

МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
Ярославский государственный университет им. П.Г.Демидова

УЧЕБНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ
ПО ДИСЦИПЛИНЕ

ВЕРИФИКАЦИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Направление подготовки (специальность):

02.04.02 Фундаментальная информатика и информационные технологии

Образовательная программа

Искусственный интеллект и компьютерные науки

очная форма обучения

Составитель:

КУЗЬМИН ЕГОР ВЛАДИМИРОВИЧ,
Д.Ф.-М.Н., ПРОФЕССОР Ф-ТА ИВТ ЯРГУ ИМ. П.Г. ДЕМИДОВА

г. Ярославль

Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)

Основная литература:

1. Кузьмин, Е. В., Верификация моделей программ : учеб. пособие для вузов / Е. В. Кузьмин ; под ред. В. А. Соколова, Ярославль, ЯрГУ, 2008, 174с
2. Кузьмин, Е. В., Верификация моделей программ [Электронный ресурс] : учеб. пособие для вузов / Е. В. Кузьмин ; под ред. В. А. Соколова, Ярославль, ЯрГУ, 2008, 174с
3. Сеницын, С. В., Основы разработки программного обеспечения на примере языка С [Электронный ресурс] / С. В. Сеницын, О. И. Хлыткин. - 2-е изд., испр., М., Национальный Открытый Университет ИНТУИТ, 2016, 212с
4. Кузьмин Е.В. Введение в теорию вычислительных процессов и структур. – Учебное пособие, Ярославль, ЯрГУ, 2006. – 140 с.
5. Грис Д. Наука программирования. – М.: Мир, 1984. – 416 с.
6. Карпов Ю. Г. Model Checking. Верификация параллельных и распределенных программных систем. – СПб.: БХВ-Петербург, 2010. – 560 с.
7. Кларк, Э. М., Верификация моделей программ : Model Checking : пер. с англ. / Э. М. Кларк мл., О. Грамберг, Д. Пелед ; под ред. Р. Смелянского, М., МЦНМО, 2002, 416с

Дополнительная литература:

1. Сеницын, С. В., Верификация программного обеспечения [Электронный ресурс] : курс / С. В. Сеницын, Н. Ю. Налютин, М., Интернет-Университет Информационных Технологий, 2007, 367с
2. Кузьмин, Е. В., Структурированные системы переходов : учеб. пособие для вузов / Е. В. Кузьмин, В. А. Соколов, М., ФИЗМАТЛИТ, 2006, 174с
3. Карпов Ю. Г. Теория автоматов. – СПб.: Питер, 2003. – 208 с.
4. Математическая логика в программировании: сб. статей 1980—1988 гг.: Пер. с англ. – М.: Мир, 1991. – 408 с.
5. Минский М. Вычисления и автоматы – М.: Мир, 1971. – 268 с.
6. Непомнящий В. А., Рязкин М. О. Прикладные методы верификации программ – М.: Радио и связь, 1988. – 256 с.
7. Питерсон Дж. Теория сетей Петри и моделирование систем. – М.: Мир, 1984. – 263 с.
8. Хоар Ч. Взаимодействующие последовательные процессы. – М.: Мир, 1989. – 264 с.
9. Хопкрофт Д., Мотвани Р., Ульман Д. Введение в теорию автоматов, языков и вычислений. – М.: Вильямс, 2002. – 528 с.

Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине (модулю)

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.uniya.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.uniya.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/паролю.

3. Электронная картотека «Книгообеспеченность» (http://www.lib.uniya.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.

Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля), включая перечень информационных справочных систем (при необходимости)

1. SMV. Symbolic Model Verifier. Carnegie Mellon University.
<http://www.cs.cmu.edu/~modelcheck/smv.html>
2. SPIN. <http://spinroot.com/spin/whatispin.html>
3. UPPAAL. <http://www.uppaal.com>

Перечень информационных технологий, используемых при изучении дисциплины, включая программное обеспечение

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");
- свободно распространяемые для учебных целей средства верификации SPIN, SMV и Uppaal.

