

Министерство просвещения Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Российский государственный профессионально-педагогический университет»
Институт инженерно-педагогического образования
Кафедра информационных систем и технологий

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Б1.О.06.01 «АЛГОРИТМИЧЕСКИЕ ЯЗЫКИ И СИСТЕМЫ
ПРОГРАММИРОВАНИЯ»**

Направление подготовки 09.03.03 Прикладная информатика

Профиль программы «Прикладная информатика (по элективным модулям)»

Автор(ы): канд. пед. наук, доцент Т.П. Телепова

Одобрена на заседании кафедры информационных систем и технологий. Протокол от «20» января 2022 г. №5.

Рекомендована к использованию в образовательной деятельности научно-методической комиссией института ИПО РГППУ. Протокол от «26» января 2022 г. №6.

Екатеринбург
2022

1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Цель освоения дисциплины «Алгоритмические языки и системы программирования»: формирование у студентов знаний и умений, необходимых для разработки программных приложений, в том числе программных прототипов решения прикладных задач. Дисциплина закладывает основы профессионального подхода к программному решению практических задач.

Задачи:

- развитие навыков поиска, критического анализа и синтеза информации, самостоятельного использования технической литературы по вопросам разработки программного обеспечения;
- формирование понимания принципов работы современных информационных технологий и программных средств, системного подхода, умений реализации типовых алгоритмов и структур данных и их модификации на выбранном языке программирования;
- накопление практического опыта разработки эффективных алгоритмов обработки информации и программ, пригодных для практического применения;
- систематизация и расширение знаний о приемах и методах программирования, подготовка к их осознанному использованию при решении различного вида прикладных задач;
- развитие навыков принятия участия в реализации профессиональных коммуникаций с заинтересованными участниками проектной деятельности и в рамках проектных групп.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Дисциплина «Алгоритмические языки и системы программирования» относится к обязательной части учебного плана.

Перечень учебных дисциплин, для которых необходимы знания, умения и владения, формируемые данной учебной дисциплиной:

1. Технологии программирования.
2. Программная инженерия.

3. РЕЗУЛЬТАТЫ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Дисциплина направлена на формирование следующих компетенций:

- ОПК-2 Способен понимать принципы работы современных информационных технологий и программных средств, в том числе



отечественного производства, и использовать их при решении задач профессиональной деятельности;

- ОПК-7 Способен разрабатывать алгоритмы и программы, пригодные для практического применения;

- ОПК-9 Способен принимать участие в реализации профессиональных коммуникаций с заинтересованными участниками проектной деятельности и в рамках проектных групп;

- УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач.

В результате освоения дисциплины (модуля) обучающийся должен:

Знать:

31. Основные принципы работы современных информационных технологий и программных средств и использовать их при решении задач профессиональной деятельности;

32. Возможности современных сред разработки приложений и технологий языков программирования.

Уметь:

У1. Разрабатывать алгоритмы и программы, пригодные для практического применения;

У2. Реализовать готовые алгоритмы на языке высокого уровня;

У3. Настраивать среду разработки, работать с применением технологии проектного программирования;

У4. Применять разработанное и протестированное программное обеспечение, библиотеки и функции, во вновь разрабатываемом приложении;

У5. Создавать программный код, соответствующий современным требованиям к его оптимальности, структуре, оформлению и устойчивости к различным видам ошибок.

Владеть:

В1. Методами поиска, критического анализа и синтеза информации, системным подходом для решения поставленных задач;

В2. Способностью принимать участие в реализации профессиональных коммуникаций с заинтересованными участниками проектной деятельности и в рамках проектных групп;

В3. Технологией отладки программного кода;

В4. Технологией использования справочной системы для поиска необходимых методов встроенных классов при решении задач.



4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1 Объем дисциплины и виды контактной и самостоятельной работы

Общая трудоёмкость дисциплины составляет 7 зач. ед. (252 час.), семестры изучения – 1, 2, распределение по видам работ представлено в табл. № 1.

Таблица 1. Распределение трудоемкости дисциплины по видам работ

Вид работы	Форма обучения
	очная
	Семестр изучения
	1, 2 сем.
	Кол-во часов
Общая трудоемкость дисциплины по учебному плану	252
Контактная работа, в том числе:	94
Лекции	16
Лабораторные работы	78
Самостоятельная работа студента	158
Промежуточная аттестация, в том числе:	
Зачет	1 сем.
Экзамен	2 сем.

