

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Ярославский государственный университет им. П.Г. Демидова»
Кафедра теоретической информатики

УТВЕРЖДАЮ

Декан факультета ИВТ

 Д.Ю. Чалый

« 24 » мая 20 22 г.

Рабочая программа дисциплины
«Семантика, спецификация и верификация программ»

Научная специальность
1.2.1 Искусственный интеллект и машинное обучение

Форма обучения
очная

Программа рассмотрена
на заседании кафедры
от 15 марта 2022 г.,
протокол № 8

Ярославль

1. Цели освоения дисциплины

Дисциплина «Семантика, спецификация и верификация программ» обеспечивает на примере программ логических контроллеров (ПЛК) приобретение знаний и умений в соответствии с ФГОС ВПО, содействует расширению научного кругозора, формированию представления о современном состоянии теоретической информатики и приобретению специальных знаний из области моделирования, построения и анализа сложных информационных управляющих систем.

Целью изучения дисциплины является: ознакомление со стандартными языками программирования логических контроллеров (ПЛК), современными подходами к спецификации, построению и анализу корректности программ ПЛК, приобретение навыков построения надежных программ логических контроллеров.

2. Место дисциплины в структуре ОП аспирантуры

Дисциплина «Семантика, спецификация и верификация программ» относится к вариативной части (дисциплина по выбору) ОП аспирантуры.

Дисциплина имеет разносторонние связи со многими другими математическими и специальными дисциплинами. При изучении дисциплины «Семантика, спецификация и верификация программ» используются знания основ структурного и объектно-ориентированного программирования, стандартных языков программирования ПЛК, теории алгоритмов (машины Тьюринга, проблемы разрешимости, проблемы сложности алгоритмов) и теории конечных автоматов, формальных языков и грамматик.

При освоении дисциплины необходимы такие личностные характеристики, как: общая образованность, организованность и трудолюбие, самостоятельность, настойчивость в достижении цели.

3. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОП аспирантуры

Процесс изучения дисциплины направлен на формирование следующих элементов компетенций в соответствии с ФГОС ВО, ОП ВО и приобретения следующих знаний, умений, навыков и (или) опыта деятельности:

Знать:

- принципы работы программируемых логических контроллеров;
- устройство и ключевые понятия программируемых логических контроллеров;
- стандартные языки программирования логических контроллеров;
- современные методы спецификации, построения и анализа корректности программ логических контроллеров;
- классические и современные алгоритмы компьютерной математики по части моделирования и анализа программ ПЛК.

Уметь:

- работать со средой разработки программ логических контроллеров (на примере популярной среды CoDeSys);
- разрабатывать программы для программируемых логических контроллеров с применением различных подходов к программированию с использованием стандартных языков ПЛК;
- проводить тестирование программ ПЛК;
- применять инструментальные средства верификации для анализа корректности программ ПЛК.
- проводить имитационное моделирование среды и объектов управления ПЛК;
- проводить спецификацию требований и свойств программ ПЛК на специальных языках.
- строить конечные согласованные модели программ ПЛК;

- разрабатывать и анализировать новые алгоритмы компьютерной математики части моделирования и анализа программ ПЛК.

Владеть:

- Разработки надёжных программ логических контроллеров на стандартных языках ПЛК.
- Проверки корректности программ логических контроллеров;
- приемами и методами математической теории, лежащей в основе направления разработки алгоритмов компьютерной математики по части моделирования и анализа программ ПЛК;
- Базовыми навыками научно-исследовательской работы в области математической теории разработки и анализа алгоритмов по части моделирования и анализа программ ПЛК.

4. Объем, структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 3 зач. ед., 108 акад. час.

№ п/п	Темы (разделы) дисциплины, их содержание	Семестр	Виды учебных занятий, включая самостоятельную работу студентов, и их трудоемкость (в академических часах)						Формы текущего контроля успеваемости Форма промежуточной аттестации (по семестрам)
			Контактная работа						
			лекции	практические	лабораторные	консультации	аттестационные испытания	самостоятельная работа	
1.	Язык спецификации. Темпоральные логики.	2	1					5	Индивидуальное задание. Зачет.
2.	Спецификация поведения программных переменных ПЛК. LTL-спецификация.	2	1					6	Индивидуальное задание. Зачет.
3.	Построение ST-программ ПЛК по LTL-спецификации.	2	1					6	Индивидуальное задание. Зачет.
4.	Построение CFC-программ ПЛК по LTL-спецификации.	2	1					6	Индивидуальное задание. Зачет.
5.	Построение LD-программ ПЛК по LTL-спецификации.	2	1					6	Индивидуальное задание. Зачет.
6.	Построение IL-программ	2	1					6	Индивидуальное

	ПЛК по LTL-спецификации.							задание. Зачет.
7.	Построение SFC-программ ПЛК по LTL-спецификации.	2	1				6	Индивидуальное задание. Зачет.
8.	Имитационное моделирование и тестирование программ ПЛК.	2	1				6	Индивидуальное задание. Зачет.
9.	Построение конечных моделей программ ПЛК по LTL-спецификации	2	1				6	Индивидуальное задание. Зачет.
10.	Дискретное моделирование работы таймеров	2	1				6	Индивидуальное задание. Зачет.
11.	Моделирование согласованного поведения датчиков	2	1				6	Индивидуальное задание. Зачет.
12.	Спецификация свойств моделей программ ПЛК	2	1				6	Индивидуальное задание. Зачет.
13.	Инструментальные средства верификации программ ПЛК. Cadence SMV. SPIN. CPNTools. Применение верификатора SMV для анализа корректности ПЛК-программ	2					6	Индивидуальное задание. Зачет.
14.	Верификация LTL-свойств программ ПЛК	2					6	Индивидуальное задание. Зачет.
15.	Анализ контр-примеров при верификации программ ПЛК	2			1		6	Индивидуальное задание. Зачет.
16.	Теория верификации моделей программ. Model checking	2			1		6	Индивидуальное задание. Зачет.
	Всего		12			2	94	