Учебно-методические указания и рекомендации к изучению тем лекционных и практических занятий, самостоятельной работе студентов

Содержание дисциплины

Дедуктивный анализ корректности программ на примере «простого» языка программирования. Спецификация программ с помощью пред- и постусловий. Доказательство корректности программ относительно спецификации, инвариантов и ограничивающей функции. Построение инвариантов и ограничивающих функций. Построение моделей параллельных и распределенных систем. Асинхронные и синхронные процессы. Взаимодействие процессов. Структура Крипке. Метод проверки модели. Верификация моделей и теория автоматов. Автоматы над бесконечными словами. Структура Крипке как автомат Бюхи. Темпоральная логика линейного времени LTL. Формула LTL как обобщенный автомат Бюхи. Редукция автомата Бюхи для формулы LTL. Пересечение языков структуры Крипке и автомата Бюхи. Проверка пустоты автомата Бюхи. Проверка модели «на лету». Верификация моделей для логики CTL. Темпоральная логика CTL. Верификация моделей для CTL. Верификация моделей и неподвижные точки. Символьная верификация моделей для CTL. Двоичные диаграммы решений. Диаграммы ROBDD. Построение и манипуляция ROBDD. Теория временных автоматов. Временные автоматы Бюхи и Мюллера. Моделирование, спецификация и верификация систем реального времени с помощью временных автоматов

Варианты контрольной работы.

I. Докажите формально, что следующий алгоритм предназначен для записи в переменную z произведения чисел a и b при $b \geq 0$ без использования операции умножения.

```

 $x, y, z := a, b, 0;$ 
do  $y > 0$  &  $\text{even}(y) \rightarrow y, x := y/2, x+x$  []
     $\text{odd}(y) \rightarrow y, z := y-1, z+x$ 
od

```

Критерии оценивания

Оценка	Критерии
Отлично	<p>Знает способы дедуктивного доказательства корректности программ; стратегию спецификации и доказательства корректности программ, написанных на процедурном языке высокого уровня.</p> <p>Умеет проводить спецификацию программ на языке предикатов; применять метод доказательства теорем для доказательства корректности программ, написанных на языках высокого уровня.</p> <p>Владеет формальными методами дедуктивного анализа корректности программ (алгоритмов).</p> <p>Произвёл полностью правильное доказательство корректности данной программы.</p>
Хорошо	<p>Знает способы дедуктивного доказательства корректности программ; стратегию спецификации и доказательства корректности программ, написанных на процедурном языке высокого уровня.</p> <p>Умеет проводить спецификацию программ на языке предикатов; применять метод доказательства теорем для доказательства корректности программ, написанных на языках высокого уровня.</p> <p>Владеет формальными методами дедуктивного анализа корректности программ (алгоритмов).</p> <p>Произвёл правильное доказательство корректности данной программы относительно спецификации, которая содержит не полное описание входного и выходного множества допустимых значений программных переменных.</p>
Удовлетворительно	<p>Знает способы дедуктивного доказательства корректности программ; стратегию спецификации и доказательства корректности программ, написанных на процедурном языке высокого уровня.</p> <p>Умеет проводить спецификацию программ на языке предикатов; применять метод доказательства теорем для доказательства корректности программ, написанных на языках высокого уровня.</p> <p>Владеет формальными методами дедуктивного анализа корректности программ (алгоритмов).</p> <p>Произвёл правильное доказательство корректности данной программы относительно неверно построенной спецификации.</p>
Неудовлетворительно	<p>Не знает способы дедуктивного доказательства корректности программ; стратегию спецификации и доказательства корректности программ, написанных на процедурном языке высокого уровня.</p> <p>Не умеет проводить спецификацию программ на языке предикатов; применять метод доказательства теорем для доказательства корректности программ, написанных на языках высокого уровня.</p> <p>Не владеет формальными методами дедуктивного анализа корректности программ (алгоритмов).</p> <p>Не выполнил доказательство корректности данной программы.</p>

II. Используя определения синтаксиса и семантики формул темпоральной логики LTL (или CTL) произведите спецификацию с последующей верификацией с помощью средства верификации SPIN и SMV указанных ниже свойств для системы асинхронных параллельных процессов со взаимным исключением, представленной на рис. 1.1. в пособии [4]:

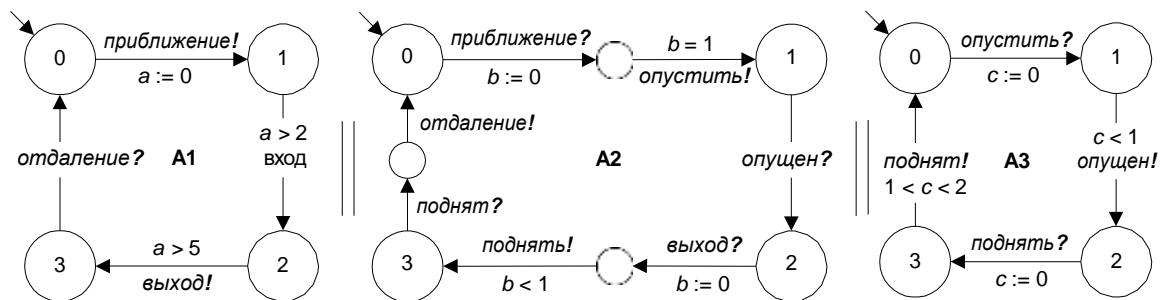
1. «Взаимное исключение». Процессы никогда не окажутся в своих критических участках одновременно. Другими словами, система ни при каких обстоятельствах не попадет в состояние, в котором процессы $Pr1$ и $Pr2$ будут находиться в своих локальных состояниях с номерами 7.
2. «Отсутствие взаимной блокировки». Не существует ситуации, при которой ни процесс $Pr1$, ни процесс $Pr2$ не могут перейти в другое локальное состояние.
3. «Отсутствие бесконечного откладывания процессов». Если один из процессов пожелает войти в свою критическую секцию, он обязательно в нее войдет. Другими словами, исключается возможность, при которой один из процессов бесконечно часто заходит в свой критический участок, а второй процесс вынужден постоянно откладывать свой вход в этот участок, бесконечно долго, таким образом, ожидая своей очереди.
4. «Справедливость». Если оба процесса одновременно вышли из своих некритических участков, т. е. процессы $Pr1$ и $Pr2$ находятся в локальных состояниях с номерами 1, то в свой критический участок первым обязательно войдет тот процесс, приоритет которого в данный момент выше (приоритет определяется исходя из значения переменной trn).
5. «Отсутствие чередования». Если первый процесс только что посетил свой критический участок и хочет вновь в него войти, а второй процесс не выражает такого желания, то первому нет необходимости дожидаться, пока второй процесс проявит себя, войдет в критическую область, а затем покинет ее. Другими словами, посещения процессами своих критических участков не обязательно должны чередоваться. Если один из процессов навсегда остается в своем некритическом участке (например, закончил работу), то это не оказывает никакого влияния на возможность входа другого процесса в свой критический участок.

Критерии оценивания

Оценка	Критерии
Отлично	Знает методы автоматической проверки корректности программной модели. Умеет строить программные модели и проводить спецификацию и верификацию программных свойств на языке темпоральной логики. Владеет формальными методами моделирования и спецификации программ. Произвёл правильную запись всех требуемых программных свойств на языке темпоральной логики LTL.
Хорошо	Знает методы автоматической проверки корректности программной модели. Умеет строить программные модели и проводить спецификацию и верификацию программных свойств на языке темпоральной логики. Владеет формальными методами моделирования и спецификации программ. Произвёл правильную запись более половины требуемых

	программных свойств на языке темпоральной логики LTL.
Удовлетворительно	Знает методы автоматической проверки корректности программной модели. Умеет строить программные модели и проводить спецификацию и верификацию программных свойств на языке темпоральной логики. Владеет формальными методами моделирования и спецификации программ. Произвёл правильную запись нескольких требуемых программных свойств на языке темпоральной логики LTL.
Неудовлетворительно	Не знает методы автоматической проверки корректности программной модели. Не умеет строить программные модели и проводить спецификацию и верификацию программных свойств на языке темпоральной логики. Не владеет формальными методами моделирования и спецификации программ. Не произвёл правильную запись ни одного из требуемых программных свойств на языке темпоральной логики LTL.

III. Проведите проверку свойств модели (реального времени) железнодорожного переезда с помощью программного средства верификации Uppaal, базирующемся на теории временных автоматов.



Модель автоматной системы в виде параллельной композиции синхронизирующихся временных автоматов

Свойства:

- 1) шлагбаум не может быть опущен (переезд не может быть закрыт) более чем на 10 мин., т.е. состояние автомата A3 «0. шлагбаум поднят» достигается как максимум через 10 мин. после попадания им в состояние «2. Шлагбаум опущен».
- 2) когда поезд войдёт в переезд, шлагбаум уже должен быть опущен.

Кроме того, при сдаче задания могут быть заданы вопросы по теории. Материал по теме задания должен быть подробно и полно освещен и проиллюстрирован на примере решения этого задания.

