

**Федеральное государственное бюджетное образовательное учреждение
высшего образования
Университет «Дубна»
(государственный университет «Дубна»)**

Институт системного анализа и управления
Кафедра распределенных информационно-вычислительных систем

УТВЕРЖДАЮ

и.о. проректора по учебно-методической работе

_____ / Анисимова О.В.

_____ 202_ год

Рабочая программа дисциплины

Введение в программирование

Направление подготовки

09.03.01 Информатика и вычислительная техника

Уровень высшего образования

бакалавриат

Направленность (профиль) программы

Технологии разработки программного обеспечения

Форма(ы) обучения

очная

Дубна, 2023

Преподаватель (преподаватели):

Беднякова Т.М.

*Фамилия И.О., должность, ученая степень (при наличии),
ученое звание (при наличии), кафедра*

подпись

Рабочая программа разработана в соответствии с требованиями ФГОС ВО по направлению подготовки высшего образования

(код и наименование направления подготовки (специальности))

Программа рассмотрена на заседании кафедры

(название кафедры)

Протокол заседания № _____ от «_____» _____ 20__ г.

Заведующий кафедрой _____

(Фамилия И.О., подпись)

СОГЛАСОВАНО

Заведующий выпускающей кафедрой _____

(Фамилия И.О., подпись)

«_____» _____ 20__ г.

Эксперт (рецензент):

*(Ф.И.О., ученая степень, ученое звание, место работы, должность; если текст рецензии не прикладывается –
подпись эксперта (рецензента), заверенная по месту работы)*

1. Цели и задачи освоения дисциплины

Цель изучения дисциплины «Введение в программирование» соотносится с общими целями образовательной программы (далее – ОПОП ВО) по направлению подготовки 09.03.01 Информатика и вычислительная техника.

Быстрое восприятие и обработка больших объемов информации требуют от современного человека хорошего логического мышления. Обучение программированию позволяет разработать и развить эту форму мышления. Помимо этого, профессиональная подготовка современного специалиста в области информационных технологий требует понимания процессов разработки программного обеспечения, умения определять необходимые структуры данных и алгоритмы их обработки. Таким образом, целями освоения данной дисциплины являются:

1. развитие у студентов алгоритмического мышления в степени, необходимой для быстрого и полного освоения компьютерных технологий;
2. развитие способности видеть и формулировать задачи новых применений компьютера в будущей профессиональной деятельности тем;
3. формирование профессиональных компетенций в области разработки информационных систем в соответствии с требованиями ФГОС ВО по данному направлению подготовки 09.03.01 Информатика и вычислительная техника;
4. обучение методике оценки качества функционирования разработанной программы;
5. разъяснение понимания места и роли программирования в будущей профессиональной деятельности.

Объектами профессиональной деятельности в рамках изучаемой дисциплины (модуля) являются: программное обеспечение средств вычислительной техники и автоматизированных систем (программы, программные комплексы и системы); математическое, информационное, техническое, лингвистическое, программное, эргономическое, организационное и правовое обеспечение перечисленных систем.

2. Место дисциплины в структуре ОПОП

Дисциплина «Введение в программирование» относится к обязательная дисциплина базовой части учебного плана направления 09.03.01 Информатика и вычислительная техника.

Дисциплина преподается в 1 семестре, на 1 курсе.

Программа опирается на знания по арифметике, алгебре, теории элементарных функций и геометрии в объеме средней школы, а также на знание областей применения компьютера и основ информатики на основе школьного курса информатики.

3. Планируемые результаты обучения по дисциплине (модулю)

Формируемые компетенции (код и наименование)	Индикаторы достижения компетенций (код и формулировка)	Планируемые результаты обучения по дисциплине (модулю)
ОПК-8. Способен разрабатывать алгоритмы и программы, пригодные для практического применения.	И(ОПК-8)-1 Формализует и предлагает алгоритмическое решение поставленной задачи.	<u>Знать:</u> методы формализации задач, базовые алгоритмы, пригодные для практического применения. <u>Уметь:</u> формализовать задачи и предлагать алгоритмы, пригодные для практического применения. <u>Владеть:</u> основными способами записи алгоритмов.
	И(ОПК-8)-2 Разрабатывает программы, пригодные для практического использования.	<u>Знать:</u> языки программирования. <u>Уметь:</u> составлять алгоритмы, писать и отлаживать коды на языке программирования, тестировать работоспособность программы. <u>Владеть:</u> средствами для написания программного кода, методами и инструментами отладки и тестирования работоспособности программы.