Содержание разделов дисциплины:

1. Язык спецификации. Темпоральные логики. Темпоральная логика деревьев вычислений STL. Темпоральная логика линейного времени LTL. Основание выбора языка спецификации.
2. Спецификация поведения программных переменных ПЛК. LTL-спецификация.
3. Построение ST-программ ПЛК по LTL-спецификации.
4. Построение SFC-программ ПЛК по LTL-спецификации.
5. Построение LD-программ ПЛК по LTL-спецификации.
6. Построение IL-программ ПЛК по LTL-спецификации.
7. Построение SFC-программ ПЛК по LTL-спецификации.
8. Имитационное моделирование и тестирование программ ПЛК.

9. Построение конечных моделей программ ПЛК по LTL-спецификации.
10. Дискретное моделирование работы таймеров.
11. Моделирование согласованного поведения датчиков.
12. Спецификация свойств моделей программ ПЛК.
13. Инструментальные средства верификации программ ПЛК. Cadence SMV. SPIN. CPNTools. Применение верификатора SMV для анализа корректности ПЛК-программ.
14. Верификация LTL-свойств программ ПЛК.
15. Анализ контр-примеров при верификации программ ПЛК.
16. Теория верификации моделей программ. Model checking.

5. Образовательные технологии, используемые при осуществлении образовательного процесса по дисциплине

Академическая лекция (или лекция общего курса) – последовательное изложение материала, осуществляемое преимущественно в виде монолога преподавателя. Требования к академической лекции: современный научный уровень и насыщенная информативность, убедительная аргументация, доступная и понятная речь, четкая структура и логика, наличие ярких примеров, научных доказательств, обоснований, фактов.

Вводная лекция – дает первое целостное представление о дисциплине (или ее разделе) и ориентирует студента в системе изучения данной дисциплины. Студенты знакомятся с назначением и задачами курса, его ролью и местом в системе учебных дисциплин и в системе подготовки специалиста. Дается краткий обзор курса, история развития науки и практики, достижения в этой сфере, имена известных ученых, излагаются перспективные направления исследований. На этой лекции высказываются методические и организационные особенности работы в рамках курса, а также дается анализ рекомендуемой учебно-методической литературы.

Семинар (семинарское занятие) – форма занятия, на котором происходит обсуждение студентами под руководством преподавателя заранее подготовленных докладов, рефератов, проектов. Семинар выполняет следующие функции: систематизация и обобщение знаний по изученному вопросу, теме, разделу (в том числе в нескольких учебных курсах); совершенствование умений работать с дополнительными источниками, сопоставлять изложение одних и тех же вопросов в различных источниках информации; умений высказывать свою точку зрения, обосновывать ее; писать рефераты, тезисы и планы докладов и сообщений, конспектировать прочитанное. План семинара озвучивается заранее и в нем обычно указываются основные вопросы, подлежащие рассмотрению и литература, рекомендуемая всем и отдельным докладчикам.

Практическое занятие – занятие, посвященное освоению конкретных умений и навыков и закреплению полученных на лекции знаний по предложенному алгоритму.

6. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного программного обеспечения и информационных справочных систем (при необходимости)

В процессе осуществления образовательного процесса используются:

- для формирования текстов материалов для промежуточной и текущей аттестации – программы Microsoft Office, издательская система LaTeX;
- для поиска учебной литературы библиотеки ЯрГУ – Автоматизированная библиотечная информационная система "БУКИ-NEXT" (АБИС "Буки-Next");
- среда программирования CoDeSys 2.3.9. – <http://www.owen.ru/>

– инструментальное средство верификации The Cadence SMV Model Checker.
<http://www.kenmcmil.com/smv.html>

7. Перечень основной и дополнительной учебной литературы, ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

а) основная:

1. Кузьмин Е.В., Рябухин Д.А., Соколов В.А. Моделирование согласованного поведения ПЛК-датчиков // Моделирование и анализ информационных систем, 2014. Т. 21, №4. С. 75-90.
2. Кузьмин Е.В., Соколов В.А., Рябухин Д.А. Построение и верификация LD-программ ПЛК по LTL-спецификации // Моделирование и анализ информационных систем, Ярославль: т. 20 (6), 2013. С. 78-94.
3. Кузьмин Е.В., Соколов В.А., Рябухин Д.А. Построение и верификация ПЛК-программ по LTL-спецификации // Моделирование и анализ информационных систем, Ярославль: т. 20 (4), 2013. С. 5-22.
4. Кузьмин Е.В., Соколов В.А. Моделирование, спецификация и построение программ логических контроллеров // Моделирование и анализ информационных систем, Ярославль: т. 20 (2), 2013. С. 104-120.
5. Кузьмин Е.В., Рябухин Д.А., Соколов В.А. О выразительности подхода к построению ПЛК-программ по LTL-спецификации // Моделирование и анализ информационных систем, Ярославль, ЯрГУ, 2015. Т. 22, №4 С. 507-520.
6. Кузьмин, Е. В., Автоматные счетчиковые машины : монография / Е. В. Кузьмин, В. А. Соколов; Яросл. гос. ун-т, Ярославль, ЯрГУ, 2012, 158с
7. Кузьмин, Е. В., Автоматные счетчиковые машины [Электронный ресурс] : монография / Е. В. Кузьмин, В. А. Соколов; Яросл. гос. ун-т, Ярославль, ЯрГУ, 2012, 158с
8. Кузьмин, Е. В., Верификация моделей программ : учеб. пособие для вузов / Е. В. Кузьмин ; под ред. В. А. Соколова, Ярославль, ЯрГУ, 2008, 174с
9. Кузьмин, Е. В., Верификация моделей программ [Электронный ресурс] : учеб. пособие для вузов / Е. В. Кузьмин ; под ред. В. А. Соколова, Ярославль, ЯрГУ, 2008, 174с

