

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Воронежский государственный технический университет»

УТВЕРЖДАЮ
Декан факультета информационных
технологий и компьютерной безопасности
_____/П.Ю. Гусев/
«31» августа 2021 г.

РАБОЧАЯ ПРОГРАММА
дисциплины

«Основы программирования на Python»

Направление подготовки 09.03.01 Информатика и вычислительная техника

Профиль Программное обеспечение автоматизированных систем

Квалификация выпускника бакалавр

Нормативный период обучения 4 года / 4 года и 11 м.

Форма обучения очная / заочная

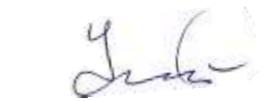
Год начала подготовки 2021

Автор программы



/Юров А.Н./

Заведующий кафедрой
Компьютерных
интеллектуальных
технологий проектирования



Чижов М.И.

Руководитель ОПОП



/В.В. Ветохин/

Воронеж 2022

1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

1.1. Цели дисциплины

Целью изучения дисциплины является изучение и применение языка программирования Python для реализации алгоритмов, обеспечение фундаментальной математической подготовки, адаптированной к решению задач на ЭВМ, позволяющей будущим специалистам ориентироваться в научно-технической информации, использовать численные методы и представление информации для создания программных решений в тех областях и сферах деятельности, в которых они будут трудиться.

Изучение дисциплины должно способствовать формированию у студентов основ научного мышления, в том числе: владение основными методами, способами и средствами управления информацией, создание алгоритмов и освоение методик при разработке программ для решения практических задач.

1.2. Задачи освоения дисциплины

- знакомство с направлениями развития лингвистических средств языков программирования для решения задач с применением вычислительной техники;
- знакомство с методами структурного и объектно-ориентированного программирования как наиболее распространенными и эффективными методами разработки программных продуктов;
- разработка алгоритмов на основе структурного и объектно-ориентированного подхода;
- ознакомление студентов с тенденцией развития программного обеспечения и указание перспективных направлений при решении практических задач;
- освоение языка Python, а также стандартов PEP8, спецификаций и последующих решений на их основе;

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина «Основы программирования на Python» относится к дисциплинам части, формируемой участниками образовательных отношений блока Б1.

Дисциплины и практики, для которых освоение данной дисциплины (модуля) необходимо как последующее:

- Объектно-ориентированное программирование;
- Среды визуального программирования;
- Компьютерная графика

3. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Процесс изучения дисциплины «Основы программирования на Python» направлен на формирование следующих компетенций:

ПК-4 - Способен разрабатывать и использовать техническую документацию в соответствии со спецификой образовательной программы

ПК-1 - Способен применять методы моделирования в профессиональной деятельности

Компетенция	Результаты обучения, характеризующие сформированность компетенции
ПК-4	знать -нормативные акты и материалы, позволяющие разрабатывать и использовать техническую документацию в соответствии со спецификой образовательной программы и готовить техническое задание в процессе разработке специализированного программного обеспечения
	уметь -оформлять техническое задание в соответствии со стандартами; -разрабатывать техническое задание для разработки специализированного программного обеспечения -обеспечивать производственный процесс машиностроительного предприятия
	владеть -программными решениями для ведения и оформления технической документации и технических заданий для решений, учитывающих специфику образовательной программы
ПК-1	знать - основы по методам моделирования, описание форм геометрических объектов, стандарты кодирования
	уметь -применять полученные знания при разработке программного обеспечения для моделирования процессов; -проводить формализацию задач в области САПР на основе программных интерфейсов производственных систем
	владеть - инструментальными средствами для разработки программного обеспечения, создавать самостоятельные приложения на основе готовых

	модулей, разрабатывать встраиваемые программные решения в известные системы автоматизированного проектирования
--	--

4. ОБЪЕМ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины «Основы программирования на Python» составляет 5 з.е.

