

Министерство науки и высшего образования Российской Федерации
Муромский институт (филиал)
федерального государственного бюджетного образовательного учреждения высшего образования
**«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»**
(МИ ВлГУ)

Кафедра *ИС*

«УТВЕРЖДАЮ»

Заместитель директора по УР

_____ Д.Е. Андрианов

« 16 » 06 2020 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Сравнительный анализ языков программирования

Направление подготовки

09.03.02 Информационные системы и технологии

Профиль подготовки

Информационные системы и технологии

Квалификация (степень) выпускника

бакалавр

Семестр	Трудоем- кость, час. / зач. ед.	Лек- ции, час.	Прак- тические занятия, час.	Лабора- торные работы, час.	Консультация, час.	Конт- роль, час.	Всего (контакт- ная работа), час.	СРС, час.	Форма промежу- точного контр. (экз., зач., зач. с оц.)
3	108 / 3	16	28		1,6	0,25	45,85	62,15	Зач. с оц.
Итого	108 / 3	16	28		1,6	0,25	45,85	62,15	

Муром, 2020 г.

1. Цель освоения дисциплины

Цель дисциплины: Целью освоения дисциплины (модуля) "Сравнительный анализ языков программирования" является обучение студентов методологии сравнения и оценки различных языков программирования.

Задачи дисциплины:

- знакомство с историей развития языков программирования и основными тенденциями их развития;
- изучение методов сравнения и оценки языков программирования;
- знакомство с синтаксисом и семантикой языков программирования Си, Паскаль, Java и другими;
- получение практических навыков решения задач с применением различных языковых конструкций.

2. Место дисциплины в структуре ОПОП ВО (Цикл (Б1.В.06))

Дисциплина относится к обязательным дисциплинам вариативной части (шифр Б1.В.ОД.5) и обеспечивает понимание особенностей, преимуществ и недостатков различных языков и стилей программирования. Курс базируется на знаниях, полученных студентами в процессе изучения основных дисциплин "Информатика", "Технологии программирования", "Функциональное программирование". Углубление и расширение вопросов, изложенных в данном курсе, будет осуществляться во время работы студентов над дисциплинами: "Управление данными", "Объектно-ориентированное программирование".

3. Компетенции обучающегося, формируемые в результате освоения дисциплины

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

ОПК-7 Способен осуществлять выбор платформ и инструментальных программно-аппаратных средств для реализации информационных систем;

ПК-1 Способность проводить исследования на всех этапах жизненного цикла программных средств.

Результатом освоения дисциплины является достижение следующих индикаторов:

УК-1.1 Знать принципы сбора, отбора и обобщения информации.

ОПК-7.1 Знать основные платформы, технологии и инструментальные программно-аппаратные средства для реализации информационных систем.

ПК-1.1 Знать методологию и основные подходы исследования на всех этапах жизненного цикла программных средств.

ПК-1.2 Уметь применять средства и способы исследования на всех этапах жизненного цикла программных средств.

ОПК-7.2 Уметь осуществлять выбор платформ и инструментальных программно-аппаратных средств для реализации информационных систем, применять современные технологии реализации информационных систем.

УК-1.2 Уметь анализировать и систематизировать разнородные данные, выявлять системные связи и отношения между изучаемыми явлениями и процессами.

ПК-1.3 Владеть методами и средствами исследования на всех этапах жизненного цикла программных средств.

ОПК-7.3 Иметь навыки владения технологиями и инструментальными программно-аппаратными средствами для реализации информационных систем.

УК-1.3 Владеть навыками формулирования и аргументирования выводов и суждений, в том числе с применением философского понятийного аппарата.

4. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 3 зачетные единицы, 108 часов.

4.1. Форма обучения: очная

Уровень базового образования: среднее общее.

Срок обучения 4г.