б) дополнительная:

1. Парр Э. Программируемые контроллеры: руководство для инженера. М.: БИНОМ. Лаборатория знаний, 2007. 516 с.
2. Петров И.В. Программируемые контроллеры. Стандартные языки и приемы прикладного проектирования. М.: СОЛОН-Пресс, 2004. 256 с.
3. Минаев И.Г., Самойленко В.В. Программируемые логические контроллеры: практическое руководство для начинающего инженера. Ставрополь: АГРУС, 2009. 100 с.
4. Руководство пользователя по программированию ПЛК в CoDeSys 2.3. ПК Пролог, 2008.
5. Визуализация CoDeSys. Дополнение к руководству пользователя по программированию ПЛК в CoDeSys 2.3. ПК Пролог, 2008.

в) ресурсы сети «Интернет»

1. Среда программирования CoDeSys 2.3.9. – <http://www.owen.ru>,
<http://www.3s-software.com>
2. SMV. The Cadence SMV Model Checker. <http://www.kenmcmil.com/smv.html>
3. SPIN. <http://spinroot.com/spin/whatispin.html>
4. UPPAAL. <http://www.uppaal.com>

5. CPNTools. <http://cpntools.org/>
6. <http://prolog-plc.ru/publications>

8. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине

Аудитории, оборудованные для проведения лекций, практических занятий и консультаций, фонд библиотеки, компьютерная техника.

Автор(ы) : профессор кафедры теоретической информатики, д. ф.-м. н.
Кузьмин Е.В.

**Приложение №1 к рабочей программе дисциплины
«Семантика, спецификация и верификация программ»**

Фонд оценочных средств

для проведения текущей и промежуточной аттестации студентов по дисциплине

1. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

1.1. Контрольные задания и иные материалы, используемые в процессе текущей аттестации

Вариант индивидуального задания для проверки сформированности компетенций ПК-2 и ПК-3.

Установка, схема которой представлена на рисунке, предназначена для приготовления смесей из двух компонентов. Первый смешиваемый компонент подается из бака 1 посредством клапана «Кл1» в изначально пустой (нет сигнала от датчика «ДУ0») резервуар до тех пор, пока не сработает датчик уровня «ДУ1». По срабатыванию датчика «ДУ1» клапан «Кл1» закрывается. Затем открывается «Кл2» и из бака 2 подается второй смешиваемый компонент до тех пор, пока не сработает датчик уровня «ДУ2», после чего клапан «Кл2» закрывается. Далее в течение T секунд с помощью мешалки в резервуаре происходит перемешивание компонентов при определенной температуре, которая поддерживается нагревателем. По истечении этого времени открывается клапан «РКл» и происходит выгрузка готовой смеси. Выгрузка заканчивается, т.е. клапан «РКл» закрывается, как только пропадает сигнал от датчика уровня «ДУ0». Клапан «АКл» используется для принудительного слива смеси из резервуара. Наличие смешиваемых компонентов в баках определяется по соответствующим датчикам уровня «ДБ1» и «ДБ2». Функционирование мешалки определяется по датчику работы привода мешалки «ДРП». Температура нагревательного элемента отслеживается с помощью датчика верхней температуры «ВТ», датчика нижней температуры «НТ» и датчика рабочей температуры «РТ». Если после запуска установки нагреватель имеет температуру меньше нижней температуры, то он включается и начинает нагреваться. Как только температура нагревателя достигает заданного верхнего значения, то нагреватель выключается и постепенно остывает. Задача нагревателя состоит в поддержании некоторой рабочей температуры, необходимой для смешивания компонентов. Рабочая температура устанавливается таким образом, чтобы быть на несколько градусов меньше нижней температуры. Запуск, экстренная остановка и плавное завершение работы установки осуществляется соответствующими кнопками панели управления «Пуск», «Стоп» и «Заверш». Под плавным завершением работы понимается выключение установки после того, как будет приготовлена текущая порция смеси и полностью выдана через клапан «РКл». Установка по приготовлению смесей является автоматической. Управление установкой осуществляется с помощью программируемого логического контроллера (ПЛК), получающего входные сигналы от датчиков установки, кнопок и переключателей панели управления оператора и подающего выходные сигналы на нагреватель, привод мешалки, приводы клапанов и лампы панели управления (см. рис.).

Задача состоит в написании четырех программ для ПЛК с 13 входами и 23 выходами, предназначенного для управления установкой. ПЛК-программы должны быть полностью написаны на стандартных языках FBD/CFC, LD, ST и SFC соответственно (один язык на одну программу). Затем программы должны быть проверены на корректность с помощью инструментального средства верификации SMV. Проверяемые свойства приводятся отдельным списком. Интерфейс ПЛК управления установкой приводится на рисунке. Подробные требования к программе ПЛК и визуализации объектов управления выдаются отдельно.

