

РОССИЙСКАЯ АКАДЕМИЯ НАУК

Федеральное государственное бюджетное учреждение науки
Институт системного программирования
Российской академии наук

«УТВЕРЖДАЮ»

Директор ИСП РАН
академик РАН,
д.ф.-м.н., профессор
В.П.Иванников

« ____ » _____ 2012 г.

РАБОЧАЯ ПРОГРАММА

УЧЕБНОГО КУРСА

«Программная инженерия. Дополнительные главы»

для подготовки аспирантов по научной специальности
05.13.11 — Математическое и программное обеспечение
вычислительных машин, комплексов и компьютерных сетей

Москва 2012

1. Цели и задачи курса

Курс «Программная инженерия. Дополнительные главы» ставит своей целью расширение теоретических знаний и практических навыков обучаемых в области программной инженерии до набора, необходимого для продуктивного участия в проектах по созданию и сопровождению сложных программных систем (ПС) с повышенными требованиями к надежности и качеству и возможности взять на себя руководство таким проектом.

Достижение названной цели основано на решении следующих учебных задач курса:

- 1) овладеть набором понятий и общей методологией анализа программных систем, используемых в программной инженерии;
- 2) получить теоретические знания о методах работы с требованиями к ПС, методах проектирования и построения ПС, методах анализа свойств и контроля различных аспектов их качества, методах организации работ по созданию и сопровождению ПС;
- 3) сформировать практические навыки применения некоторых методов организации работ по разработке и сопровождению ПС, а также ряда методов работы с требованиями, проектирования, анализа свойств и контроля качества ПС.

2. Место курса в структуре послевузовского профессионального образования (аспирантура)

Курс «Программная инженерия. Дополнительные главы» относится к курсам по выбору учебного плана подготовки аспирантов по научной специальности 05.13.11 «Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей».

Изучение данного курса базируется на следующих курсах подготовки бакалавров, магистров или специалистов:

- «Языки программирования»;
- «Основы программной инженерии»;
- «Объектно-ориентированный анализ и проектирование»;
- «Базы данных»;
- «Сети ЭВМ и безопасность»

Для успешного изучения курса аспиранту необходимо знать общесистемное программное обеспечение, основные средства разработки ПО, уметь работать с персональной ЭВМ и с клиент-серверными инструментами поддержки разработки ПО.

Получаемые в рамках курса знания будут востребованы при подготовке к кандидатскому экзамену по научной специальности 05.13.11 «Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей», в научно-исследовательской работе и при выполнении диссертации на соискание ученой степени кандидата технических наук.

3. Требования к результатам освоения курса

В результате изучения курса «Программная инженерия. Дополнительные главы» аспирант должен:

- иметь представление о месте и роли программной инженерии в современном обществе, о ее взаимосвязи с другими дисциплинами;

- знать систему базовых понятий программной инженерии, методы организации работ по созданию и сопровождению ПС, методы выполнения основных видов работ в рамках жизненного цикла ПС: анализа требований, проектирования и построения программного обеспечения, анализа свойств ПС и контроля их качества;
- уметь применять на практике изученные методы организации работ по созданию и сопровождению ПС и методы выполнения основных видов работ в рамках жизненного цикла ПС: анализа требований, проектирования и построения программного обеспечения, анализа свойств ПС и контроля их качества.

4. Содержание и структура курса

4.1 Содержание разделов курса

№	Наименование раздела	Содержание раздела	Форма текущего контроля
1	Предмет и основные понятия программной инженерии	Проблемы разработки сложного программного обеспечения (ПО). Общие принципы программной инженерии: абстракция и уточнение, модульность, повторное использование. Жизненный цикл ПО и процессы его разработки. Основные модели жизненного цикла: водопадная, итеративная, спиральная. Стандарты программной инженерии.	Т
2	Управление разработкой ПС	Управление разработкой программного обеспечения. Управление содержанием проекта и качеством. Планирование и управление ресурсами. Метрики программного обеспечения и их использование. Управление рисками. Управление персоналом. Структура и культура организаций. Мотивация работников. Управление конфликтами. Ведение переговоров и информационное обеспечение проектов. Различные методы организации разработки ПО: RUP, XP, MSF, SCRUM, FDD.	Т