4.1.1. Структура дисциплины

№ п\п	Раздел (тема) дисциплины	Семестр	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)								Форма текущего контроля успеваемости (по неделям семестра), форма промежуточной аттестации (по семестрам)	
			Лекции	Семинары	Практические занятия	Лабораторные работы	Контрольные работы	СРС	КП / КР	Консультация		Контроль
1	Введение в языки программирования. Процедурные языки программирования	3	2					23				устный опрос
2	Языки, ориентированные на данные. Объектно- ориентированные языки	3	2					7				отчет по ПР
3	Особенности языка C++	3	2					4				отчет по ПР
4	ООП в C++	3	2		4			5				отчет по ПР
5	Непроцедурные языки. Обзор языка Erlang	3	2					3				устный опрос
6	Язык Java	3	2		8			3				отчет по ПР
7	Особенности ООП в Java	3	2		6			2				отчет по ПР
8	Особенности архитектуры Java	3	2		10			15,15				отчет по ПР
Всего за семестр		108	16		28			62,15		1,6	0,25	Зач. с оц.
Итого		108	16		28			62,15		1,6	0,25	

4.1.2. Содержание дисциплины

4.1.2.1. Перечень лекций

Семестр 3

Раздел 1. Введение в языки программирования. Процедурные языки программирования

Лекция 1.

Тема 1. Введение в языки программирования. Процедурные языки программирования (2 часа).

Раздел 2. Языки, ориентированные на данные. Объектно-ориентированные языки

Лекция 2.

Тема 2. Языки, ориентированные на данные. Объектно-ориентированные языки (2 часа).

Раздел 3. Особенности языка C++

Лекция 3.

Тема 3. Особенности языка C++ (2 часа).

Раздел 4. ООП в C++

Лекция 4.

Тема 4. ООП в C++ (2 часа).

Раздел 5. Непроцедурные языки. Обзор языка Erlang

Лекция 5.

Тема 5. Непроцедурные языки. Обзор языка Erlang (2 часа).

Раздел 6. Язык Java

Лекция 6.

Тема 6. Язык Java (2 часа).

Раздел 7. Особенности ООП в Java

Лекция 7.

Тема 7. Особенности ООП в Java (2 часа).

Раздел 8. Особенности архитектуры Java

Лекция 8.

Тема 8. Особенности архитектуры Java (2 часа).

4.1.2.2. Перечень практических занятий

Семестр 3

Раздел 1. ООП в C++

Практическое занятие 1.

Разработка объектно-ориентированной программы на C++ (2 часа).

Практическое занятие 2.

Разработка объектно-ориентированной программы на C++ (2 часа).

Раздел 2. Язык Java

Практическое занятие 3.

Разработка консольных java-приложений (2 часа).

Практическое занятие 4.

Разработка консольных java-приложений (2 часа).

Практическое занятие 5.

Обработка строк. Использование регулярных выражений в java-приложениях (2 часа).

Практическое занятие 6.

Обработка строк. Использование регулярных выражений в java-приложениях (2 часа).

Раздел 3. Особенности ООП в Java

Практическое занятие 7.

Оформление кода на Java. Оформление текстовых документов в соответствии с ГОСТ (2 часа).

Практическое занятие 8.

Оформление кода на Java. Оформление текстовых документов в соответствии с ГОСТ (2 часа).

Раздел 4. Особенности архитектуры Java

Практическое занятие 9.

Использование систем управления версиями (2 часа).

Практическое занятие 10.

Использование систем управления версиями (2 часа).

Практическое занятие 11.

Коллекции в java (2 часа).

Практическое занятие 12.

Коллекции в java (2 часа).

Раздел 5. Особенности ООП в Java

Практическое занятие 13.

Графический интерфейс java-приложений. Swing и AWT (2 часа).

Раздел 6. Особенности архитектуры Java

Практическое занятие 14.

Разработка сетевых приложений на Java (2 часа).

<https://www.mivlgu.ru/iop/mod/resource/view.php?id=15817>

4.1.2.3. Перечень лабораторных работ

Не планируется.

4.1.2.4. Перечень учебно-методического обеспечения для самостоятельной работы

Методические указания для самостоятельной работы размещены на информационно-образовательном портале института по ссылке <https://www.mivlgu.ru/iop/course/view.php?id=5058>.

Для самостоятельной работы также используются издания из списка приведенной ниже основной и дополнительной литературы.