Распределение трудоемкости дисциплины по видам занятий
очная форма обучения

Виды учебной работы	Всего часов	Семестры
		5
Аудиторные занятия (всего)	90	90
В том числе:		
Лекции	36	36
Лабораторные работы (ЛР)	54	54
Самостоятельная работа	54	54
Курсовой проект	+	+
Часы на контроль	36	36
Виды промежуточной аттестации - экзамен	+	+
Общая трудоемкость: академические часы	180	180
зач.ед.	5	5

заочная форма обучения

Виды учебной работы	Всего часов	Семестры
		7
Аудиторные занятия (всего)	24	24
В том числе:		
Лекции	8	8
Лабораторные работы (ЛР)	16	16
Самостоятельная работа	147	147
Курсовой проект	+	+
Часы на контроль	9	9
Виды промежуточной аттестации - экзамен	+	+
Общая трудоемкость: академические часы	180	180
зач.ед.	5	5

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

5.1 Содержание разделов дисциплины и распределение трудоемкости по видам занятий

очная форма обучения

№ п/п	Наименование темы	Содержание раздела	Лекц	Лаб. зан.	СРС	Всего, час
1	Введение	Общие сведения. Простейшие программы. Переменные и арифметические выражения	6	10	8	24

2	Основы структурного программирования	Условные операторы. Циклические конструкции Работа со строками. Массивы и конструкции пакетного хранения данных. Функции	6	10	8	24
3	Контейнерные типы данных	Списки, кортежи, словари, множества. Исключения	6	10	8	24
4	Основы ООП	ООП. Классы	6	8	10	24
5	Высокоуровневые операции и модули	Дата и время. Работа с файлами. Модули. Импорт данных	6	8	10	24
6	Обменные форматы данных и графический интерфейс приложений	Графические возможности. Введение в графический интерфейс. Обмен данными	6	8	10	24
Итого			36	54	54	144

заочная форма обучения

№ п/п	Наименование темы	Общие сведения. Простейшие программы. Переменные и арифметические выражения	Лекц	Лаб. зан.	СРС	Всего, час
1	Введение	Условные операторы. Циклические конструкции Работа со строками. Массивы и конструкции пакетного хранения данных. Функции	2	4	24	30
2	Основы структурного программирования	Списки, кортежи, словари, множества. Исключения	2	4	24	30
3	Контейнерные типы данных	ООП. Классы	2	2	24	28
4	Основы ООП	Дата и время. Работа с файлами. Модули. Импорт данных	2	2	24	28
5	Высокоуровневые операции и модули	Графические возможности. Введение в графический интерфейс. Обмен данными	-	2	26	28
6	Обменные форматы данных и графический интерфейс приложений	Общие сведения. Простейшие программы. Переменные и арифметические выражения	-	2	25	27
Итого			8	16	147	171

5.2 Перечень лабораторных работ

Очная форма обучения

Неделя семестра	Наименование лабораторной работы	Объем часов	Виды контроля
1/1	Среды и средства разработки на языке Python	4	Отчет и защита
1/2-1/3	Базовые конструкции и ключевые слова	8	Отчет и защита
1/4-1/6	Структурный подход к созданию приложений	8	Отчет и защита
1/7-1/8	Дополнительные возможности Python	8	Отчет и защита
1/9-1/10	Работа с модулями	12	Отчет и защита
1/11-1/17	Работа с контейнерными типами данных	8	Отчет и защита
1/18	Введение в графическую разработку приложений	6	Отчет и защита
Итого часов:		54	

Заочная форма обучения

Неделя семестра	Наименование лабораторной работы	Объем часов	Виды контроля
	Среды и средства разработки на языке Python	2	Отчет и защита
	Базовые конструкции и ключевые слова	2	Отчет и защита
	Структурный подход к созданию приложений	2	Отчет и защита
	Дополнительные возможности Python	2	Отчет и защита
	Работа с модулями	4	Отчет и защита
	Работа с контейнерными типами данных	2	Отчет и защита
	Введение в графическую разработку приложений	2	Отчет и защита
Итого часов:		16	

6. ПРИМЕРНАЯ ТЕМАТИКА КУРСОВЫХ ПРОЕКТОВ (РАБОТ) И КОНТРОЛЬНЫХ РАБОТ

В соответствии с учебным планом освоение дисциплины предусматривает выполнение курсового проекта в 5 семестре для очной формы обучения, в 7 семестре для заочной формы обучения.

Примерная тематика курсового проекта: «Расчетные задачи с применением графического интерфейса»

Задачи, решаемые при выполнении курсового проекта:

- расчетные задачи;
- простейшие геометрические преобразования;
- построение деревьев, введение в графы;
- сортировка данных;
- программирование математических выражений и структур;
- численные методы;
- вывод графических объектов на экран (графики функций, простейшие геометрические объекты);
- файловые операции;
- работа со строковыми и табличными данными;
- создание многооконных кроссплатформенных приложений;
- интеграция со сторонними приложениями.