4. Объем дисциплины

Объем дисциплины составляет 4 зачетных единиц, всего 95 академических часов.

5. Содержание дисциплины

Наименование разделов и тем дисциплины	Всего(академ. часы)	в том числе:			
		Контактная работа (работа во взаимодействии с преподавателем)			Самостоятельная работа обучающегося
		Лекции	Практические (семинарские) занятия	Всего	
История развития программирования.	2	1	0	1	1
Алгоритмы: свойства, типы, способы записи.	5	1	2	3	2

Знакомство с языком программирования Python.	4	1	1	2	2
Переменные, как способ хранения данных. Типизация в языках программирования.	4	1	1	2	2
Арифметические операции.	4	1	1	2	2
Условные алгоритмы. Логические выражения и операторы.	4	1	1	2	2
Циклические алгоритмы, цикл с условием while.	4	1	1	2	2
Функции в программировании.	4	1	1	2	2
Цикл for, списки. Алгоритмы обработки списков.	4	1	1	2	2
Алгоритмы сортировки.	4	1	1	2	2
Оценка сложности алгоритма.	4	1	1	2	2
Понятие рекурсии, примеры рекурсивных алгоритмов.	4	1	1	2	2
Модуль для работы с графикой Turtle (Python).	4	1	1	2	2
Введение в ООП, понятия объект, класс. Взаимоотношения между классами.	4	1	1	2	2
Стек, алгоритмы на основе стека, очередь.	4	1	1	2	2
Строки, операции над строками. Алгоритмы обработки строк.	4	1	1	2	2
Отладка программ.	4	1	1	2	2

Чтение и запись данных в файл.	4	1	1	2	2
Структуры данных: кортежи, множества и словари.	4	1	1	2	2
Функции как объект. Лямбда функции.	4	1	1	2	2
Определение операторов. Переопределения функций и декораторы. 'Магические функции'.	4	1	1	2	2
Система контроля версий Git, назначение, порядок работы	4	1	1	2	2
Тестирование приложений.	4	1	1	2	2
Пример создания приложения.	4	1	1	2	2
Итого за семестр / курс	95	24	24	48	47

Содержание дисциплины

- История развития программирования.
- Алгоритмы: свойства, типы, способы записи.
Понятие и свойства алгоритмов. Линейные, условные и циклические алгоритмы. Способы записи алгоритмов. Примеры алгоритмов.
- Знакомство с языком программирования Python. Консольный ввод и вывод данных. Среда разработки. Особенности синтаксиса.
- Переменные, как способ хранения данных. Типизации в языках программирования. Типы данных. Преобразование типов. Хранение данных в памяти компьютера. Ссылочный тип данных и тип значение.
- Арифметические операции. Основы работы с вещественными числами, округление.
- Условные алгоритмы. Логические выражения и операторы. Синтаксис условных операторов в Python.
- Циклические алгоритмы, цикл с условием while. Алгоритмы вычисления суммы ряда: с заданной точностью, рекуррентного ряда.
Алгоритмы обработки числовых последовательностей. Алгоритм Евклида. Подсчет количества цифр в числе. Проверка, является ли число простым.

- Функции в программирование. Локальный и глобальный контекст. Параметры функций.
- Цикл for, списки. Операции над списками. Алгоритмы обработки списков. Поиск минимального (максимального) значения в списке. Алгоритмы обработки списков (вычисление суммы, подсчет количества элементов, удовлетворяющих заданному условию). Алгоритмы обработки списков, с учетом дополнительного условия. Алгоритм перемешивание массива. Алгоритмы поиска простых чисел (Решето Эратосфена)
- Алгоритмы сортировки: выбора, пузырек, улучшенный пузырек, шейкер, вставками, Шелла, быстрая сортировка.
- Эффективность программы, алгоритма. Оценка сложности алгоритма.
- Понятие рекурсии, примеры рекурсивных алгоритмов: Рекурсивные алгоритмы: вычисление факториала, кривая дракона, ханойские башни, числа Фибоначчи.
- Модуль для работы с графикой Turtle (Python).
- Введение в ООП, понятия объект, класс. Взаимоотношения между классами. Основные принципы ООП: инкапсуляция. полиморфизм, наследование.
- Стек, алгоритмы на основе стека, очередь.
- Строки, операции над строками. Алгоритмы обработки строк.
- Ошибки и исключения. Обработка исключений.
- Отладка программ.
- Чтение и запись данных в файл.
- Структуры данных: кортежи, множества и словари.
- Функции как объект. Лямбда функции. Работа со списками без использования операторов цикла.
- Определение операторов. Переопределения функций и декораторы. “Магические функции”.
- Система контроля версий Git, назначение, порядок работы.
- Пример создания приложения.
- Тестирование приложений.

