

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**Федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Нижегородский государственный университет им.
Н.И. Лобачевского»**

Институт информационных технологий, математики и механики

УТВЕРЖДЕНО
решением Ученого совета ННГУ
от 30.11.2022 г. протокол № 13

Рабочая программа дисциплины

Программирование на скриптовых языках

Уровень высшего образования
магистратура

Направление подготовки
010402 Прикладная математика и информатика

Направленность образовательной программы
Вычислительные методы и суперкомпьютерные технологии

Форма обучения
очная

Нижегород
2022

1. Место и цели дисциплины в структуре ООП

Место дисциплины в структуре ООП

Данная дисциплина (Б1.В.ДВ.04.01) читается во втором семестре магистратуры, относится к части, формируемой участниками образовательных отношений. Дисциплина опирается на базовые знания студентов по языкам и методам программирования, алгоритмам и структурам данных.

Цель освоения дисциплины

Целями освоения дисциплины «Программирование на скриптовых языках» являются овладение скриптовым языком Python, а также рассмотрение круга прикладных задач, решаемых с помощью скриптовых языков общего назначения.

2. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции (код, содержание индикатора)	Результаты обучения по дисциплине	
<i>ПК-5 Способен разрабатывать и применять математические методы, системное и прикладное программное обеспечение для решения задач научной деятельности</i>	<i>ПК-5.1. Знает типовые математические методы и методологии разработки системного и прикладного программного обеспечения для решения задач научной деятельности</i>	<i>ЗНАТЬ Методику разработки программ на скриптовых языках на примере языка Python для решения задач научной деятельности.</i>	Лабораторная работа (зачет)
	<i>ПК-5.2. Умеет применять типовые математические методы и методологии разработки системного и прикладного программного обеспечения для решения задач научной деятельности</i>	<i>УМЕТЬ Применять на практике язык Python для решения задач научной деятельности.</i>	Лабораторная работа (текущий контроль)

ПК-12 Способен разрабатывать и применять математические методы, системное и прикладное программное обеспечение для решения задач производственно-технологической деятельности	ПК-12.1. Знает типовые математические методы и методологии разработки системного и прикладного программного обеспечения для решения производственно-технологических задач	ЗНАТЬ Методику разработки программ на скриптовых языках на примере языка Python для решения задач производственно-технологической деятельности.	Лабораторная работа (зачет)
	ПК-12.2. Умеет применять типовые математические методы и методологии разработки системного и прикладного программного обеспечения для решения производственно-технологических задач	УМЕТЬ Применять на практике язык Python для решения задач производственно-технологической деятельности.	Лабораторная работа (текущий контроль)

3. Структура и содержание дисциплины

3.1. Трудоемкость дисциплины

	Очная форма обучения
Общая трудоемкость	3 ЗЕТ
Часов по учебному плану	108
в том числе	
аудиторные занятия (контактная работа):	33
– занятия лекционного типа	16
– занятия семинарского типа	0
– занятия лабораторного типа	16
– текущий контроль (КСР)	1
самостоятельная работа	75
Промежуточная аттестация: зачет	0

3.2. Содержание дисциплины

Наименование и краткое содержание разделов и тем дисциплины	Всего (часы)	в том числе				Самостоятельная работа обучающегося, часы
		Контактная работа (работа во взаимодействии с преподавателем), часы				
		Занятия лекционного типа	Занятия семинарского типа	Занятия лабораторного типа	Всего контактных часов	
Введение в язык программирования Python	14	2		2	4	10
Объектно-ориентированные средства в языке Python	21	4		4	8	13
Парсинг систем разметки на языке Python	17	2		2	4	13
Использование СОМ объектов в скриптовых языках	17	2		2	4	13
Виды тестирования, оценка качества тестирования, средства статического анализа	21	4		4	8	13
Автоматизация тестирования графического пользовательского интерфейса	17	2		2	4	13
Текущий контроль (КСР)	1				1	
Промежуточная аттестация – зачет	0					
Итого	108	16		16	33	75

Практические занятия (семинарские занятия /лабораторные работы) организуются, в том числе в форме практической подготовки, которая предусматривает участие обучающихся в выполнении отдельных элементов работ, связанных с будущей профессиональной деятельностью.

