



Основная профессиональная образовательная программа
09.03.03 Прикладная информатика
(Прикладная информатика в цифровой экономике)

Министерство науки и высшего образования Российской Федерации
ИВАНОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Кафедра информационных технологий и прикладной математики

ОДОБРЕНО:
Руководитель ОП
_____ С.В. Данилова
(подпись)

30 августа 2024 г.

Рабочая программа дисциплины
Языки программирования

Уровень высшего образования:	бакалавриат
Квалификация выпускника:	бакалавр
Направление подготовки:	09.03.03 Прикладная информатика
Направленность (профиль) образовательной программы:	Прикладная информатика в цифровой экономике

Иваново



Основная профессиональная образовательная программа
09.03.03 Прикладная информатика
(Прикладная информатика в цифровой экономике)

1. Цели освоения дисциплины

Целью дисциплины «Языки программирования» является формирование у будущих специалистов знаний и умений, соответствующих следующим трудовым функциям, входящим в состав обобщенных трудовых функций "Разработка и отладка программного кода" и "Разработка требований и проектирование программного обеспечения" профессионального стандарта "Программист":

- написание программного кода с использованием языков программирования, определения и манипулирования данными;
- анализ требований к программному обеспечению.
- проектирование программного обеспечения.

2. Место дисциплины в структуре ОП

Дисциплина «Языки программирования» относится к обязательным дисциплинам блока 1 «Дисциплины (модули)».

Требования к входным знаниям и умениям студента – знание информатики в объеме средней школы.

Успешное освоение данной дисциплины будет способствовать готовности студентов к освоению дисциплин: Базы данных, Технологии разработки программного обеспечения, Программная инженерия экономических информационных систем, Математическое и имитационное моделирование, Web-программирование, Электронный документооборот на предприятии, Корпоративные информационные системы, Технологии искусственного интеллекта, Цифровая экономика, прохождению производственной практики, выполнению выпускной аттестационной работы бакалавра.

3. Планируемые результаты обучения по дисциплине

3.1. Компетенции, формированию которых способствует дисциплина

б) общепрофессиональные (ОПК):

- 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.

в) профессиональные (ПК):

- ПК-2 Способен разрабатывать и адаптировать прикладное программное обеспечение.

3.2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с индикаторами достижения формируемых компетенций

В результате освоения дисциплины обучающийся должен:

Знать:

- методы и приемы алгоритмизации поставленных задач (ОПК-7);
- нотации и программные продукты для графического отображения алгоритмов (ОПК-7; ОПК-2);
- алгоритмы решения типовых задач, области и способы их применения (ОПК-7; ПК-2);
- синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования (ОПК-7);
- технологии программирования (ОПК-2; ОПК-7; ПК-2);
- особенности выбранной среды программирования (ОПК-2);
- инструментарий для создания и актуализации исходных текстов программ (ОПК-2; ПК-2);



Основная профессиональная образовательная программа
09.03.03 Прикладная информатика
(Прикладная информатика в цифровой экономике)

- методы повышения читаемости программного кода (ОПК-7).

Уметь:

- использовать методы и приемы алгоритмизации поставленных задач (ОПК-7);
- использовать программные продукты для графического отображения алгоритмов (ОПК-7; ОПК-2);
- применять стандартные алгоритмы в соответствующих областях (ОПК-7; ПК-2);
- применять выбранные языки программирования для написания программного кода (ОПК-7; ПК-2);
- использовать выбранную среду программирования (ОПК-2; ПК-2);
- применять нормативные документы, определяющие требования к оформлению программного кода (ОПК-7);
- применять инструментарий для создания и актуализации исходных текстов программ (ОПК-2; ОПК-7; ПК-2).

Иметь:

- навыки разработки алгоритмов решения поставленных задач в соответствии с требованиями технического задания (ОПК-2; ПК-2);
- навыки создания программного кода в соответствии с техническим заданием (готовыми спецификациями) (ОПК-7; ПК-2);
- навыки приведения наименований переменных, функций, классов, структур данных и файлов в соответствие с установленными требованиями (ОПК-7; ПК-2);
- навыки структурирования исходного программного кода в соответствии с установленными требованиями (ОПК-7; ПК-2);
- навыки комментирования и разметки программного кода в соответствии с установленными требованиями (ОПК-7; ПК-2);
- навыки форматирования исходного программного кода в соответствии с установленными требованиями (ОПК-7; ПК-2).