При реализации дисциплины организуется практическая подготовка путем проведения практических занятий, предусматривающих участие обучающихся в выполнении отдельных элементов работ, связанных с будущей профессиональной деятельностью (68 часов).

Практическая подготовка также включает в себя отдельные занятия лекционного типа, которые предусматривают передачу учебной информации обучающимся, необходимой для последующего выполнения работ, связанных с будущей профессиональной деятельностью (34 часа).

Практическая подготовка при изучении дисциплины реализуется непосредственно в университете.

6. Перечень учебно-методического обеспечения по дисциплине

Для обеспечения реализации программы дисциплины разработаны:

1. методические материалы к практическим (семинарским) и к лекционным занятиям;
2. методические материалы по организации самостоятельной работы обучающихся;
3. методические материалы по организации изучения дисциплины с применением электронного обучения, дистанционных образовательных технологий;
4. методические рекомендации для обучающихся с ограниченными возможностями здоровья и инвалидов по освоению программы дисциплины.
5. Методические материалы по дисциплине (модулю) и образовательной программе в целом представлены на официальном сайте образовательной организации (раздел «Сведения об образовательной организации» – Образование – Образовательные программы).

7. Фонды оценочных средств по дисциплине

Для аттестации обучающихся на соответствие их персональных достижений поэтапным требованиям образовательной программы по дисциплине (модулю) разработаны фонды оценочных средств, позволяющие оценить результаты обучения (знания, умения, навыки) и сформированные (формируемые) компетенции. Эти фонды включают теоретические вопросы, типовые практические задания, контрольные работы, домашние работы, тесты и иные оценочные материалы, используемые при проведении процедур текущего контроля успеваемости и промежуточной аттестации. Фонды оценочных средств представлены в приложении к рабочей программе.

При необходимости обучающиеся с ограниченными возможностями здоровья и инвалиды обеспечиваются оценочными материалами в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

1. в печатной форме увеличенным шрифтом,
2. в форме электронного документа,
3. в форме аудиофайла,
4. в печатной форме на языке Брайля.

Для лиц с нарушениями слуха:

1. в печатной форме,
2. в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

1. в печатной форме,
2. в форме электронного документа,
3. в форме аудиофайла.

8. Ресурсное обеспечение

Перечень литературы

Основная литература

1. Бэрри, П. Изучаем программирование на Python / Бэрри Пол, перевод с английского М. А. Райтман. — М.: Э, 2018. — 624 с.: ил. — (Мировой компьютерный бестселлер). — ISBN 978-5-699-98595-1.
2. Гэддис, Т. Начинаем программировать на Python / Гэддис Тони. — 4-е изд. — СПб.: БХВ-Петербург, 2019. — 768 с.: ил. — Предм.указ.: с. 739. — ISBN 978-5-9775-4002-5.
3. Лутц М. Python: карманный справочник / Лутц Марк. — 5-е изд. — М.: Вильямс, 2019. — 320 с.: ил. — Предм.указ.: с. 313. — ISBN 978-5-907114-60-9.
4. Вайсфельд, М. Объектно-ориентированное мышление / Вайсфельд Мэтт. — СПб.: Питер, 2014. — 304 с.: ил. — (Библиотека программиста). — ISBN 9785496007931