Критерии оценивания

Оценка	Критерии
Отлично	Знает методы автоматической проверки корректности

	<p>программной модели. Умеет строить модели систем реального времени с помощью формализма временных автоматов и проводить спецификацию свойств таких систем на языках временных темпоральных логик. Владеет формальными методами анализа корректности программ.</p> <p>Правильно построена временная модель, правильно записаны временные свойства, проведена проверка свойств модели.</p>
Хорошо	<p>Знает методы автоматической проверки корректности программной модели. Умеет строить модели систем реального времени с помощью формализма временных автоматов и проводить спецификацию свойств таких систем на языках временных темпоральных логик. Владеет формальными методами анализа корректности программ.</p> <p>Правильно построена временная модель, записаны временные свойства с небольшими неточностями, проведена проверка свойств модели.</p>
Удовлетворительно	<p>Знает методы автоматической проверки корректности программной модели. Умеет строить модели систем реального времени с помощью формализма временных автоматов и проводить спецификацию свойств таких систем на языках временных темпоральных логик. Владеет формальными методами анализа корректности программ.</p> <p>Построена временная модель с неточностями, записаны временные свойства с неточностями, проведена проверка свойств модели.</p>
Неудовлетворительно	<p>Не знает методы автоматической проверки корректности программной модели. Не умеет строить модели систем реального времени с помощью формализма временных автоматов и проводить спецификацию свойств таких систем на языках временных темпоральных логик. Не владеет формальными методами анализа корректности программ.</p> <p>Неправильно построена временная модель, неправильно записаны временные свойства, не проведена проверка свойств модели.</p>

Вопросы на Экзамен.

Теория семантики и верификации программ

1. Корректность программ. Спецификация и верификация. Верификация и тестирование.
2. Спецификация программ. Предусловие. Постусловие. Примеры спецификаций программ. Представление начальных и конечных значений переменных. Наброски доказательств.
3. Семантика простого языка программирования. Преобразователь предикатов wr . Спецификация программ через преобразователь предикатов wr . Свойства wr .
4. Семантика простого языка программирования. Команды `skip`, `abort` и композиция команд. Команда присваивания. Кратное присваивание. Присваивание элементу массива.

5. Семантика простого языка программирования. Команда выбора. Примеры. Теорема о команде выбора. Доказательство корректности программ, не содержащих команд повторения. Примеры доказательств.
6. Семантика простого языка программирования. Команда повторения. Инвариант. Ограничивающая функция. Теорема о цикле, инварианте и ограничивающей функции. Доказательство программ, содержащих циклы. Список условий для проверки цикла. Примеры доказательств корректности цикла.
7. Построение программ. Стратегия построения команд выбора.
8. Построение программ. Построение циклов исходя из инвариантов и ограничений.
9. Построение инвариантов цикла. Теория воздушного шарика. Основная идея и стратегии построения инвариантов.
10. Построение инвариантов цикла методом устранения конъюнктивного члена. Примеры.
11. Построение инвариантов цикла методом замены константы переменной. Примеры.
12. Построение инвариантов цикла методом расширения области значений переменной. Примеры.
13. Построение инвариантов цикла методом комбинирования пред- и постусловий. Примеры.

Модели вычислительных процессов

1. Модели вычислительных процессов: сети Петри, взаимодействующие процессы и т.д.
2. Взаимодействие процессов. Асинхронные и синхронные процессы. Синхронизация параллельных процессов. Проблема критических участков. Анализ подходов к решению проблемы. Алгоритм Деккера. Программная реализация взаимоисключений.
3. Структура Крипке и метод автоматической верификации моделей.

Верификация моделей и теория автоматов

1. Автоматы над бесконечными словами.
2. Структура Крипке как автомат Бюхи.
3. Темпоральная логика линейного времени LTL.
4. Формула LTL как обобщенный автомат Бюхи. Замыкание формулы. Правила разметки последовательностей. Построение обобщенного автомата Бюхи по формуле LTL.
5. Редукция обобщенного автомата Бюхи для формулы логики LTL. Исключение избыточных переходов. Построение автомата исходя из необходимости. Определение эквивалентных состояний.
6. Пересечение языков структуры Крипке и автомата Бюхи.
7. Проверка пустоты автомата Бюхи.
8. Проверка модели «на лету».

Верификация моделей для логики CTL

1. Темпоральная логика CTL.
2. Верификация моделей для CTL.
3. Верификация моделей и неподвижные точки.

4. Символьная верификация моделей для CTL.

Двоичные диаграммы решений

1. Двоичные диаграммы решений ROBDD.
2. Построение и манипуляция ROBDD. Процедура Mk.
3. Построение и манипуляция ROBDD. Процедура Build.
4. Построение и манипуляция ROBDD. Процедура Apply.
5. Построение и манипуляция ROBDD. Процедура Restrict.
6. Построение и манипуляция ROBDD. Кванторы существования и всеобщности.

Теория временных автоматов

1. Детерминированные и недетерминированные временные автоматы Бюхи и Мюллера.
2. Разрешимые свойства временных автоматов.
3. Верификация систем реального времени с помощью временных автоматов.