Критерии оценивания

Оценка	Критерии
Отлично Высокий уровень формирования компетенции	Правильно построен алгоритм решения задачи. Программа реализована и выполняется в соответствии с поставленной задачей и правильно выполняется для всех тестов. Разработаны и выполняются полные наборы тестов для проверки работы программы. Студент легко ориентируется в коде работы, вносит поправки, применяет и объясняет изучаемые элементы языка программирования. Используются эффективные методы решения
Хорошо Провдвинутый уровень формирования компетенции	Правильно построен алгоритм решения задачи. Программа реализована и выполняется в соответствии с поставленной задачей. Программа правильно выполняется для большинства тестов, но необходимы доработки программного кода. Разработаны и выполняются наборы тестов для проверки работы программы. Студент ориентируется в коде работы, вносит поправки, возможно не для всех заданных вопросов. Студент применяет и объясняет основные изучаемые элементы языка программирования (технологии программирования), возможно неполное понимание всех нюансов применяемых средств программирования. Используются эффективные методы решения.
Удовлетворительно Пороговый уровень формирования компетенции	Алгоритм решения задачи построен правильно за исключением отдельных небольших частей решения. Архитектурное решение задачи в основном правильное. Программа реализована и выполняется, но не для всех тестов, доработка программного кода не всегда приводит к желаемому результату. Разработаны и выполняются не полные наборы тестов для проверки работы программы. Студент ориентируется в коде работы, но затрудняется внести поправки в случае достаточно сложных вопросов. Студент применяет, но не объясняет основные изучаемые элементы языка программирования (технологии программирования). Используются неэффективные методы решения
Неудовлетворительно Компетенции не сформированы	Задача решена неверно или решена совсем другая задача. Программа не выполняется или выполняется неверно для большинства тестов. Отсутствует понимание кода программы и применяемых средств программирования. Код является дубликатом уже существующего решения. Студент не может написать код для решения простых задач.

1.2. Список вопросов и (или) заданий для проведения промежуточной аттестации

Вопросы для подготовки к зачету.

1. Язык спецификации. Темпоральные логики.
2. Спецификация поведения программных переменных ПЛК. LTL-спецификация.
3. Построение ST-программ ПЛК по LTL-спецификации.
4. Построение SFC-программ ПЛК по LTL-спецификации.
5. Построение LD-программ ПЛК по LTL-спецификации.
6. Построение IL-программ ПЛК по LTL-спецификации.
7. Построение SFC-программ ПЛК по LTL-спецификации.
8. Имитационное моделирование и тестирование программ ПЛК.
9. Построение конечных моделей программ ПЛК по LTL-спецификации.
10. Дискретное моделирование работы таймеров.
11. Моделирование согласованного поведения датчиков.
12. Спецификация свойств моделей программ ПЛК.
13. Инструментальные средства верификации программ ПЛК. Cadence SMV. SPIN. CPNTools. Применение верификатора SMV для анализа корректности ПЛК-программ.
14. Верификация LTL-свойств программ ПЛК.
15. Анализ контр-примеров при верификации программ ПЛК.
16. Теория верификации моделей программ. Model checking.

2. Перечень компетенций, этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкалы оценивания

2.1. Шкала оценивания сформированности компетенций и ее описание

Оценивание уровня сформированности компетенций в процессе освоения дисциплины осуществляется по следующей трехуровневой шкале:

Пороговый уровень - предполагает отражение тех ожидаемых результатов, которые определяют минимальный набор знаний и (или) умений и (или) навыков, полученных студентом в результате освоения дисциплины. Пороговый уровень является обязательным уровнем для студента к моменту завершения им освоения данной дисциплины.

Продвинутый уровень - предполагает способность студента использовать знания, умения, навыки и (или) опыт деятельности, полученные при освоении дисциплины, для решения профессиональных задач. Продвинутый уровень превосходит пороговый уровень по нескольким существенным признакам.

Высокий уровень - предполагает способность студента использовать потенциал интегрированных знаний, умений, навыков и (или) опыта деятельности, полученных при освоении дисциплины, для творческого решения профессиональных задач и самостоятельного поиска новых подходов в их решении путем комбинирования и использования известных способов решения применительно к конкретным условиям. Высокий уровень превосходит пороговый уровень по всем существенным признакам.

2.2. Перечень компетенций, этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования

Форма контроля	Этапы формирования (№ темы (раздела))	Показатели оценивания	Шкала и критерии оценивания компетенций на различных этапах их формирования		
			Пороговый уровень	Продвинутый уровень	Высокий уровень
Индивидуальное задание. Зачет.	1-16	<p>Знать:</p> <ul style="list-style-type: none"> – принципы работы программируемых логических контроллеров; – устройство и ключевые понятия программируемых логических контроллеров; – стандартные языки программирования логических контроллеров; – современные методы спецификации, построения и анализа корректности программ логических контроллеров. 	<p>Частичное выполнение задачи. Частичное освоение материалов курса.</p> <p>Знать:</p> <ul style="list-style-type: none"> – принципы работы программируемых логических контроллеров; – устройство и ключевые понятия программируемых логических контроллеров; – стандартные языки программирования логических контроллеров; – современные методы спецификации, построения и анализа корректности 	<p>Неполное (но в целом верное) выполнение задачи. Неполное (но в значительном объеме) освоение материалов курса.</p> <p>Знать:</p> <ul style="list-style-type: none"> – принципы работы программируемых логических контроллеров; – устройство и ключевые понятия программируемых логических контроллеров; – стандартные языки программирования логических контроллеров; – современные методы спецификации, построения и анализа корректности 	<p>Выполнение задачи в полном объеме. Освоение материалов курса в полном объеме.</p> <p>Знать:</p> <ul style="list-style-type: none"> – принципы работы программируемых логических контроллеров; – устройство и ключевые понятия программируемых логических контроллеров; – стандартные языки программирования логических контроллеров; – современные методы спецификации, построения и анализа корректности программ логических контроллеров. <p>Уметь:</p> <ul style="list-style-type: none"> – работать со средой разработки программ логических контроллеров (на примере популярной среды CoDeSys);