Курсовой проект включает в себя графическую часть и расчетно-пояснительную записку.

Темы курсовых проектов согласно заданному варианту:

1. Решение алгоритмических задач с условиями.
2. Решение алгоритмических задач с конструкциями повторов.

3. Решение алгоритмических задач с функциями.
4. Решение алгоритмических задач с файлами.
5. Разработка текстового редактора.
6. Разработка табличного редактора.
7. Разработка архиватора.
8. Разработка файлового менеджера.
9. Разработка векторного редактора.
10. Разработка графического редактора.
11. Разработка приложения по просмотру графических форматов.
12. Разработка приложения по просмотру 3D моделей.
13. Разработка приложения по анализу аппаратной части.
14. Разработка приложения по тестированию оборудования ЭВМ.
15. Разработка инженерного калькулятора

7. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

7.1. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

7.1.1 Этап текущего контроля

Результаты текущего контроля знаний и межсессионной аттестации оцениваются по следующей системе:

«аттестован»;

«не аттестован».

Компетенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Аттестован	Не аттестован
ПК-4	<p>знать</p> <ul style="list-style-type: none"> -нормативные акты и материалы, позволяющие разрабатывать и использовать техническую документацию в соответствии со спецификой образовательной программы и готовить техническое задание в процессе разработке специализированного программного обеспечения 	тест	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	<p>уметь</p> <ul style="list-style-type: none"> -оформлять техническое задание в соответствии со стандартами; -разрабатывать техническое задание для разработки специализированного программного 	тест	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах

	обеспечения -обеспечивать производственный процесс машиностроительного предприятия			
	владеть -программными решениями для ведения и оформления технической документации и технических заданий для решений, учитывающих специфику образовательной программы	тест	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
ПК-1	знать - основы по методам моделирования, описание форм геометрических объектов, стандарты кодирования	текст	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	уметь -применять полученные знания при разработке программного обеспечения для моделирования процессов; -проводить формализацию задач в области САПР на основе программных интерфейсов производственных систем	тест	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах
	владеть - инструментальными средствами для разработки программного обеспечения, создавать самостоятельные приложения на основе готовых модулей, разрабатывать встраиваемые программные решения в известные системы автоматизированного проектирования	тест	Выполнение работ в срок, предусмотренный в рабочих программах	Невыполнение работ в срок, предусмотренный в рабочих программах

7.1.2 Этап промежуточного контроля знаний

Результаты промежуточного контроля знаний оцениваются в 5 семестре для очной формы обучения, 7 семестре для заочной формы обучения по четырехбалльной системе:

«отлично»;

«хорошо»;

«удовлетворительно»;

«неудовлетворительно».

Компетенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Отлично	Хорошо	Удовл.	Неудовл.
ПК-4	знать -нормативные акты и материалы, позволяющие разрабатывать и использовать техническую документацию в соответствии со спецификой образовательной программы и готовить техническое задание в процессе разработке специализированного программного обеспечения	Тест	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов
	уметь -оформлять техническое задание в соответствии со стандартами; -разрабатывать техническое задание для разработки специализированного программного обеспечения -обеспечивать производственный процесс машиностроительного предприятия	Решение стандартных практических задач	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
	владеть -программными решениями для ведения и оформления технической документации и технических заданий для решений, учитывающих специфику образовательной программы	Решение прикладных задач в конкретной предметной области	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены
ПК-1	знать - основы по методам моделирования, описание форм геометрических объектов, стандарты кодирования	Тест	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов
	уметь -применять полученные знания при разработке программного обеспечения для	Решение стандартных практических задач	Задачи решены в полном объеме и получены верные	Продемонстрирован верный ход решения всех, но не получен	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены

моделирования процессов; -проводить формализацию задач в области САПР на основе программных интерфейсов производственных систем		ответы	верный ответ во всех задачах		
владеть - инструментальными средствами для разработки программного обеспечения, создавать самостоятельные приложения на основе готовых модулей, разрабатывать встраиваемые программные решения в известные системы автоматизированного проектирования	Решение прикладных задач в конкретной предметной области	Задачи решены в полном объеме и получены верные ответы	Продемонстрирован верный ход решения всех, но не получен верный ответ во всех задачах	Продемонстрирован верный ход решения в большинстве задач	Задачи не решены

7.2 Примерный перечень оценочных средств (типичные контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности)

7.2.1 Примерный перечень заданий для подготовки к тестированию

1.Класс - это:

- любой тип данных, определяемый пользователем;
- описание объекта, который определяется пользователем и включает данные, а также методы по работе с ними;
- структура, для которой в программе имеются методы работы с нею

Верный ответ: 2

2.Уровни доступа атрибутов и методов класса в Python

- все переменные, так и методы, могут быть только открытыми (public);
- только переменные могут быть открытыми (public);
- Python не существует квалификаторов доступа к полям класса

Верный ответ: 3

3.Что называется конструктором?