4. Объем и содержание дисциплины

Объем дисциплины составляет 8 зачетных единиц (288 академических часов).

4.1. Содержание дисциплины по разделам (темам), соотнесенное с видами и трудоемкостью занятий лекционно-семинарского типа

Объем иной контактной работы и самостоятельной работы обучающегося по дисциплине указан в учебном плане образовательной программы.

№ п/п	Разделы (темы) дисциплины	Семестр	Виды занятий, их объем (в ак. часах, по очной форме обучения)		Формы текущего контроля успеваемости (по очной форме обучения)
			Занятия лекционного типа	Занятия семинарского типа	Формы промежуточной аттестации
1.	Понятия алгоритма и языка программирования.	1	4	2 лабор. зан.	Обсуждение результатов лабораторных работ.
2.	Данные и память. Абстракция данных.	1	4	2 лабор. зан.	Обсуждение результатов лабораторных работ.
3.	Язык программирования высокого уровня C / C++ / Python и среда разработки Visual Studio Code.	1	4	4 лабор. зан.	Обсуждение результатов лабораторных работ.



Основная профессиональная образовательная программа
09.03.03 Прикладная информатика
(Прикладная информатика в цифровой экономике)

4.	Типы данных. Ключевые слова. Литералы в программировании.	1	4	4 лабор. зан	Обсуждение результатов лабораторных работ.
5.	Операции и операторы языка программирования C / C++ / Python.	1	4	4 лабор. зан	Обсуждение результатов лабораторных работ.
6.	Программирование алгоритмов циклической структуры. Итерационные циклы.	1	4	4 лабор. зан	Обсуждение результатов лабораторных работ.
7.	Массивы: одномерные, двумерные (матрицы), многомерные.	1	4	4 лабор. зан	Обсуждение результатов лабораторных работ.
8.	Указатели. Операции над указателями. Указатели на массивы. Массивы указателей.	1	4	4 лабор. зан	Обсуждение результатов лабораторных работ.
9.	Функции, определяемые программистом, в языке C / C++ / Python.	1	4	4 лабор. зан	Обсуждение результатов лабораторных работ.
Итого за семестр:			36	32	Экзамен
10.	Классы в C++ / Python.	2	4	4 лабор. зан	Обсуждение результатов лабораторных работ.
11.	Статические и динамические структуры данных.	2	4	4 лабор. зан	Обсуждение результатов лабораторных работ.
12.	Принципы объектно-ориентированного программирования.	2	4	4 лабор. зан	Обсуждение результатов лабораторных работ.
13.	Наследование классов.	2	4	4 лабор. зан	Обсуждение результатов лабораторных работ.
14.	Шаблоны в языке C++ / Python.	2	4	4 лабор. зан	Обсуждение результатов лабораторных работ.
15.	Технология разработки объектно-ориентированных программ на C++ / Python.	2	4	4 лабор. зан	Обсуждение результатов лабораторных работ.
16.	Обработка исключительных ситуаций.	2	4	4 лабор. зан	Обсуждение результатов лабораторных работ.
17.	Тестирование и отладка программ.	2	2	2 лабор. зан	Обсуждение результатов лабораторных работ.
18.	Документирование программ.	2	2	2 лабор. зан	Обсуждение результатов лабораторных работ.
Итого за семестр:			32	32	Экзамен
Итого по дисциплине:			68	64	Экзамен, экзамен

4.2. Развернутое описание содержания дисциплины по разделам (темам)

1. Понятия алгоритма и языка программирования.

Предмет и задачи курса. Понятие алгоритма. Свойства алгоритма: дискретность, детерминированность, конечность, массовость. Формы записи алгоритмов. История развития языков программирования. Классификация языков программирования. Программы на языках программирования высокого уровня. Жизненный цикл программы: абстракция, математическая модель, алгоритм, запись алгоритма на языке программирования, трансляция в машинный код, компоновка программы, тестирование и отладка программы, эксплуатация и сопровождение программ.