3	Анализ требований и контроль качества ПС	Анализ требований к программному обеспечению. Анализ предметной области. Методы выделения требований. Методы описания и систематизации требований. Использование различных видов графических диаграмм при описании требований. Понятие качества программного обеспечения и его основные характеристики. Методы контроля качества программного обеспечения. Различные техники проведения экспертизы. Проверка моделей. Дедуктивная верификация. Тестирование и его разновидности. Основные методы построения тестов.	Т
4	Архитектура ПС и методы проектирования	Понятие архитектуры программного обеспечения. Методы разработки и анализа архитектуры программного обеспечения на основе сценариев его работы и модификации. Методы описания архитектуры. Образцы анализа, архитектурные стили, образцы проектирования, идиомы и образцы организации работ. Системы образцов. Примеры образцов и их использования. Принципы проектирования пользовательского интерфейса.	Т

4.2 Структура курса

Общая трудоемкость курса составляет 2,5 зачетные единицы (90 часов).

Вид работы	Трудоемкость, часов
	2 курс
Общая трудоемкость	90
Аудиторная работа:	32
<i>Лекции (Л)</i>	32
<i>Практические занятия (ПЗ)</i>	-
<i>Лабораторные работы (ЛР)</i>	-
Самостоятельная работа:	58
Самостоятельное изучение разделов	-
Самоподготовка (проработка и повторение лекционного материала и материала учебников и учебных пособий, выполнение практических заданий)	58
Вид итогового контроля (зачет, экзамен)	Кандидатский экзамен

Трудоемкость отдельных разделов курса.

№	Наименование разделов	Количество часов				
		Всего	Аудиторная работа			Вне-ауд. работа СР
			Л	ПЗ	ЛР	
1	Предмет и основные понятия программной инженерии	8	4	-	-	4
2	Управление разработкой ПС	22	8	-	-	14
3	Анализ требований и контроль качества ПС	30	10	-	-	20
4	Архитектура ПС и методы проектирования	30	10	-	-	20
	<i>Итого:</i>	90	32	-	-	58

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

Форма контроля знаний:

- кандидатский экзамен по специальности.

Контрольно-измерительные материалы

На кандидатском экзамене аспирант должен продемонстрировать знания в объеме основной программы кандидатского экзамена по специальности 05.13.11 «Математическое обеспечение вычислительных машин, комплексов и компьютерных сетей», а также дополнительной программы, в которую, в зависимости от выбранной аспирантом специализации, могут входить вопросы, рассматриваемые в данном курсе.

Перечень контрольных вопросов для дополнительной программы:

1. Предмет и основные понятия программной инженерии
 - a. Что является объектом и предметом программной инженерии?
 - b. Каковы наиболее существенные характеристики сложных ПС?
 - c. Каковы основные принципы программной инженерии?
 - d. Приведите примеры использования различных видов абстракции в ПО.
 - e. Каковы основные цели выделения интерфейсов?
 - f. Каковы основные цели многократного использования кода?
 - g. Что такое жизненный цикл ПО?
 - h. Каковы основные виды деятельности при разработке и сопровождении ПО?
 - i. Приведите примеры артефактов разработки ПО.

- j. Каковы основные модели жизненного цикла ПО?
- k. Чем спиральная модель ЖЦ ПО отличается от итеративной?