- метод, имя которого совпадает с именем класса и который вызывается автоматически при объявлении класса (до создания объекта класса);
- метод, имя которого `__init__()` и который вызывается при создании

объекта класса;

3. метод, имя которого совпадает с именем класса и который необходимо явно вызывать из головной программы при объявлении объекта класса;

4. метод, имя которого совпадает с именем класса и который вызывается автоматически при создании объекта класса;

Верный ответ: 2

4. Объект — это?

1. переменная, содержащая указатель на класс;

2. экземпляр класса;

3. класс, который содержит в себе данные и методы их обработки;

Верный ответ: 2

5. Декоратор класса в Python

1. метод, который уничтожает объект;

2. метод, который удаляет объект;

3. метод, который освобождает память, занимаемую объектом;

4. метод-обёртка, в которую можно завернуть другой метод, и, тем самым, изменить его функциональность, не меняя код;

Верный ответ: 4

6. Выберите правильное утверждение?

1. деструктор — это метод класса, применяемый для удаления объекта;

2. деструктор — это метод класса, применяемый для освобождения памяти, занимаемой объектом;

3. деструктор — это отдельная функция головной программы, применяемая для освобождения памяти, занимаемой объектом;

4. деструктор наследуется, но должен быть перегружен;

Верный ответ: 2

7. Что называется наследованием?

1. это механизм, посредством которого производный класс получает элементы родительского и может дополнять либо изменять их свойства и методы;

2. это механизм переопределения методов базового класса;

3. это механизм, посредством которого производный класс получает все поля базового класса;

4. это механизм, посредством которого производный класс получает элементы родительского, может их дополнить, но не может

переопределить;

Верный ответ: 1

8. Для доступа к элементам объекта используются:

1. при обращении через имя объекта – точка;
2. при обращении через имя объекта – два двоеточия;
3. при обращении через имя объекта – два двоеточия и скобки

Верный ответ: 1

9. Функции и методы класса начинаются с сокращенного обозначения:

1. fun;
2. def;
3. res

Верный ответ: 2

10. Кортежи в Python – это

1. структура данных без каких-либо выраженных атрибутивных свойств;
2. структура данных используется для хранения последовательности упорядоченных и изменяемых элементов;
3. структура данных используется для хранения последовательности упорядоченных и неизменяемых элементов.

Верный ответ: 3

7.2.2 Примерный перечень заданий для решения стандартных задач

Вопрос 1

Как получить данные от пользователя?

Варианты ответов

1. Использовать метод read()
2. Использовать метод get()
3. Использовать метод cin()
4. Использовать метод readLine()
5. Использовать метод input()

Верный ответ: 5

Вопрос 2

Какая функция выводит что-либо в консоль (на экран монитора)?

Варианты ответов

- 1.out();
- 2.log();
- 3.print();
- 4.write();

Верный ответ: 4

Вопрос 3

Переменная int:

Варианты ответов

- 1.вещественная переменная
- 2.символьная строка
- 3.логическая переменная
- 4.целая переменная

Верный ответ: 4

Вопрос 4

Переменная float:

Варианты ответов

- 1.целая переменная
- 2.вещественная переменная
- 3.логическая переменная

Верный ответ: 2

Вопрос 5

Какие имена являются правильными в Python:

Варианты ответов

- 1.sum
- 2.41And
- 3.A+B

Верный ответ: 1

Вопрос 6

Какой ответ будет при выполнении следующего скрипта на Python:

```
a=5
```

```
b=a+5
```

```
a=b*100
```

```
print(a)
```

Варианты ответов

1.25

2.250

3.1000

4.0

Верный ответ: 3

Вопрос 7

Какой ответ будет при выполнении следующего вычисления `print(23%2)`:

Варианты ответов

1.0

2.1

3.23

4.-1

Верный ответ: 2

Вопрос 8

Какой ответ будет при выполнении следующего вычисления `print(9**2)`:

Варианты ответов

1.18

2.92

3.81

4.0

Верный ответ: 3

Вопрос 9

Тело цикла - это...