Перечень тем, вынесенных на самостоятельное изучение:

1. Типы перечисления.
2. Массивы и контроль соответствия типов.
3. Операторы goto.
4. Организация подпрограмм в различных языках программирования.
5. Параметры функций и их передача.
6. Стековая архитектура.
7. Распределение памяти, динамическое распределение памяти.
8. Представление вещественных чисел.
9. Полиморфизм языков программирования.
10. Динамическая диспетчеризация.
11. Работа с базами данных в Java.
12. Структура байт-кода Java.
13. Исключения в языке PL/1. Исключения в Ada. Обработка ошибок в языке Eiffel.
14. Параллельное программирование. Что такое параллелизм.
15. Взаимные исключения.
16. Мониторы и защищенные переменные.
17. Передача сообщений.
18. Языки параллельного программирования Occam, Linda.
19. Декомпозиция программ. Раздельная компиляция. Пакеты, модули.
20. Динамический полиморфизм в языке C++.
21. Логическое программирование. Чистое логическое программирование. Унификация.
22. Язык Prolog.

4.1.2.5. Перечень тем контрольных работ, рефератов, ТР, РГР, РПР

Не планируется.

4.1.2.6. Примерный перечень тем курсовых работ (проектов)

Не планируется.

4. 2. Форма обучения: заочная

Уровень базового образования: среднее профессиональное.

Срок обучения 3г 6м.

Семестр	Трудоем- кость, час. / зач. ед.	Лек- ции, час.	Практи- ческие занятия, час.	Лабора- торные работы, час.	Консультация, час.	Конт- роль, час.	Всего (контакт- ная работа), час.	СРС, час.	Форма промежуточного контроля (экз., зач., зач. с оц.)
4	108 / 3	4	6		2	0,5	12,5	91,75	Зач. с оц.(3,75)
Итого	108 / 3	4	6		2	0,5	12,5	91,75	3,75

4.2.1. Структура дисциплины

№ п/п	Раздел (тема) дисциплины	Семестр	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)								Форма текущего контроля успеваемости (по неделям семестра), форма промежу- точной аттестации (по семестрам)	
			Лекции	Семинары	Практические занятия	Лабораторные работы	Контрольные работы	СРС	КП / КР	Консультация		Контроль
1	Введение в языки программирования. Процедурные языки программирования.	4	2		6			48				устный опрос
2	Языки, ориентированные на данные. Объектно-ориентированные языки. Непроцедурные языки.	4	2					43,75				устный опрос
Всего за семестр		108	4		6		+	91,75		2	0,5	Зач. с оц.(3,75)
Итого		108	4		6			91,75		2	0,5	3,75

4.2.2. Содержание дисциплины

4.2.2.1. Перечень лекций

Семестр 4

Раздел 1. Введение в языки программирования. Процедурные языки программирования.

Лекция 1.

Введение в языки программирования. Процедурные языки программирования (2 часа).

Раздел 2. Языки, ориентированные на данные. Объектно-ориентированные языки. Непроцедурные языки.

Лекция 2.

Языки, ориентированные на данные. Объектно-ориентированные языки. Непроцедурные языки (2 часа).

4.2.2.2. Перечень практических занятий

Семестр 4

Раздел 1. Введение в языки программирования. Процедурные языки программирования.

Практическое занятие 1.

Разработка объектно-ориентированной программы на C++ (2 часа).

Практическое занятие 2.

Разработка консольных java-приложений (2 часа).

Практическое занятие 3.

Обработка строк. Использование регулярных выражений в java-приложениях (2 часа).

4.2.2.3. Перечень лабораторных работ

Не планируется.

4.2.2.4. Перечень учебно-методического обеспечения для самостоятельной работы

Методические указания для самостоятельной работы размещены на информационно-образовательном портале института по ссылке <https://www.mivlgu.ru/iop/course/view.php?id=5058>.

Для самостоятельной работы также используются издания из списка приведенной ниже основной и дополнительной литературы.

Перечень тем, вынесенных на самостоятельное изучение:

1. Типы перечисления.
2. Массивы и контроль соответствия типов.
3. Операторы goto.
4. Организация подпрограмм в различных языках программирования.
5. Параметры функций и их передача.
6. Стековая архитектура.
7. Распределение памяти, динамическое распределение памяти.
8. Представление вещественных чисел.
9. Полиморфизм языков программирования.
10. Динамическая диспетчеризация.
11. Работа с базами данных в Java.
12. Структура байт-кода Java.
13. Исключения в языке PL/I. Исключения в Ada. Обработка ошибок в языке Eiffel.
14. Параллельное программирование. Что такое параллелизм.
15. Взаимные исключения.
16. Мониторы и защищенные переменные.
17. Передача сообщений.
18. Языки параллельного программирования Occam, Linda.
19. Декомпозиция программ. Раздельная компиляция. Пакеты, модули.
20. Динамический полиморфизм в языке C++.
21. Логическое программирование. Чистое логическое программирование. Унификация.
22. Язык Prolog.