2. Управление разработкой ПС

- a. Каковы основные виды деятельности при управлении разработкой ПО?
- b. Какие техники построения структуры работ в проекте вы знаете?
- c. Какие техники планирования работ в проекте вы знаете?
- d. Что такое критический путь?
- e. Что такое конфигурационное управление?
- f. Что такое управление качеством? Какие задачи оно решает?
- g. В чем различия открытого и закрытого стилей управления?
- h. Какие методы мотивации сотрудников вы знаете?
- i. Какие методы разрешения конфликтов в коллективе вы знаете?
- j. Какие техники ведения переговоров вы знаете?
- k. Почему книга Брукса называется «Мифический человеко-месяц»?
- l. Что такое функциональные точки?
- m. Какие типы организационных структур вы знаете?
- n. Какие типы организационной культуры вы знаете?
- o. Каковы структура итерации в RUP?
- p. Какова структура итерации в XP?
- q. Какова структура итерации в SCRUM?
- r. Приведите 5 примеров техник, используемых в рамках XP.
- s. Какую роль конфигурационное управление играет в XP?

3. Анализ требований и контроль качества ПС

- a. Чем анализ предметной области отличается от анализа требований к ПО?
- b. Каковы могут быть источники требований к ПО?
- c. Чем различаются функции ПО и технические требования к ПО?
- d. Какие техники выделения требований вы знаете?
- e. Какими свойствами должны обладать технические требования к ПО?
- f. Какие графические диаграммы используются для описания требований?
- g. Что такое диаграммы потоков данных?
- h. Каковы основные элементы диаграмм сущностей и связей?
- i. Каковы основные элементы диаграмм вариантов использования?
- j. Что такое вариант использования?
- k. Каковы основные характеристики качества ПО?

- l. Какие методы контроля качества ПО вы знаете?
- m. Что такое экспертиза? Какие виды экспертиз вы знаете?
- n. Как проводится инспекция ПО по Фагану?
- o. Что такое дедуктивная верификация ПО?
- p. Что такое тестирование ПО?
- q. Каковы основные задачи тестирования?
- r. Чем различаются модульное, интеграционное и системное тестирование?
- s. Чем различаются тестирование черного ящика и белого ящика?
- t. Что такое регрессионное тестирование?
- u. Что такое тестовый вариант?

4. Архитектура ПС и методы проектирования

- a. Что такое модульная структура и архитектура ПО?
- b. Какие архитектурные стили вы знаете?
- c. В каких ситуациях используются конвейерные архитектуры?
- d. В каких ситуациях используются архитектуры на основе репозитория?
- e. Как проводится анализ архитектуры методом SAAM?
- f. Какие методы проектирования ПО вы знаете?
- g. Какие графические диаграммы используются для описания проектных решений?
- h. В чем различия диаграмм классов и компонентов в UML?
- i. В чем различия диаграмм состояний и сценариев в UML?
- j. Приведите примеры образцов анализа.
- k. Приведите примеры образцов проектирования.
- l. Приведите примеры образцов организации работ.
- m. Какие образцы проектирования используются при создании распределенного ПО?
- n. Какие образцы проектирования используются при создании интерактивного ПО?
- o. Каковы принципы оценки понятности интерфейса пользователя?
- p. Как оценивается производительность пользователей методом GOMS?
- q. В чем различие сущностных и обычных вариантов использования?
- r. Как проводится тестирование удобства использования ПО?
- s. Как проводится экспертиза удобства использования ПО?

6. Учебно-методическое обеспечение курса

6.1 Рекомендуемая литература

6.1.1 Основная литература

1. И. Соммервилл. Инженерия программного обеспечения. М.: Вильямс, 2002.
2. I. Sommerville. Software Engineering. 9-th edition. Addison-Wesley, 2010.
3. К. Гецци, М. Джазайери, Д. Мандриоли. Основы инженерии программного обеспечения. БХВ-Петербург, 2005.
4. Б. Майер. Объектно-ориентированное программирование. Концепции разработки. М.: Русская редакция, 2004.
5. R. S. Pressman. Software Engineering. A Practitioner's Approach. 7-th edition. McGraw-Hill Science/Engineering/Math, 2009.
6. S. L. Pfleeger, J. Atlee. Software Engineering. Theory and Practice. 4-th edition. Prentice Hall, 2009.