		<p>Уметь:</p> <ul style="list-style-type: none"> – работать со средой разработки программ логических контроллеров (на примере популярной среды CoDeSys); – разрабатывать программы для программируемых логических контроллеров с применением различных подходов к программированию с использованием стандартных языков ПЛК; – проводить тестирование программ ПЛК; – применять инструментальные средства верификации для анализа корректности программ ПЛК. 	<p>программ логических контроллеров.</p> <p>Уметь:</p> <ul style="list-style-type: none"> – работать со средой разработки программ логических контроллеров (на примере популярной среды CoDeSys); – разрабатывать программы для программируемых логических контроллеров с применением различных подходов к программированию с использованием стандартных языков ПЛК; – проводить тестирование программ ПЛК; – применять инструментальные средства верификации для 	<p>программ логических контроллеров.</p> <p>Уметь:</p> <ul style="list-style-type: none"> – работать со средой разработки программ логических контроллеров (на примере популярной среды CoDeSys); – разрабатывать программы для программируемых логических контроллеров с применением различных подходов к программированию с использованием стандартных языков ПЛК; – проводить тестирование программ ПЛК; – применять инструментальные средства верификации для анализа корректности программ ПЛК. – проводить имитационное моделирование 	<ul style="list-style-type: none"> – разрабатывать программы для программируемых логических контроллеров с применением различных подходов к программированию с использованием стандартных языков ПЛК; – проводить тестирование программ ПЛК; – применять инструментальные средства верификации для анализа корректности программ ПЛК. – проводить имитационное моделирование среды и объектов управления ПЛК; – проводить спецификацию требований и свойств программ ПЛК на специальных языках. <p>Владеть навыками:</p> <ul style="list-style-type: none"> – Разработки надёжных программ логических контроллеров на стандартных языках ПЛК. – Проверки корректности программ логических
--	--	--	--	---	---

		<ul style="list-style-type: none"> – проводить имитационное моделирование среды и объектов управления ПЛК; – проводить спецификацию требований и свойств программ ПЛК на специальных языках. – строить конечные согласованные модели программ ПЛК. <p>Владеть навыками:</p> <ul style="list-style-type: none"> – Разработки надёжных программ логических контроллеров на стандартных языках ПЛК. – Проверки корректности программ логических контроллеров 	<p>анализа корректности программ ПЛК.</p> <ul style="list-style-type: none"> – проводить имитационное моделирование среды и объектов управления ПЛК; – проводить спецификацию требований и свойств программ ПЛК на специальных языках. – строить конечные согласованные модели программ ПЛК. <p>Владеть навыками:</p> <ul style="list-style-type: none"> – Разработки надёжных программ логических контроллеров на стандартных языках ПЛК. – Проверки корректности программ логических контроллеров 	<p>среды и объектов управления ПЛК;</p> <ul style="list-style-type: none"> – проводить спецификацию требований и свойств программ ПЛК на специальных языках. – строить конечные согласованные модели программ ПЛК. <p>Владеть навыками:</p> <ul style="list-style-type: none"> – Разработки надёжных программ логических контроллеров на стандартных языках ПЛК. – Проверки корректности программ логических контроллеров 	<p>контроллеров</p>
--	--	---	---	--	---------------------

<p>Индивидуальное задание. Зачет.</p>	<p>1-16</p>	<p>Знать: - классические и современные алгоритмы компьютерной математики по части моделирования и анализа программ ПЛК. Уметь: - разрабатывать и анализировать новые алгоритмы компьютерной математики части моделирования и анализа программ ПЛК. Владеть: - приемами и методами математической теории, лежащей в основе направления разработки алгоритмов компьютерной математики по части моделирования и анализа программ ПЛК. - базовыми навыками научно-исследовательской работы в области математической теории разработки и анализа алгоритмов по части моделирования и анализа программ ПЛК.</p>	<p>Частичное выполнение задачи. Частичное освоение материалов курса. Знать: - классические и современные алгоритмы компьютерной математики по части моделирования и анализа программ ПЛК. Уметь: - разрабатывать и анализировать новые алгоритмы компьютерной математики части моделирования и анализа программ ПЛК. Владеть: - приемами и методами математической теории, лежащей в основе направления разработки алгоритмов компьютерной математики по части моделирования и анализа программ ПЛК. - базовыми навыками научно-исследовательской работы в области математической теории разработки и анализа</p>	<p>Неполное (но в целом верное) выполнение задачи. Неполное (но в значительном объеме) освоение материалов курса. Знать: - классические и современные алгоритмы компьютерной математики по части моделирования и анализа программ ПЛК. Уметь: - разрабатывать и анализировать новые алгоритмы компьютерной математики части моделирования и анализа программ ПЛК. Владеть: - приемами и методами математической теории, лежащей в основе направления разработки алгоритмов компьютерной математики по части моделирования и анализа программ ПЛК. - базовыми навыками научно-исследовательской работы в области математической теории разработки и анализа алгоритмов по части моделирования и анализа программ ПЛК.</p>	<p>Выполнение задачи в полном объеме. Освоение материалов курса в полном объеме. Знать: - классические и современные алгоритмы компьютерной математики по части моделирования и анализа программ ПЛК. Уметь: - разрабатывать и анализировать новые алгоритмы компьютерной математики части моделирования и анализа программ ПЛК. Владеть: - приемами и методами математической теории, лежащей в основе направления разработки алгоритмов компьютерной математики по части моделирования и анализа программ ПЛК. - базовыми навыками научно-исследовательской работы в области математической теории разработки и анализа алгоритмов по части моделирования и анализа программ ПЛК.</p>
---	-------------	--	--	---	--

			алгоритмов по части моделирования и анализа программ ПЛК.		
--	--	--	---	--	--

3. Методические рекомендации преподавателю по процедуре оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Целью процедуры оценивания является определение степени овладения студентом ожидаемыми результатами обучения (знаниями, умениями, навыками и (или) опытом деятельности).