2. Данные и память. Абстракция данных.

Абстракция данных. Данные и память. Данные в языках программирования. Структуризация данных. Определение типа данных. Статический и динамический контроль типов и классификация языков в соответствии с ним. Слабая, сильная и строгая типизации в статически типизируемых языках. Конструкции структурного программирования и теорема о структурировании. Абстракция управления.

3. Язык программирования высокого уровня C / C++ / Python и среда разработки Visual Studio Code.

Среда разработки Visual Studio Code. Консольные приложения. Структура программы. Оформление исходного кода программы на языке программирования высокого уровня C / C++ / Python. Ключевые слова. Программирование алгоритмов разветвляющейся структуры.

4. Типы данных. Ключевые слова. Литералы в программировании.

Определение типа данных в C / C++ / Python. Стандартные типы и действия с ними. Литералы в программировании. Целочисленные литералы. Строковые литералы. Иерархия типов данных в C / C++ / Python. Преобразования типов. Статический и динамический контроль типов и классификация языков в соответствии с ним. Слабая, сильная и строгая типизации в статически типизируемых языках. Типы данных, определяемые пользователем.

5. Операции и операторы языка программирования C / C++ / Python.

Простые данные и вычисления. Выражения, правила их записи и выполнения. Классификация выражений в зависимости от типа результата. Операции и операторы языка программирования C / C++ / Python: арифметические, битовые, логические операции. Приоритеты выполнения операций. Директивы препроцессора, ввод и вывод в C / C++ / Python. Библиотечные функции в C / C++ / Python.

6. Программирование алгоритмов циклической структуры. Итерационные циклы.

Программирование алгоритмов циклической структуры, организация специального выхода из цикла. Понятие итерационного цикла. Генератор случайных чисел.

7. Массивы: одномерные, двумерные (матрицы), многомерные.

Одномерные массивы. Способы объявления и инициализации элементов массива. Соблюдение границ массива. Алгоритмы обработки числовых массивов с использованием индексов. Двумерный массив. Инициализация и обработка двумерного массива (матрицы). Многомерные массивы. Инициализация многомерных массивов.

8. Указатели. Операции над указателями. Указатели на массивы. Массивы указателей.

Понятие указателя. Операции получения адреса объекта и разыменования указателя. Операции над указателями. Алгоритмы обработки числовых массивов с использованием указателей. Массивы указателей. Ступенчатые массивы. Указатели на массивы.

9. Функции, определяемые программистом, в языке C / C++ / Python.

Функции, определяемые программистом. Описание, схема вызова функции. Формальные и фактические параметры, их взаимодействие. Параметры сложных типов. Механизмы отведения памяти под параметры. Стек. Возвращаемое значение. Механизмы передачи информации из функции и из основной программы. область видимости имен. Локальные и глобальные идентификаторы. Вложенные функции. Функции с параметрами по умолчанию. Перегруженные функции. Использование статических переменных в функции. Понятие рекурсии. Реализация рекурсивных алгоритмов.

10. Классы в C++ / Python.

Определение класса. Личная и общая части определения класса. Объекты класса. Создание и уничтожение объектов класса. Конструкторы и деструкторы. Способы реализации инкапсуляции. Интерфейс и реализация контейнерных классов для моделирования структур



данных. Статические члены объектов класса. Вложенные и локальные классы. Примеры описания и использования классов.

11. Статические и динамические структуры данных.

Статические структуры данных. Описание. Примеры использования. Динамические структуры данных: списки, очереди, деревья. Общие свойства динамических структур данных. Списки: односвязные, двусвязные. Очереди: циклическая очередь. Деревья. Примеры реализации динамических структур данных.

12. Принципы объектно-ориентированного программирования.

Объекты, их методы и свойства. Инкапсуляция. Наследование. Виртуальные методы.

13. Наследование классов.