**Распределение трудоемкости по видам контактной работы для заочной формы обучения (при наличии) корректируется в соответствии с учебным планом заочной формы обучения.*

4.2 Содержание и тематическое планирование дисциплины

Таблица 2. Тематический план дисциплины

Наименование разделов и тем дисциплины (модуля)	Сем.	Всего, час.	Вид контактной работы, час.			СРС
			Лекции	Практ. занятия	Лаб. работы	
1. Введение в язык программирования	1	22	2	-	10	10
2. Базовые основы программирования	1	60	4	-	16	40
3. Стандартная библиотека, графические пакеты языка программирования	1,2	66	4	-	22	40
4. Основы объектно-ориентированного	2	50	2	-	16	32



программирования						
5. Технология разработки графического интерфейса	2	54	4	-	14	36

**Распределение часов по разделам (темам) дисциплины для заочной формы обучения осуществляется научно-педагогическим работником, ведущим дисциплину.*

4.3 Содержание разделов (тем) дисциплин

Раздел 1. Введение в язык программирования

Основные характеристики языка Java. Java-платформы. Пакет программного обеспечения Java Development Kit. Установка и настройка JDK. Понятие интегрированной среды разработки приложений. IDE Eclipse для Java. Создание, компиляция и выполнение Java-программы. Первая программа на Java. Иерархическая структура программного кода Java. Комментарии. Ключевые слова и идентификаторы. Тип данных, литералы, переменные. Примитивные типы данных Java, их преобразование и приведение. Ссылочные типы данных. Преобразование и приведение примитивных типов. Операции над целочисленными операндами.

Раздел 2. Базовые основы программирования

Структура простой программы на Java. Пакеты классов и инструкция package. Обращение в программе к внешним классам и пакетам, директива import. Модификаторы доступа. Этапы разработки программы. Консольный ввод данных в программу. Соглашение по именованию переменных, констант, классов и методов в программе. Идентификаторы. Переменные. Предложения и выражения с присваиванием. Именованные константы. Вычисление выражений и приоритет операторов. Составные операторы присваивания, операторы инкремента и декремента, управляющие последовательности. Структуры выбора if/else. Логические операторы и структура выбора switch/case. Циклы.

Раздел 3. Стандартная библиотека, графические пакеты языка программирования

Организация библиотеки стандартных классов Java. Пакеты, классы. Вызов класса из пакета. Пакет java.lang. Класс Math пакета lang. Графический пакет java.awt. Графическая библиотека Swing. Графическая платформа JavaFX. Классы Graphics и Graphics2D. Графический контекст. Методы класса Graphics. Методы работы с цветом. Основы оконной графики. Методы класса Frame. Визуализация графики в окне. Создание графических рисунков.

Раздел 4. Основы объектно-ориентированного программирования

Основные принципы объектно-ориентированного программирования. Ссылочные типы данных. Переменные ссылочного типа. Доступ к объектам с помощью ссылочных переменных. Создание ссылочных переменных. Поля



данных ссылочного типа и значение null. Различия между переменными примитивных и ссылочных типов. Присвоение значений переменным ссылочных типов. Определение классов и объектов. Методы. Конструкторы. Способы написания классов.

Раздел 5. Технология разработки графического интерфейса

Особенности пользовательского интерфейса и требования к нему. Представление о событийном программировании. Свойства, события. Панель компонентов. Компоненты, позволяющие реализовать ввод и вывод данных, выбор значений из списков, управление (флажки и селекторы), вывод табличных данных. Свойства компонентов, обеспечивающих удобство работы пользователя (управление фокусом, видимость и активность элементов, оформление). Обеспечение контроля вводимых значений. Работа с диалоговыми окнами (открытие и сохранение файлов, выбор цвета, характеристик шрифта и т.д.). Использование стандартных элементов пользовательского интерфейса: меню, строка статуса, контекстное меню.

5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Для изучения дисциплины используются различные образовательные технологии:

1. Традиционные образовательные технологии представлены комбинацией объяснительно-иллюстративного и репродуктивного методов обучения. Осуществляются с использованием информационных лекций, семинаров, практических занятий или лабораторных работ. При использовании данных методов деятельность учащегося направлена на получение теоретических знаний и формирования практических умений по дисциплине.

2. Для поддержки самостоятельной работы обучающихся использованы информационно-коммуникационные образовательные технологии, в частности, облачные технологии, электронная информационно-образовательная среда (ЭИОС), электронные средства обучения и электронно-библиотечные системы. При этом результативность организации самостоятельной работы обучающихся существенно повышается за счет доступности материалов, упорядоченности работ и возможности получения консультации преподавателя.