Процедура оценивания степени овладения студентом ожидаемыми результатами обучения осуществляется с помощью методических материалов, представленных в разделе «Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций»

3.1 Критерии оценивания степени овладения знаниями, умениями, навыками и (или) опытом деятельности, определяющие уровни сформированности компетенций

Пороговый уровень (общие характеристики):

- владение основным объемом знаний по программе дисциплины;
- знание основной терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы без существенных ошибок;
- владение инструментарием дисциплины, умение его использовать в решении стандартных (типовых) задач;
- способность самостоятельно применять типовые решения в рамках рабочей программы дисциплины;
- усвоение основной литературы, рекомендованной рабочей программой дисциплины;
- знание базовых теорий, концепций и направлений по изучаемой дисциплине;
- самостоятельная работа на практических и лабораторных занятиях, периодическое участие в групповых обсуждениях, достаточный уровень культуры исполнения заданий.

Продвинутый уровень (общие характеристики):

- достаточно полные и систематизированные знания в объёме программы дисциплины;
- использование основной терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы, умение делать выводы;
- владение инструментарием дисциплины, умение его использовать в решении учебных и профессиональных задач;
- способность самостоятельно решать сложные задачи (проблемы) в рамках рабочей программы дисциплины;
- усвоение основной и дополнительной литературы, рекомендованной рабочей программой дисциплины;
- умение ориентироваться в базовых теориях, концепциях и направлениях по изучаемой дисциплине и давать им сравнительную оценку;
- самостоятельная работа на практических и лабораторных занятиях, участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.

Высокий уровень (общие характеристики):

- систематизированные, глубокие и полные знания по всем разделам дисциплины;
- точное использование терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы, умение делать обоснованные выводы;

- безупречное владение инструментарием дисциплины, умение его использовать в постановке и решении научных и профессиональных задач;
- способность самостоятельно и творчески решать сложные задачи (проблемы) в рамках рабочей программы дисциплины;
- полное и глубокое усвоение основной и дополнительной литературы, рекомендованной рабочей программой дисциплины;
- умение ориентироваться в основных теориях, концепциях и направлениях по изучаемой дисциплине и давать им критическую оценку;
- активная самостоятельная работа на практических и лабораторных занятиях, творческое участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.

3.2 Описание процедуры выставления оценки

В зависимости от уровня сформированности каждой компетенции по окончании освоения дисциплины студенту выставляется оценка. Для дисциплин, изучаемых в течение нескольких семестров, оценка может выставляться не только по окончании ее освоения, но и в промежуточных семестрах. Вид оценки («отлично», «хорошо», «удовлетворительно», «неудовлетворительно», «зачтено», «незачтено») определяется рабочей программой дисциплины в соответствии с учебным планом.

Оценка «отлично» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована на высоком уровне.

Оценка «хорошо» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована не ниже, чем на продвинутом уровне.

Оценка «удовлетворительно» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована не ниже, чем на пороговом уровне.

Оценка «неудовлетворительно» выставляется студенту, у которого хотя бы одна компетенция (полностью или частично формируемая данной дисциплиной) сформирована ниже, чем на пороговом уровне.

Оценка «зачет» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована не ниже, чем на пороговом уровне.

Оценка «незачтено» выставляется студенту, у которого хотя бы одна компетенция (полностью или частично формируемая данной дисциплиной) сформирована ниже, чем на пороговом уровне.

Приложение №2 к рабочей программе дисциплины «Семантика, спецификация и верификация программ»

Методические указания для студентов по освоению дисциплины

Программируемый логический контроллер (ПЛК) — это программно управляемый дискретный автомат, имеющий некоторое множество входов, подключенных посредством датчиков к объекту управления, и множество выходов, подключенных к исполнительным устройствам. ПЛК контролирует состояния входов и вырабатывает определенные последовательности программно заданных действий, отражающихся в изменении выходов.

ПЛК предназначен для работы в режиме реального времени в условиях промышленной среды. ПЛК имеют широкое распространение. Они задействованы начиная с бытовых приборов и заканчивая сложными системами (процессами) управления промышленными объектами. Любое устройство, способное автоматически выполнять некоторые операции, имеет в своем составе управляющий контроллер — модуль, обеспечивающий логику работы устройства.

Задачей прикладного программирования ПЛК является реализация алгоритма управления конкретным устройством. При этом к управляющим программам ПЛК, реализующим логику процессов, предъявляются строгие требования корректной работы, поскольку программная ошибка может привести к серьезным негативным последствиям (причинение вреда здоровью персонала из-за нарушения технологического процесса, экономический ущерб по причине аварийного простоя или перерасхода сырья и т. п.).

Изначально ПЛК предназначались для управления последовательными логическими процессами, что и обусловило слово «логический» в названии ПЛК. Современные ПЛК помимо простых логических операций способны выполнять цифровую обработку сигналов, управление приводами, регулирование, функции операторского управления и т. д.

Физически, типичный ПЛК представляет собой блок, имеющий определенный набор выходов и входов, для подключения датчиков и исполнительных механизмов. Логика управления описывается программно на основе микрокомпьютерного ядра.

Языки программирования ПЛК определяются международным стандартом МЭК 61131-3. Этот стандарт включает в себя описание пяти языков программирования ПЛК: SFC, IL, ST, LD и FBD.