Базовый и производный классы. Функции-элементы и функции-друзья. Правила доступа к элементам производного класса. Иерархия классов. Одиночное и множественное наследование. Особенности доступа при множественном наследовании. Правила преобразования указателей. Инициализация объектов. Отличия инициализации от присваивания. Инициализация баз и членов. Полный объект конечного производного класса. Примеры описания и использования классов с наследованием.

14. Шаблоны в языке C++ / Python.

Шаблоны классов и функций. Наследование шаблонных классов. Правила отождествления параметров шаблона. Применение шаблонных классов для создания контейнерных классов. Анализ вариантов шаблонных классов на примере шаблона вектора с операцией сортировки.

15. Технология разработки объектно-ориентированных программ на C++ / Python.

Особенности разработки объектноориентированных программ на C++ / Python. Проектирование классов. Механизмы функционирования программ. Стиль программирования. Надежность программ на C++ / Python.

16. Обработка исключительных ситуаций.

Анализ различных моделей обработки исключительных ситуаций. Стандартные средства контроля подтверждений. Проверка предусловий и постусловий, вычисления инвариантов. Контроль асинхронных событий. Реализация модели обработки синхронных ситуаций с завершением в C++. Возбуждение ситуации, Описание блоков с контролем и реакций на ситуации. Система классов для описания исключительных ситуаций. Обработка исключительных ситуаций при обработке исключительных ситуаций. Примеры контроля ситуаций в контейнерных классах.

17. Тестирование и отладка программ.

Общие положения: цели, объекты и проблемы тестирования. Понятия тестов, тестовых наборов и тестовых сценариев. Понятие формальной верификации алгоритмов и программ. Критерии качества тестирования: полнота покрытия операторов, маршрутов и данных. Виды тестирования: автономное и комплексное тестирование; тестирование белого и черного ящика; регрессионное и нагрузочное тестирование. Некоторые методы тестирования: инспекция кода, метод многократной разработки, метод эквивалентов и граничных условий. Средства автоматизации тестирования.

18. Документирование программ.

Общие положения: цели и задачи документирования. Основные виды программных и эксплуатационных документов. Принципы создания руководства пользователя.

5. Образовательные технологии

Организация учебного процесса осуществляется в форме лекций, лабораторных занятий и индивидуальной самостоятельной работы студентов.



Основная профессиональная образовательная программа
09.03.03 Прикладная информатика
(Прикладная информатика в цифровой экономике)

Учебный процесс по дисциплине «Языки программирования» основан на использовании следующих инновационных образовательных технологий:

1. Технология проблемного обучения – основные темы курса на лекциях и лабораторных занятиях раскрываются через постановку и последующее разрешение проблемы создания алгоритма решения задачи и ее разрешение в виде функционирующей программы.
2. Технология тестового контроля качества образования – в процессе и по завершении теоретического обучения выполняется компьютерное тестирование.
3. Информационно-компьютерные технологии – применяются при выполнении лабораторных работ, самостоятельной внеаудиторной подготовке в виде самотестирования по сети Internet и использования учебных материалов в электронной форме.
4. Технология смешанного обучения.

6. Учебно-методическое обеспечение самостоятельной работы обучающихся

Методика преподавания учебной дисциплины решает следующие основные задачи:

- определяет задачи обучения студентов по дисциплине;
- научно обосновывает содержание учебной программы, намечает последовательность ее изучения в комплексе с другими дисциплинами;
- определяет пути реализации принципов обучения при изучении дисциплины, формы и методы обучения;
- вырабатывает требования к методической подготовке преподавателей;
- изучает историю методики преподавания дисциплины;
- внедряет передовой опыт обучения;
- вырабатывает рекомендации по воспитанию обучаемых в процессе изучения дисциплины.

В соответствии с этими задачами осуществляется отбор научного материала, его систематизация и переработка в интересах развития и совершенствования содержания учебной дисциплины.

Методика разработана применительно к утвержденной рабочей программе для студентов с учетом требований Государственного образовательного стандарта высшего образования по направлению подготовки 09.03.03 «Прикладная информатика», и вооружает преподавателей необходимыми знаниями, способствует их внедрению в практику обучения и воспитания студентов.