Варианты ответов

1. группа команд, не входящих в циклическую структуру
2. произвольный текст
3. произвольная группа команд
4. группа команд, повторяющихся некоторое число раз

Верный ответ: 4

Вопрос 10

Если говорят, что программа "заиклилась", то это значит...

Варианты ответов

1. тело цикла не выполняется
2. тело цикла выполняется бесконечно долго
3. тело цикла выполняется только один раз

Верный ответ: 2

7.2.3 Примерный перечень заданий для решения прикладных задач

1. Функции. Область видимости в функциях.
2. Передача переменных и массивов в функцию и возврат значений.
3. Использование функций с параметрами по умолчанию.
4. Перегруженные функции
5. Рекурсивные функции.
6. Понятие структур. Описание структур.
7. Контейнерные типы данных: стеки и очереди
8. Контейнерные типы данных: списки
9. Контейнерные типы данных: кортежи
10. Применение исключений в разработке программ

7.2.4 Примерный перечень вопросов для подготовки к зачету

Не предусмотрено учебным планом

7.2.5 Примерный перечень заданий для решения прикладных задач

1. Структура и синтаксис простейшего консольного приложения.
2. Консольный ввод-вывод, использование функций из математического модуля.
3. Отображение типов данных. Пример консольного приложения
4. Беззнаковые целочисленные типы данных, представление в памяти.
5. Вещественные типы данных, их представление в памяти.
6. Символьный тип данных, представление символов в памяти.

7. Преобразование типов данных.
8. Арифметические и логические операции. Приоритеты операций.
9. Битовые операции.
10. Переменные-константы.
11. Условный оператор if ... else. Примеры
12. Оператор цикла for.
13. Строковые переменные.
14. Операция присваивания и ее разновидности.
15. Понятие массивов. Одномерные массивы.

7.2.6. Методика выставления оценки при проведении промежуточной аттестации

Экзамен проводится по тест-билетам, каждый из которых содержит 10 вопросов и задачу. Каждый правильный ответ на вопрос в тесте оценивается 1 баллом, задача оценивается в 10 баллов (5 баллов верное решение и 5 баллов за верный ответ). Максимальное количество набранных баллов – 20.

1. Оценка «Неудовлетворительно» ставится в случае, если студент набрал менее 6 баллов.

2. Оценка «Удовлетворительно» ставится в случае, если студент набрал от 6 до 10 баллов

3. Оценка «Хорошо» ставится в случае, если студент набрал от 11 до 15 баллов.

4. Оценка «Отлично» ставится, если студент набрал от 16 до 20 баллов.)

7.2.7 Паспорт оценочных материалов

№ п/п	Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции	Наименование оценочного средства
1	Введение	ПК-4, ПК-1	Тест, контрольная работа, защита лабораторных работ, защита реферата, требования к курсовому проекту.
2	Основы структурного программирования	ПК-4, ПК-1	Тест, контрольная работа, защита лабораторных работ, защита реферата, требования к курсовому проекту.
3	Контейнерные типы данных	ПК-4, ПК-1	Тест, контрольная работа, защита лабораторных работ, защита реферата, требования к курсовому проекту.
4	Основы ООП	ПК-4, ПК-1	Тест, контрольная работа, защита лабораторных работ, защита реферата, требования к курсовому проекту.
5	Высокоуровневые операции и	ПК-4, ПК-1	Тест, контрольная работа,

	модули		защита лабораторных работ, защита реферата, требования к курсовому проекту.
6	Обменные форматы данных и графический интерфейс приложений	ПК-4, ПК-1	Тест, контрольная работа, защита лабораторных работ, защита реферата, требования к курсовому проекту.

7.3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Тестирование осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных тест-заданий на бумажном носителе. Время тестирования 30 мин. Затем осуществляется проверка теста экзаменатором и выставляется оценка согласно методики выставления оценки при проведении промежуточной аттестации.

Решение стандартных задач осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных задач на бумажном носителе. Время решения задач 30 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методики выставления оценки при проведении промежуточной аттестации.

Решение прикладных задач осуществляется, либо при помощи компьютерной системы тестирования, либо с использованием выданных задач на бумажном носителе. Время решения задач 30 мин. Затем осуществляется проверка решения задач экзаменатором и выставляется оценка, согласно методики выставления оценки при проведении промежуточной аттестации.