Текущий контроль успеваемости реализуется в форме проверки работ на занятиях лабораторного типа.

Промежуточная аттестация проходит в форме зачета.

4. Учебно-методическое обеспечение самостоятельной работы обучающихся

Самостоятельная работа студентов подразделяется на следующие категории:

- Изучение учебной литературы (см. перечень образовательных материалов).
- Выполнение лабораторных работ на следующие темы:
 - Лабораторная работа «Разбор C++ проекта с помощью скрипта на языке Python и применение технологии Single Compilation Unit».
 - Лаб. работа «Автоматическое создание графиков и отчётов в MS Excel на языке Python».
 - Лабораторная работа «Разбор текстовых файлов с применением регулярных выражений».
 - Лабораторная работа «Интеграция C/C++ библиотек в скрипты на языке Python».

- Лабораторная работа «Автоматизация сбора данных с веб-страниц».
- Лабораторная работа «Автоматизация тестирования графического интерфейса пользователя на языке Python».

Задания для проведения текущего контроля и промежуточной аттестации по итогам освоения дисциплины приведены в п. 5.2.

5. Фонд оценочных средств для промежуточной аттестации по дисциплине, включающий:

5.1. Описание шкал оценивания результатов обучения по дисциплине

Уровень сформированности компетенций (индикатора достижения компетенций)	Шкала оценивания сформированности компетенций						
	Плохо	Неудовлетворительно	Удовлетворительно	Хорошо	Очень хорошо	Отлично	Превосходно
	Не зачтено		Зачтено				
<u>Знания</u>	Отсутствие знаний теоретического материала. Невозможность оценить полноту знаний вследствие отказа обучающегося от ответа	Уровень знаний ниже минимальных требований. Имели место грубые ошибки.	Минимально допустимый уровень знаний. Допущено много негрубых ошибок.	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько незначительных ошибок	Уровень знаний в объеме, соответствующем программе подготовки, без ошибок.	Уровень знаний в объеме, превышающем программу подготовки.
<u>Умения</u>	Отсутствие минимальных умений. Невозможность оценить наличие умений вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы основные умения. Имели место грубые ошибки.	Продemonстрированы основные умения. Решены типовые задачи с негрубыми ошибками. Выполнены все задания, но не в полном объеме.	Продemonстрированы все основные умения. Решены все основные задачи с негрубыми ошибками. Выполнены все задания, в полном объеме, но некоторые с недочетами.	Продemonстрированы все основные умения. Решены все основные задачи. Выполнены все задания, в полном объеме, но некоторые с недочетами.	Продemonстрированы все основные умения, решены все основные задачи с отдельными незначительными недочетами, выполнены все задания в полном объеме.	Продemonстрированы все основные умения, решены все основные задачи. Выполнены все задания, в полном объеме без недочетов
<u>Навыки</u>	Отсутствие владения материалом. Невозможность оценить наличие навыков вследствие отказа	При решении стандартных задач не продемонстрированы базовые навыки. Имели	Имеется минимальный набор навыков для решения стандартных задач с некоторыми недочетами.	Продemonстрированы базовые навыки при решении стандартных задач с некоторыми недочетами	Продemonстрированы базовые навыки при решении стандартных задач без ошибок и недочетов.	Продemonстрированы навыки при решении нестандартных задач без ошибок и недочетов.	Продemonстрирован творческий подход к решению нестандартных задач.

	обучающего ся от ответа	место грубые ошибки.					
--	----------------------------	----------------------------	--	--	--	--	--

Шкала оценки при промежуточной аттестации

Оценка		Уровень подготовки
Зачтено	Превосходно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «превосходно»
	Отлично	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «отлично», при этом хотя бы одна компетенция сформирована на уровне «отлично»
	Очень хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «очень хорошо», при этом хотя бы одна компетенция сформирована на уровне «очень хорошо»
	Хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «хорошо», при этом хотя бы одна компетенция сформирована на уровне «хорошо»
	Удовлетворительно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «удовлетворительно», при этом хотя бы одна компетенция сформирована на уровне «удовлетворительно»
Не зачтено	Неудовлетворительно	Хотя бы одна компетенция сформирована на уровне «неудовлетворительно», ни одна из компетенций не сформирована на уровне «плохо»
	Плохо	Хотя бы одна компетенция сформирована на уровне «плохо»