4.2.2.5. Перечень тем контрольных работ, рефератов, ТР, РГР, РПР

1. Разработать на любых двух языках следующую программу: Умножение матриц.
2. Разработать на любых двух языках следующую программу: Сортировка элементов вектора методом пузырька.
3. Разработать на любых двух языках следующую программу: Сортировка элементов вектора методом выбора.
4. Разработать на любых двух языках следующую программу: Объединение двух матриц в одну матрицу.
5. Разработать на любых двух языках следующую программу: Разбиение матрицы на две матрицы.

6. Разработать на любых двух языках следующую программу: Разбиение вектора на два вектора.
7. Разработать на любых двух языках следующую программу: Объединение двух векторов в вектор.
8. Разработать на любых двух языках следующую программу: Объединение двух векторов в матрицу, состоящую из двух столбцов.
9. Разработать на любых двух языках следующую программу: Объединение двух векторов в матрицу, состоящую из двух строк.
10. Разработать на любых двух языках следующую программу: Разбиение вектора на два вектора.
11. Разработать на любых двух языках следующую программу: Поиск позиции заданного элемента в матрице.
12. Разработать на любых двух языках следующую программу: Поиск минимального элемента и его индексов в матрице.
13. Разработать на любых двух языках следующую программу: Разложение матрицы в вектор.
14. Разработать на любых двух языках следующую программу: Составление матрицы из вектора.
15. Разработать на любых двух языках следующую программу: Удаление заданного элемента в векторе.
16. Разработать на любых двух языках следующую программу: Вставка элемента в заданную позицию вектора.
17. Разработать на любых двух языках следующую программу: Формирования вектора частот появления элементов в матрице.
18. Разработать на любых двух языках следующую программу: Поиск максимального элемента и его индексов в матрице.
19. Разработать на любых двух языках следующую программу: Вставка меньшей матрицы в большую матрицу на заданную позицию.
20. Разработать на любых двух языках следующую программу: Вставка меньшего вектора в больший вектор на заданную позицию.
21. Разработать на любых двух языках следующую программу: Поиск заданной последовательности (подвектора) в векторе.
22. Разработать на любых двух языках следующую программу: Проверка присутствия заданного элемента в векторе.
23. Разработать на любых двух языках следующую программу: Проверка присутствия заданного элемента в матрице.
24. Разработать на любых двух языках следующую программу: Выборка подматрицы из большей матрицы.
25. Разработать на любых двух языках следующую программу: Выборка подвектора из большего вектора.
26. Разработать на любых двух языках следующую программу: Определение наибольшего общего делителя двух чисел.
27. Разработать на любых двух языках следующую программу: Свертка двух векторов разного размера.
28. Разработать на любых двух языках следующую программу: Свертка двух матриц разного размера.
29. Разработать на любых двух языках следующую программу: Поиск матрицы в большей матрице.
30. Разработать на любых двух языках следующую программу: Выборка главной и побочной диагоналей из квадратной матрицы.
31. Разработать на любых двух языках следующую программу: Поворот матрицы на 90 градусов по часовой стрелке.
32. Разработать на любых двух языках следующую программу: Поворот матрицы на 90 градусов против часовой стрелки.

33. Разработать на любых двух языках следующую программу: Вычисление среднего значения элементов матрицы.

34. Разработать на любых двух языках следующую программу: Вычисление количества элементов в матрице, которые больше заданного числа.

4.2.2.6. Примерный перечень тем курсовых работ (проектов)

Не планируется.

5. Образовательные технологии

В процессе изучения дисциплины применяется контактная технология преподавания (за исключением самостоятельно изучаемых студентами вопросов). При проведении лабораторных работ применяется имитационный или симуляционный подход, когда преподавателем разбирается на конкретном примере проблемная ситуация, все шаги решения задачи студентам демонстрируются при помощи мультимедийной техники. Затем студенты самостоятельно решают аналогичные задания.

