2', '3'],
- 2) [1, 2, 3]
- 3) [(1, 2, 3)],
- 4) ([1], [2], [3])

Правильный ответ: 1,3

Ключи к тестовым заданиям:

| | | | | | | | | | |
|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 4 | 4 | 4 | 4 | 3 | 3 | 4 | 2 | 2 | 2 |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| 13 | 4 | 12 | 124 | 14 | 124 | 2 | 2 | 234 | 13 |

Критерии оценки на экзамене

| «отлично» | «хорошо» | «удовлетворительно» | «неудовлетворительно» |
|--|--|--|---|
| студент раскрывает теоретический вопрос билета, практическое задание выполняет без ошибок, уверенно отвечает на дополнительные вопросы | студент раскрывает теоретический вопрос, практическое задание выполняет без ошибок, на дополнительные вопросы отвечает неуверенно, допускает не точности в определениях. | студент раскрывает теоретический вопрос не в полной мере, допускает неточности в формулировках (1-2 ошибки), практическое задание выполнено частично, с допущением ошибок в расчётах | Теоретический вопрос не раскрыт, практическое задание не выполнено. |