Язык IL (Instruction List) дословно — список инструкций. Это типичный ассемблер с аккумулятором и переходами по меткам.

Язык ST (Structured Text) — язык высокого уровня, синтаксически представляющий собой несколько адаптированный язык Паскаль.

Язык релейных диаграмм LD (Ladder Diagram) или релейно-контактных схем (РКС) — графический язык, реализующий структуры электрических цепей.

FBD (Function Block Diagram) — графический язык программирования, диаграммы в рамках которого очень напоминают принципиальную схему электронного устройства на микросхемах.

SFC (Sequential Function Chart) — последовательные функциональные схемы. Диаграммы SFC являются высокоуровневым графическим инструментом. Графическая диаграмма SFC состоит из шагов и переходов между ними. Разрешение перехода определяется условием. С шагом связаны определенные действия. Описания действий выполняются на любом языке МЭК. Сам SFC не содержит каких-либо управляющих команд ПЛК. Действия могут быть описаны и в виде вложенной SFC-схемы. Можно создать несколько уровней подобных вложений, но в конечном счете действия нижнего уровня все равно необходимо будет описать на IL, ST, LD или FBD. Целью применения SFC является разделение задачи на простые этапы с формально определенной логикой

работы системы. SFC дает возможность быстрого построения прототипа системы без программирования и привязки к конкретным аппаратным средствам.

Указанные языки представляют собой простой, но достаточно мощный инструмент для реализации задач ПЛК. При программировании одной из самых удобных и удачных комбинаций языков является сочетание SFC-ST-LD, при котором управляющий каркас программы строится в виде схем SFC, условия переходов по шагам SFC-схемы описываются на языке ST, а сами SFC-шаги наполняются LD-диаграммами.

В современных ПЛК наряду с дискретными широко используются аналоговые входы и выходы. В сфере применения ПЛК дискретными называют бинарные входы и выходы, т. е. входы и выходы, значения сигналов на которых способны принимать только два состояния — логического нуля и логической единицы.

Датчиками, формирующими входной дискретный сигнал, являются кнопки ручного управления, концевые датчики, датчики движения, датчики уровня, контактные термометры и т. п. Сфера применения бинарных выходов также довольно очевидна: электромагнитные реле, силовые пускатели, электромагнитные клапаны, световые сигнализаторы и т. д.

Аналоговый (непрерывный) сигнал отражает уровень напряжения или тока, соответствующий некоторой физической величине в каждый момент времени. Этот уровень может относиться к любой физической величине — температуре, давлению, весу, положению, скорости, частоте и т. д. Несмотря на название, аналоговые сигналы в ПЛК обязательно преобразуются в цифровую, т. е. заведомо дискретную форму представления. Но в технических документах ПЛК встречается именно указание количества дискретных и аналоговых входов и выходов.

В рамках дисциплины «Спецификация и верификация ПЛК» с целью сделать акцент именно на построении логики управления все задачи (объекты управления и рабочие процессы) подобраны таким образом, чтобы их реализация могла быть осуществлена на «дискретных» ПЛК, т. е. ПЛК, имеющих только дискретные входы и выходы.

Важно отметить, что задачи в рамках индивидуальных заданий являются учебными и не отражают в полной мере всего того, с чем разработчик системы управления может встретиться на практике. Более того, была сделана попытка выделить исключительно область компетенции программиста ПЛК. Рассмотренные задачи не касаются вопросов грамотного проектирования безопасных системы автоматического управления (на основе ПЛК), что является ответственной сферой деятельности подготовленных специалистов — квалифицированных инженеров-проектировщиков. Каждая задача помимо словесного описания снабжена принципиальной схемой объекта управления и интерфейсом управляющего ПЛК. Таким образом, задача является строго задачей программиста, выполняется в четко очерченных рамках технического задания и состоит в реализации управляющей логики ПЛК на языках стандарта МЭК 61131-3. Предполагается, что разработка управляющей программы ПЛК может осуществляться в режиме эмуляции без подключения внешнего оборудования. При этом одно из основных требований, предъявляемых к программе, — это простота ее восприятия и понимания всеми специалистами, занятыми при построении системы автоматического управления на базе ПЛК.

Подробные сведения о строении программируемых логических контроллеров и способах их программирования могут быть найдены в источниках из списка литературы.

Современные ПЛК универсального назначения, поддерживающие стандарт МЭК 61131-3, программируются на персональных компьютерах с помощью специализированных комплексов, среди которых одним из наиболее популярным является CoDeSys, разработанный компанией 3S (Smart Software Solutions).

CoDeSys (Controllers Development System) — это современный инструмент для программирования контроллеров. CoDeSys предоставляет программисту удобную среду

для программирования контроллеров на языках стандарта МЭК 61131-3. Используемые редакторы и отладочные средства базируются на широко известных и хорошо себя зарекомендовавших принципах, знакомых по другим популярным средам профессионального программирования (такие, как Visual C++). Производители ПЛК обычно поставляют этот комплекс в качестве бесплатного приложения или размещают его на своих сайтах.

Предполагается, что задачи индивидуальных заданий будут выполняться в среде CoDeSys не ниже русифицированной версии 2.3.9. При этом необходимо активно использовать встроенные средства визуализации объектов и процессов управления. Информация по работе с комплексом CoDeSys V.2.3 и встроенными средствами визуализации может быть найдена по ссылкам в списке литературы.