Выбор методов проведения занятий обусловлен учебными целями, содержанием учебного материала, временем, отводимым на занятия.

На занятиях в тесном сочетании применяется несколько методов, один из которых выступает ведущим. Он определяет построение и вид занятий.

На лекциях излагаются лишь основные, имеющие принципиальное значение и наиболее трудные для понимания и усвоения теоретические и практические вопросы.

Теоретические знания, полученные студентами на лекциях и при самостоятельном изучении курса по литературным источникам, закрепляются при выполнении лабораторных работ.

Целями проведения лабораторных работ являются:

- приобретение практических навыков разработки программ с применением инструментального программного обеспечения;
- контроль самостоятельной работы студентов по освоению курса;
- обучение навыкам профессиональной деятельности.

Цели лабораторных работ достигаются наилучшим образом в том случае, если им предшествует определенная подготовительная внеаудиторная работа. Поэтому преподаватель



Основная профессиональная образовательная программа
09.03.03 Прикладная информатика
(Прикладная информатика в цифровой экономике)

обязан довести до всех студентов график выполнения лабораторных работ с тем, чтобы они могли заниматься целенаправленной самостоятельной работой.

Работы рекомендуется выполнять в той последовательности, в которой они написаны, потому что в некоторых работах используются элементы, полученные в предыдущей работе.

На занятиях со студентами должны широко использоваться разнообразные средства обучения, способствующие более полному и правильному пониманию темы лекции или лабораторного занятия, а также выработке практических навыков по работе с ППО.

К средствам обучения студентов относятся:

- речь преподавателя;
- технические средства обучения: - персональные компьютеры с установленным прикладным программным обеспечением;
- - учебники, учебные пособия, лекции в электронном виде.

7. Характеристика оценочных средств для текущего контроля успеваемости и промежуточной аттестации обучающихся по дисциплине

Для контроля усвоения материала дисциплины «Языки программирования» предусмотрен текущий и промежуточный контроль. Текущий контроль основан на анализе результатов выполнения лабораторных работ и собеседовании по их темам. Промежуточный контроль заключается в сдаче экзамена по дисциплине.

Для проведения экзаменов в письменной или тестовой форме разрабатывается перечень вопросов, утверждаемый заведующим кафедрой. В перечень включаются вопросы из различных разделов курса, позволяющие проверить и оценить теоретические знания студентов и умение применять их для решения практических задач.

Экзамен в письменной форме проводится одновременно для всех студентов академической группы. Время выполнения задания составляет не более одного академического часа. При проведении экзамена в письменной форме оценка выставляется на основе правил, принятых кафедрой, которые должны быть сообщены студентам до начала зачетной (экзаменационной) сессии.

Аналогичные правила могут быть заложены в программы компьютерного тестирования.

При контроле знаний в устной форме преподаватель использует метод индивидуального собеседования, в ходе которого обсуждает со студентом один или несколько вопросов из учебной программы. При необходимости могут быть предложены дополнительные вопросы, задачи и примеры. По окончании ответа на вопросы преподаватель объявляет студенту результаты сдачи экзамена.

8. Учебно-методическое и информационное обеспечение дисциплины

Основная литература:

1. Программирование на языке C++ в среде Qt Creator - Москва: Национальный Открытый Университет «ИНТУИТ», 2016 http://biblioclub.ru/index.php?page=book_red&id=428929
2. Страуструп, Б. Язык программирования C++ для профессионалов / Б. Страуструп. - М. : Интернет-Университет Информационных Технологий, 2006. - 568 с.. То же [Электронный ресурс]. - URL: [http://biblioclub.ru/index.php?page=book_red&id=234816]
3. Корчуганова М. Р., Иванов К. С., Бондарева Л. В. Объектно-ориентированное программирование на C++: электронное учебное пособие - Кемерово: Кемеровский государственный университет, 2015. [http://biblioclub.ru/index.php?page=book_red&id=481559]
4. Очеретовый А.С. Классы объектов-матриц. Методические указания. Иваново: Ивановский государственный университет 2014. 25 с.