6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов.

Фонды оценочных средств приведены в приложении.

7. Учебно-методическое и информационное обеспечение дисциплины **Сравнительный анализ языков программирования**

7.1. Основная учебно-методическая литература по дисциплине

1. Кауфман В.Ш. Языки программирования. Концепции и принципы [Электронный ресурс]/ Кауфман В.Ш.— Электрон. текстовые данные.— Саратов: Профобразование, 2019.— 464 с. - <http://www.iprbookshop.ru/88014.html>

2. Липпман С. Язык программирования C++. Полное руководство [Электронный ресурс]/ Липпман С., Лажоие Ж.— Электрон. текстовые данные.— Саратов: Профобразование, 2019.— 1104 с. - <http://www.iprbookshop.ru/89862.html>

3. Сузи Р.А. Язык программирования Python [Электронный ресурс]: учебное пособие/ Сузи Р.А.— Электрон. текстовые данные.— Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020.— 350 с. - <http://www.iprbookshop.ru/97589.html>

7.2. Дополнительная учебно-методическая литература по дисциплине

1. Программирование на C++ / Нейл Дейл, Чип Уимз, Марк Хедингтон — М.: ДМК Пресс, 2013. — 672 с.: ил. - <http://www.iprbookshop.ru/6904>

2. Васюткина И.А. Технология разработки объектно-ориентированных программ на JAVA [Электронный ресурс]: учебно-методическое пособие/ Васюткина И.А.— Электрон. текстовые данные.— Новосибирск: Новосибирский государственный технический университет, 2012.— 152 с. - <http://www.iprbookshop.ru/45047>

3. Серебряков В.А. Теория и реализация языков программирования: учебное пособие — М.: ФИЗМАТЛИТ, 2012. — 236 с. - <http://www.iprbookshop.ru/24388>

4. Кауфман В.Ш. Языки программирования. Концепции и принципы — М.: ДМК Пресс, 2013. - 464 с. - <http://www.iprbookshop.ru/6932>

5. Молдованова О.В. Языки программирования и методы трансляции [Электронный ресурс]: учебное пособие/ Молдованова О.В.— Электрон. текстовые данные.— Новосибирск: Сибирский государственный университет телекоммуникаций и информатики, 2012.— 134 с. - <http://www.iprbookshop.ru/54809>

7.3. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем

В образовательном процессе используются информационные технологии, реализованные на основе информационно-образовательного портала института (www.mivlgu.ru/iop), и инфокоммуникационной сети института:

- предоставление учебно-методических материалов в электронном виде;
- взаимодействие участников образовательного процесса через локальную сеть института и Интернет;
- предоставление сведений о результатах учебной деятельности в электронном личном кабинете обучающегося.

Информационные справочные системы:

электронно-библиотечная система "BOOK.ru" (<https://www.book.ru>);

электронно-библиотечная система "ibooks.ru" (<http://ibooks.ru>);

электронно-библиотечная система "IPRBooks" (<http://www.iprbookshop.ru>).

Программное обеспечение:

Лаборатория разработки информационных систем

Microsoft Visual Studio (подписка DreamSpark Premium Electronic Software; Delivery (3 year) Renewal, договор №453 от 16.12.2014 года)

LibreOffice (freeware, MPL)

TortoiseSVN (freeware, General Public License)

Eclipse for Java EE (freeware, Eclipse Public License)

7.4. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

iprbookshop.ru

8. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Лекционная аудитория

Лаборатория разработки информационных систем

12 персональных компьютеров

проектор Sanyo PDG-DSU20

экран настенный Drapper Apex Star

9. Методические указания по освоению дисциплины

Для успешного освоения теоретического материала обучающийся: знакомится со списком рекомендуемой основной и дополнительной литературы; уточняет у преподавателя, каким дополнительным пособиям следует отдать предпочтение; ведет конспект лекций и прорабатывает лекционный материал, пользуясь как конспектом, так и учебными пособиями.