Предполагается, что для выполнения индивидуального задания будет применяться подход (технология) к построению и верификации программ логических контроллеров (ПЛК) для «дискретных» задач логического управления. В рамках этого подхода спецификация программного поведения проводится на языке темпоральной логики линейного времени LTL. Программирование осуществляется на языках ST, LD, IL, FBD/CFC и SFC по LTL-спецификации специального вида. Анализ корректности LTL-спецификации производится с помощью программного средства символьной проверки модели Cadence SMV. Этот подход к программированию ПЛК обеспечивает возможность анализа корректности ПЛК-программ с помощью метода проверки модели (model checking). Изменение значения каждой программной переменной предлагается описывается с помощью пары LTL-формул. Первая LTL-формула описывает ситуации, при которых происходит возрастание значения соответствующей переменной, вторая LTL-формула задает условия, приводящие к уменьшению значения переменной. Рассматриваемые для спецификации поведения переменных LTL-формулы являются конструктивными в том смысле, что по ним производится построение ПЛК-программы, которая соответствует темпоральным свойствам, выраженным этими формулами. Программирование ПЛК сводится к построению LTL-спецификации поведения каждой программной переменной. По LTL-спецификации строится SMV-модель, которая затем проверяется на корректность методом проверки модели с помощью средства верификации Cadence SMV.

При построении модели программы важно обеспечить согласованное поведение ПЛК-датчиков. Предлагается описывать поведение датчиков с помощью трех групп LTL-формул, которые при проверке справедливости программных свойств будут оказывать влияние на программную модель, приближая ее к реальному поведению исходной ПЛК-программы.

Учебно-методическое обеспечение самостоятельной работы студентов по дисциплине

Для самостоятельной работы особенно рекомендуется использовать учебную литературу.

Также для подбора учебной литературы рекомендуется использовать широкий спектр интернет-ресурсов:

1. Электронно-библиотечная система «Университетская библиотека online» (www.biblioclub.ru) - электронная библиотека, обеспечивающая доступ к наиболее востребованным материалам-первоисточникам, учебной, научной и художественной литературе ведущих издательств (*регистрация в электронной библиотеке – только в сети университета. После регистрации работа с системой возможна с любой точки доступа в Internet.).

2. Информационная система "Единое окно доступа к образовательным ресурсам" (<http://window.edu.ru/library>).

Целью создания информационной системы "Единое окно доступа к образовательным ресурсам" (ИС "Единое окно ") является обеспечение свободного

доступа к интегральному каталогу образовательных интернет-ресурсов и к электронной библиотеке учебно-методических материалов для общего и профессионального образования.

Информационная система "Единое окно доступа к образовательным ресурсам" создана по заказу Федерального агентства по образованию в 2005-2008 гг. Главной разработчик проекта - Федеральное государственное автономное учреждение Государственный научно-исследовательский институт информационных технологий и телекоммуникаций (ФГАУ ГНИИ ИТТ "Информика") www.informika.ru.

ИС "Единое окно" объединяет в единое информационное пространство электронные ресурсы свободного доступа для всех уровней образования в России. Разделы этой системы:

- Электронная библиотека – является крупнейшим в российском сегменте Интернета хранилищем полнотекстовых версий учебных, учебно-методических и научных материалов с открытым доступом. Библиотека содержит более 30 000 материалов, источниками которых являются более трехсот российских вузов и других образовательных и научных учреждений. Основу наполнения библиотеки составляют электронные версии учебно-методических материалов, подготовленные в вузах, прошедшие рецензирование и рекомендованные к использованию советами факультетов, учебно-методическими комиссиями и другими вузовскими структурами, осуществляющими контроль учебно-методической деятельности.

- Интегральный каталог образовательных интернет-ресурсов содержит представленные в стандартизированной форме метаданные внешних ресурсов, а также содержит описания полнотекстовых публикаций электронной библиотеки. Общий объем каталога превышает 56 000 метаописаний (из них около 25 000 - внешние ресурсы). Расширенный поиск в "Каталоге" осуществляется по названию, автору, аннотации, ключевым словам с возможной фильтрацией по тематике, предмету, типу материала, уровню образования и аудитории.

- Избранное. В разделе представлены подборки наиболее содержательных и полезных, по мнению редакции, интернет-ресурсов для общего и профессионального образования.

- Библиотеки вузов. Раздел содержит подборки сайтов вузовских библиотек, электронных каталогов библиотек вузов и полнотекстовых электронных библиотек вузов.

Для самостоятельного подбора литературы в библиотеке ЯрГУ рекомендуется использовать:

1. Личный кабинет (http://lib.uniyar.ac.ru/opac/bk_login.php) дает возможность получения on-line доступа к списку выданной в автоматизированном режиме литературы, просмотра и копирования электронных версий изданий сотрудников университета (учеб. и метод. пособия, тексты лекций и т.д.) Для работы в «Личном кабинете» необходимо зайти на сайт Научной библиотеки ЯрГУ с любой точки, имеющей доступ в Internet, в пункт меню «Электронный каталог»; пройти процедуру авторизации, выбрав вкладку «Авторизация», и заполнить представленные поля информации.

2. Электронная библиотека учебных материалов ЯрГУ (http://www.lib.uniyar.ac.ru/opac/bk_cat_find.php) содержит более 2500 полных текстов учебных и учебно-методических материалов по основным изучаемым дисциплинам, изданных в университете. Доступ в сети университета, либо по логину/пароллю.

3. Электронная картотека «Книгообеспеченность» (http://www.lib.uniyar.ac.ru/opac/bk_bookreq_find.php) раскрывает учебный фонд научной библиотеки ЯрГУ, предоставляет оперативную информацию о состоянии книгообеспеченности дисциплин основной и дополнительной литературой, а также цикла дисциплин и специальностей. Электронная картотека «Книгообеспеченность» доступна в сети университета и через Личный кабинет.