Основная профессиональная образовательная программа
09.03.03 Прикладная информатика
(Прикладная информатика в цифровой экономике)

5. Очеретовый А.С. Интегрированная среда программирования Microsoft Visual Studio. Методические указания. Иваново: Ивановский государственный университет 2017. 33 с.

Дополнительная литература:

1. Кирнос В. Н. Информатика II. Основы алгоритмизации и программирования на языке C++: учебно-методическое пособие - Томск: Эль Контент, 2013 [http://biblioclub.ru/index.php?page=book_red&id=208651]
2. Кетков, Ю.Л. Введение в языки программирования С и С++ / Ю.Л. Кетков. - М. : Интернет-Университет Информационных Технологий, 2008. - 252 с. http://biblioclub.ru/index.php?page=book_red&id=234040]
3. Самоучитель по Python. URL: <https://pythonworld.ru/samouchitel-python>
4. Официальный сайт языка Python. URL: <https://www.python.org/doc/>

Ресурсы информационно-телекоммуникационной сети «Интернет»:

Система электронной поддержки образовательного процесса «Мой университет» <https://uni.ivanovo.ac.ru>

Профессиональные базы данных и информационно-справочные системы:

ЭБС «Университетская библиотека онлайн» www.biblioclub.ru;
<http://lib.ivanovo.ac.ru/index.php/polnotekstovye-resursy/ebs-universitetskaya-biblioteka>

Электронная библиотека ИвГУ <http://lib.ivanovo.ac.ru/index.php/polnotekstovye-resursy/elibnew>

Электронный каталог НБ ИвГУ <http://lib.ivanovo.ac.ru/index.php/ek>

Программное обеспечение: операционная система Microsoft Windows, пакет офисных программ Microsoft Office и(или) LibreOffice, интернет-браузер Microsoft Edge и(или) Yandex Browser, среда разработки Visual Studio Code с подключенными и настроенными модулями C++ и Python.

9. Материально-техническое обеспечение дисциплины

Учебные аудитории:

- для проведения занятий лекционного типа с комплектом специализированной учебной мебели и техническими средствами обучения, служащими для предоставления учебной информации большой аудитории;
- для проведения занятий семинарского типа, консультаций, текущего контроля и промежуточной аттестации с комплектом специализированной учебной мебели и техническими средствами обучения;
- для проведения занятий семинарского типа, консультаций, текущего контроля и промежуточной аттестации, выполнения курсовых работ (проектов) с комплектом специализированной учебной мебели и техническими средствами обучения.

Помещение для самостоятельной работы, оснащенное комплектом специализированной учебной мебели, компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в ЭИОС.

Демонстрационное оборудование и учебно-наглядные пособия для занятий лекционного типа, обеспечивающие тематические иллюстрации: макеты, демонстрационные устройства, электронные пособия (презентации, электронные словари и т.п.), визуальные пособия – видеоматериалы, электронные блоки, детали устройств и др., печатные пособия.



Основная профессиональная образовательная программа
09.03.03 Прикладная информатика
(Прикладная информатика в цифровой экономике)

Компьютерный класс, оборудованный проектором и персональными компьютерами, которые объединены ЛВС с выходом в Интернет.



Основная профессиональная образовательная программа
09.03.03 Прикладная информатика
(Прикладная информатика в цифровой экономике)

Автор(ы) рабочей программы дисциплины: ст. преподаватель кафедры ИТиПМ Сидоров М.В.

Программа рассмотрена и утверждена на заседании кафедры Информационных технологий и прикладной математики (ИТиПМ) «30» августа 2024 г., протокол № 1

Программа обновлена
протокол заседания кафедры № _____ от «_____» _____ 20__ г.
Согласовано:
Руководитель ОП _____ Данилова С. В.
(подпись)

Программа обновлена
протокол заседания кафедры № _____ от «_____» _____ 20__ г.
Согласовано:
Руководитель ОП _____ Данилова С. В.
(подпись)

Программа обновлена
протокол заседания кафедры № _____ от «_____» _____ 20__ г.
Согласовано:
Руководитель ОП _____ Данилова С. В.
(подпись)