На практических занятиях пройденный теоретический материал подкрепляется решением задач по основным темам дисциплины. Занятия проводятся в компьютерном классе, используя специальное программное обеспечение. Обучающиеся выполняют индивидуальную задачу в соответствии с заданием на практическую работу. Полученные результаты сводятся в отчет и защищаются по традиционной методике на следующем практическом занятии.

Самостоятельная работа оказывает важное влияние на формирование личности будущего специалиста, она планируется обучающимся самостоятельно. Каждый обучающийся самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием дисциплины. Он выполняет внеаудиторную работу и изучение разделов, выносимых на самостоятельную работу, по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Форма заключительного контроля при промежуточной аттестации – зачет с оценкой. Для проведения промежуточной аттестации по дисциплине разработаны фонд оценочных средств и балльно-рейтинговая система оценки учебной деятельности студентов. Оценка по дисциплине выставляется в информационной системе и носит интегрированный характер, учитывающий результаты оценивания участия студентов в аудиторных занятиях, качества и своевременности выполнения заданий в ходе изучения дисциплины и промежуточной аттестации.

Программа составлена в соответствии с требованиями ФГОС ВО по направлению
09.03.02 Информационные системы и технологии
и профилю подготовки Информационные системы и технологии
Рабочую программу составил к.т.н., доцент каф. ИС, Варламов Алексей Дмитриевич

Рецензент(ы) Директор обособленного подразделения ООО "Ред Софт Центр"
Гуреев А. П.

(Подпись)

Программа рассмотрена и одобрена на заседании кафедры ИС
протокол № _____ от _____ 2020 года.
Заведующий кафедрой ИС _____ Андрианов Д.Е.
(Подпись)

Рабочая программа рассмотрена и одобрена на заседании учебно-методической
комиссии факультета _____

протокол № _____ от _____ 2020 года.
Председатель комиссии _____
(Подпись) (Ф.И.О.)

Программа переутверждена:
на _____ учебный год. Протокол заседания кафедры № _____ от _____ 20__ года.
Заведующий кафедрой _____
(Подпись) (Ф.И.О.)

Программа переутверждена:
на _____ учебный год. Протокол заседания кафедры № _____ от _____ 20__ года.
Заведующий кафедрой _____
(Подпись) (Ф.И.О.)

Программа переутверждена:
на _____ учебный год. Протокол заседания кафедры № _____ от _____ 20__ года.
Заведующий кафедрой _____
(Подпись) (Ф.И.О.)

РЕЦЕНЗИЯ
на рабочую программу дисциплины
«Сравнительный анализ языков программирования»
по направлению подготовки 09.03.02 Информационные системы и технологии

Рабочая программа дисциплины «Сравнительный анализ языков программирования» составлена в соответствии с требованиями федерального государственного образовательного стандарта по направлению подготовки 09.03.02 Информационные системы и технологии.

На изучение данного курса по учебному плану отводится 108 час. (3 ЗЕТ).
Формой итогового контроля изучения дисциплины является зачет с оценкой.

Цель дисциплины: Целью освоения дисциплины (модуля) "Сравнительный анализ языков программирования" является обучение студентов методологии сравнения и оценки различных языков программирования.

Задачи дисциплины:

- знакомство с историей развития языков программирования и основными тенденциями их развития;
- изучение методов сравнения и оценки языков программирования;
- знакомство с синтаксисом и семантикой языков программирования Си, Паскаль, Java и другими;
- получение практических навыков решения задач с применением различных языковых конструкций.

Содержание занятий соответствуют требованиям образовательного стандарта. Имеется перечень вопросов для самостоятельной работы студентов, способствующий более глубокому изучению дисциплины.

Освоение дисциплины позволит студентам приобрести теоретические и практические знания, необходимые при решении задач в будущей практической деятельности.

Предлагаемые фонды оценочных средств для выявления уровня знаний и умений обучаемых полностью охватывает содержание курса и соответствуют ФГОС.

Перечень учебно-методической литературы достаточен для изучения дисциплины. Имеются ссылки на электронно-библиотечные системы.

Рабочая программа дисциплины «Сравнительный анализ языков программирования» рекомендуется для использования в учебном процессе по направлению подготовки 09.03.02 Информационные системы и технологии.

Рецензент:

Директор обособленного
подразделения ООО "Ред Софт
Центр"

Гуреев А. П.

16.06.2020 г.