3. Технология обучения в сотрудничестве применяются при проведении семинарских, практических и лабораторных занятий, нацелены на совместную работу в командах или группах и достижение качественного образовательного результата.

4. При реализации образовательной программы с применением дистанционных образовательных технологий и электронного обучения:

- состав видов контактной работы по дисциплине (модулю), при необходимости, может быть откорректирован в направлении снижения доли



занятий лекционного типа и соответствующего увеличения доли консультаций (групповых или индивидуальных) или иных видов контактной работы;

- информационной основой проведения учебных занятий, а также организации самостоятельной работы обучающихся по дисциплине (модулю) являются представленные в электронном виде методические, оценочные и иные материалы, размещенные в электронной информационно-образовательной среде (ЭИОС) университета, в электронных библиотечных системах и открытых Интернет-ресурсах;

- взаимодействие обучающихся и педагогических работников осуществляется с применением ЭИОС университета и других информационно-коммуникационных технологий (видеоконференцсвязь, облачные технологии и сервисы, др.);

- соотношение контактной и самостоятельной работы по дисциплине (модулю) может быть изменено в сторону увеличения последней, в том числе самостоятельного изучения теоретического материала.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ

6.1 Основная литература

1. Джошуа Блох Java. Эффективное программирование : практическое пособие. - Саратов : Профобразование, 2017. - 310 с. - Режим доступа: <http://www.iprbookshop.ru/64057>.

2. Мейер, Б. Объектно-ориентированное программирование и программная инженерия / Б. Мейер. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ) ; Саратов : Ай Пи Эр Медиа, 2019. — 285 с. — Режим доступа: <http://www.iprbookshop.ru/79706.html>.

3. Букунов С. В., Букунова О. В. Основы объектно-ориентированного программирования : учебное пособие. - Санкт-Петербург : Санкт-Петербургский государственный архитектурно-строительный университет, 2017. - 196 с. - Режим доступа: <http://www.iprbookshop.ru/74339>.

4. Богачёв К. Ю. Основы параллельного программирования : учебное пособие / Богачёв К. Ю. — Москва : Лаборатория знаний, 2020. — 345 с. — Режим доступа: <https://e.lanbook.com/book/135516>.

6.2 Дополнительная литература

1. Задачи по программированию [Электронный ресурс] : учеб. пособие / С.М. Окулов [и др.]. — Электрон. дан. — Москва : Издательство «Лаборатория знаний», 2017. — 826 с. — Режим доступа: <https://e.lanbook.com/book/94162>. — Загл. с экрана.

2. Быкадорова Е. А. Программирование. Практикум : учебное пособие / Быкадорова Е. А. — Санкт-Петербург : Лань, 2020. — 60 с. — Режим доступа: <https://e.lanbook.com/book/139323>.



3. Гаврилов А. В., Клименков С. В., Харитонов А. Е., Цопа Е. А. Программирование на языке Java : учебное пособие. - Санкт-Петербург : Университет ИТМО, 2015. - 123 с. - Режим доступа: <http://www.iprbookshop.ru/68692>.

4. Грацианова, Т.Ю. Программирование в примерах и задачах [Электронный ресурс] : учеб. пособие — Электрон. дан. — Москва : Издательство «Лаборатория знаний», 2016. — 373 с. — Режим доступа: <https://e.lanbook.com/book/90242>. — Загл. с экрана.

6.3 Программное обеспечение и Интернет-ресурсы

Интернет-ресурсы:

1. Интернет-издание о высоких технологиях. Режим доступа: <http://www.cnews.ru/reviews/free/hardnsoft/service/>

2. Научная электронная библиотека eLIBRARY. Режим доступа: <https://elibrary.ru/defaultx.asp>

3. Российская государственная библиотека. Режим доступа: <https://www.rsl.ru/>

Программное обеспечение:

1. Операционная система Windows.
2. Офисная система Office Professional Plus.
3. Среда разработки Visual Studio.

Информационные системы и платформы:

1. Система дистанционного обучения «Moodle».
2. Информационная система «Таймлайн».
3. Платформа для организации и проведения вебинаров «Mirapolis Virtual Room».

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Перечень материально-технического обеспечения для реализации образовательного процесса по дисциплине:

1. Учебная аудитория для проведения занятий лекционного типа.
2. Учебная аудитория для проведения занятий семинарского (практического) типа, проведения групповых и индивидуальных консультаций, проведения текущего контроля и промежуточной аттестации.
3. Компьютерный класс.
4. Помещения для самостоятельной работы.