Дополнительная литература

1. Кнут, Д.Э. Искусство программирования. Т. 1. Основные алгоритмы. / Д.Э. Кнут. — М.: Вильямс, 2016. — 720 с.
2. Кнут, Д.Э. Искусство программирования. Т.3. Сортировка и поиск. / Д.Э. Кнут. — М.: Вильямс, 2014. — 832 с.
3. Макконнелл, С. Совершенный код. Мастер-класс / Стив Макконнелл. — СПб.: БХВПетербург, 201 . — 896 с.— ISBN: 978-5-9909805-1-8
4. Бэрри, П. Изучаем программирование на Python / Пол Бэрри.— М.: Эксмо, 2019. — 624 с.— 978-5-699-98595-1
5. Бейдер, Д. Чистый Python. Тонкости программирования для профи / Дэн Бейдер. — СПб.: Питер, 2018. — 288 с.: ил. — (Серия «Библиотека программиста»).— ISBN 978-5-4461-0803-9

Профессиональные базы данных и информационные справочные системы

Профессиональные ресурсы сети «Интернет»

1. Электронно-библиотечная система «Университетская библиотека онлайн»: www.bibloclub.ru
2. Электронно-библиотечная система «ЮРАЙТ» <https://biblio-online.ru>

3. Научная электронная библиотека (НЭБ): <http://elibrary.ru/defaultx.asp>
4. Электронно-библиотечная система «Znanium» <http://znanium.com/>
5. БД российских журналов East View : <http://dlib.eastview.com>
6. Базы данных компании EBSCO Publishing: <http://search.ebscohost.com/>
7. БД российских научных журналов на Elibrary.ru (РУНЭБ): <http://elibrary.ru/defaultx.asp>.
8. <http://www.scopus.com/home.url>
9. Web of Science [webofknowledge.com](http://www.webofknowledge.com)
10. Материалы <https://habrahabr.ru/flows/develop/>
11. Материалы Национального открытого университета ИНТУИТ: http://www.intuit.ru/studies/higher_education
12. Сообщество аналитиков: <http://www.uml2.ru/>.

Необходимое программное обеспечение

- Visual Studio Code (1.57.0)
- PyCharm (2021.1.1)

Необходимое материально-техническое обеспечение

Для проведения лекционных занятий требуется специализированная аудитория, оснащенная компьютером, проектором, электронной доской.

Для проведения семинарских занятий требуется специализированный компьютерный класс, подключенный к сети Интернет и обеспечением доступа в электронную информационно-образовательную среду организации.

Обучающиеся из числа инвалидов и лиц с ограниченными возможностями здоровья могут использовать специализированное программное и материально-техническое обеспечение:

- обучающиеся с нарушениями опорно-двигательного аппарата при необходимости могут использовать адаптивные технические средства: функцию «сенсорная клавиатура», «управление указателем мыши с клавиатуры», специально оборудованные джойстики, увеличенные выносные кнопки, клавиатуры с большими клавишами или накладки «Клавиата»;
- обучающиеся с ограничениями по зрению могут прослушать доступный аудиоматериал или прочесть тексты, увеличив шрифт на экране монитора компьютера. Рекомендуется использовать экранную лупу и другие визуальные вспомогательные средства, чтобы изменить шрифт текста, межстрочный интервал, синхронизацию с речью и т.д., программы экранного доступа (скринридеры для прочтения текстовой информации через синтезированную речь) и/или включить функцию «экранного диктора» на персональном компьютере с операционной

системой Windows 7, 8, 10, Vista, XP. Студенты с полным отсутствием зрения могут использовать тексты, напечатанные шрифтом Брайля, а для набора текста на компьютере – клавиатуры Брайля;

- обучающиеся с ограничениями по слуху могут воспользоваться индивидуальными техническими средствами (аппараты «Глобус», «Монолог», индивидуальными слуховыми аппаратами, компьютерной аудиогарнитурой, наушниками и др.) при прослушивании необходимой информации, а также услугами сурдопереводчика

При необходимости обучающиеся с ограниченными возможностями здоровья и инвалиды обеспечиваются печатными и (или) электронными образовательными ресурсами (образовательная программа, учебники, учебные пособия и др.) в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

1. в печатной форме увеличенным шрифтом,
2. в форме электронного документа,
3. в форме аудиофайла,
4. в печатной форме на языке Брайля.

Для лиц с нарушениями слуха:

1. в печатной форме,
2. в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

1. в печатной форме,
2. в форме электронного документа,
3. в форме аудиофайла.