6.1.2 Дополнительная литература

1. Л. Басс, П. Клементс, Р. Кацман. Архитектура программного обеспечения на практике. СПб.: Питер, 2006.
2. У. Ройс. Управление проектами по созданию программного обеспечения. М.: Лори, 2002.
3. Л. Константайн, Л. Локвуд. Разработка программного обеспечения. СПб.: Питер, 2004.
4. Э. Дж. Брауде. Технология разработки программного обеспечения. СПб.: Питер, 2004.
5. Е. А. Жоголев. Лекции по технологии программирования: Учебное пособие. М.: Издательский отдел факультета ВМиК МГУ, 2001.
6. В. В. Липаев. Методы обеспечения качества крупномасштабных программных средств. М., Синтег, 2003.
7. Д. Леффингуэлл, Д. Уидриг. Принципы работы с требованиями к программному обеспечению. Унифицированный подход. М.: Вильямс, 2002.
8. А. Коберн. Современные методы описания требований к системам. М.: Лори, 2002.
9. А. Якобсон, Г. Буч, Дж. Рамбо. Унифицированный процесс разработки программного обеспечения. СПб.: Питер, 2002.
10. К. Бек. Экстремальное программирование. СПб.: Питер, 2002.
11. Э. Хант, Д. Томас. Программист-прагматик. М.: Лори, 2004.
12. Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влиссидес. Приемы объектно-ориентированного проектирования. Паттерны проектирования. СПб.: Питер-ДМК, 2001.
13. М. Fowler. Analysis Patterns: Reusable Object Models. Addison-Wesley, 1997.
14. М. Фаулер и др. Архитектура корпоративных программных приложений. М.: Вильямс, 2004.
15. Ф. Брукс. Мифический человеко-месяц или Как создаются программные системы. СПб.: Символ-Плюс, 2001.
16. Т. Демарко, Т. Листер. Человеческий фактор: успешные проекты и команды. СПб.: Символ-Плюс, 2005.
17. S. McConnell. Code Complete: A Practical Handbook of Software Construction. 2-nd edition. Microsoft Press, 2004.

6.1.3 Периодические издания

Журналы:

1. Программирование
2. IEEE Transactions on Software Engineering
3. ACM Computing Reviews
4. ACM Computing Surveys

6.1.4 Интернет-ресурсы

1. IEEE Guide to the Software Engineering Body of Knowledge, 2004.
http://www.swebok.org/ironman/pdf/SWEBOK_Guide_2004.pdf.
2. SWEBOK V3 Review.
<http://computer.centraldesktop.com/swebokv3review/>.
3. Единое окно доступа к образовательным ресурсам: информационная система. – Электрон. дан. – ФГУ ГНИИ ИТТ «Информика», 2005 – 2011; Министерство образования и науки РФ, 2005 – 2010.
<http://window.edu.ru/>.

6.2 Средства обеспечения освоения курса

6.2.1 Методические указания и материалы

1. В. В. Кулямин. Технологии программирования. Компонентный подход. М.: Интернет-университет информационных технологий, Бином, 2007.

6.2.2 Программное обеспечение

1. Платформа для разработки ПО Java SE.
<http://www.oracle.com/technetwork/java/javase/downloads/index.html>.
2. Среда разработки с открытым кодом Eclipse.
<http://www.eclipse.org>.
3. Среда выполнения модульных тестов TestNG.
<http://testng.org>.

7. Материально-техническое обеспечение курса

Для получения необходимой информации и самостоятельной работы аспирантов используются web-ресурсы Интернет и локальная библиотека электронных материалов.

При чтении лекций используется курс лекций Кулямина В. В. «Технологии программирования. Компонентный подход».

В компьютерных классах ИСП РАН (ауд. 109) аспиранты могут самостоятельно ознакомиться с программным обеспечением, используемым при разработке и сопровождении программных систем.

Программу составил к.ф.–м.н. Кулямин В.В.

Программа принята на заседании Ученого Совета ИСП РАН
протокол № 2012-5 от 23 мая 2012 г.