Защита курсовой работы, курсового проекта или отчета по всем видам практик осуществляется согласно требованиям, предъявляемым к работе, описанным в методических материалах. Примерное время защиты на одного студента составляет 20 мин.

8 УЧЕБНО МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ)

8.1 Перечень учебной литературы, необходимой для освоения дисциплины

1. Ковалевская, Е. В. Методы программирования : учебное пособие / Е. В. Ковалевская, Н. В. Комлева. — Москва : Евразийский открытый институт, 2011. — 320 с. — ISBN 978-5-374-00356-7. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/10784.html>

2. Фарафонов, А. С. Программирование на языке высокого уровня : методические указания к проведению лабораторных работ по курсу «Программирование» / А. С. Фарафонов. — Липецк : Липецкий государственный технический университет, ЭБС АСВ, 2013. — 32 с. — ISBN 2227-8397. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/22912.html>

3. Юров А.Н. Методические указания к лабораторным работам № 1-3 по дисциплине “Программирование” 14-2013 2013, магн.

4. Юров А.Н. Методические указания к лабораторным работам № 4-5 по дисциплине “Программирование” 15-2013 2013, магн.

8.2 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного программного обеспечения, ресурсов информационно-телекоммуникационной сети «Интернет», современных профессиональных баз данных и информационных справочных систем:

Лицензионное ПО:

- Среда разработки приложений Visual Studio
- PyScripter (open source)

Ресурсы информационно-телекоммуникационной сети «Интернет»:

- <http://www.edu.ru/>

Современные профессиональные базы данных:

- eLIBRARY.RU
- База ГОСТ docplan.ru
- Образовательный портал ВГТУ

Информационные справочные системы:

- <http://window.edu.ru>
- <https://wiki.cchgeu.ru/>

9 МАТЕРИАЛЬНО-ТЕХНИЧЕСКАЯ БАЗА, НЕОБХОДИМАЯ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА

Специализированная лекционная аудитория, оснащенная оборудованием для лекционных демонстраций и проекционной аппаратурой

Учебные лаборатории:

- 1 “Лаборатория интеллектуальных систем проектирования”
- 2 “Лаборатория компьютерного моделирования и дизайна”
- 3 “ Интернет-лаборатория ”

10. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

По дисциплине «Основы программирования на Python» читаются лекции, проводятся лабораторные работы, выполняется курсовой проект.

Основой изучения дисциплины являются лекции, на которых излагаются наиболее существенные и трудные вопросы, а также вопросы, не нашедшие отражения в учебной литературе.

Лабораторные работы выполняются на лабораторном оборудовании в соответствии с методиками, приведенными в указаниях к выполнению работ.

Методика выполнения курсового проекта изложена в учебно-методическом пособии. Выполнять этапы курсового проекта должны своевременно и в установленные сроки.

Контроль усвоения материала дисциплины производится проверкой курсового проекта, защитой курсового проекта.

Вид учебных занятий	Деятельность студента
Лекция	Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; пометать важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначение вопросов, терминов, материала, которые вызывают трудности, поиск ответов в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на лекции или на практическом занятии.
Лабораторная работа	Лабораторные работы позволяют научиться применять теоретические знания, полученные на лекции при решении конкретных задач. Чтобы наиболее рационально и полно использовать все возможности лабораторных для подготовки к ним необходимо: следует разобрать лекцию по соответствующей теме, ознакомиться с соответствующим разделом учебника, проработать дополнительную литературу и источники, решить задачи и выполнить другие письменные задания.
Самостоятельная работа	Самостоятельная работа студентов способствует глубокому усвоению учебного материала и развитию навыков самообразования. Самостоятельная работа предполагает следующие составляющие: <ul style="list-style-type: none"> - работа с текстами: учебниками, справочниками, дополнительной литературой, а также проработка конспектов лекций; - выполнение домашних заданий и расчетов; - работа над темами для самостоятельного изучения; - участие в работе студенческих научных конференций, олимпиад; - подготовка к промежуточной аттестации.
Подготовка к промежуточной аттестации	Готовиться к промежуточной аттестации следует систематически, в течение всего семестра. Интенсивная подготовка должна начаться не позднее, чем за месяц-полтора до промежуточной аттестации. Данные перед экзаменом, экзаменом три дня эффективнее всего использовать для повторения и систематизации материала.