Критерий оценивания лабораторной работы

Результаты работы	Оценка
Работа выполнена в полном объеме и в срок, результаты работы алгоритма корректные на тестовых примерах, результаты работы представлены преподавателю.	Зачтено
Работа не выполнена или выполнена не в полном объеме (программа работает некорректно на тестовых примерах, результаты работы не представлены преподавателю).	Не зачтено

5.2. Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения

5.2.1. Типовые темы лабораторных работ для оценки сформированности компетенции ПК-5

Пример простой лабораторной работы

Реализовать скрипт на языке Python для прямого сравнения двух файлов независимо от порядка строк. Имена файлов задаются позиционными аргументами командной строки. Например, следующие файлы одинаковы с точностью до порядка строк:

файл1.txt:

это тестовый файл

файл тестовый сие есть
это ещё одна строка

файл2.txt:
файл тестовый сие есть
это ещё одна строка
это тестовый файл

Таким образом команда **python compare.py файл1.txt файл2.txt** должна выдать сообщение о том, что файлы одинаковы в приведённом смысле. Если они не одинаковы, выводить для каждого файла список строк, уникальных для этого файла. Например:

файл1.txt (unique lines):
этой строки нет во втором файле

Пример сложной лабораторной работы

Реализовать скрипт на языке Python, который считывает XML-структуру проектного файла MS Visual Studio 2008 (.vcproj) и формирует новый проектный файл со следующими свойствами:

- все .cpp файлы исключены из сборки для всех конфигураций (но остаются в проекте).
- все .cpp файлы включаются (#include) в один _merged_.cpp, который добавляется к проекту и включается в сборку.

5.2.2. Типовые темы лабораторных работ для оценки сформированности компетенции ПК-12

Пример простой лабораторной работы

Написать следующий скрипт на языке Python. Ключом командной строки “--bad-words” задан список “плохих” слов, разделённых точкой с запятой (;). Ключом -f|--file задан проверяемый файл. Написать скрипт, который выводит все строки файла, содержащие хотя бы одно плохое слово. Например, так:

```
> python check_bad_words.py --bad-words error;warning;assert --file test.txt
```

```
Bad word “error” was found at line 3: Unknown error occurred
```

```
Bad word “warning” was found at line 7: Warning: function printf is deprecated, use printf_s instead
```

В случае, если ключ --file не задан, делать эту проверку для всех файлов *.txt в текущей папке (без углубления в подпапки).

Пример сложной лабораторной работы

Реализовать скрипт на языке Python, который считывает файл с матрицей целых чисел (разделённых табуляцией или пробелами) и отсортированный по возрастанию первый столбец использует для создания в MS Excel графика следующего вида: по оси абсцисс – значения из отсортированного столбца, по оси ординат – количество ненулевых элементов, пройденных от начала столбца.

6. Учебно-методическое и информационное обеспечение дисциплины

а) основная литература:

- Северенс Ч. Курс "Введение в программирование на Python". URL: <http://www.intuit.ru/studies/courses/12179/1172/info>.

б) дополнительная литература:

- Хахаев И. Курс "Практикум по алгоритмизации и программированию на Python". URL: <http://www.intuit.ru/studies/courses/3489/731/info>.
- Сузи Р. Курс "Язык программирования Python". URL: <http://www.intuit.ru/studies/courses/49/49/info>.

7. Материально-техническое обеспечение дисциплины

Помещения представляют собой учебные аудитории для проведения учебных занятий, предусмотренных программой (лекционного и лабораторного типа), оснащенные оборудованием и техническими средствами обучения.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечены доступом в электронную информационно-образовательную среду.

Программа составлена в соответствии с требованиями ОС ННГУ по направлению 01.04.02 Прикладная математика и информатика.

Автор: В.В. Рябов

Заведующий кафедрой Р.Г. Стронгин

Программа одобрена на заседании методической комиссии института информационных технологий, математики и механики от 30 ноября 2022 года, протокол № 3.