

ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«Приднестровский государственный университет
имени Т.Г. Шевченко»

Рыбницкий филиал

Кафедра информатики и программной инженерии

«УТВЕРЖДАЮ»

Директор филиала ПГУ им. Т.Г. Шевченко в г. Рыбнице
профессор  И.А. Павлинов

« 15 » 09 2017 г.



РАБОЧАЯ ПРОГРАММА

на 2017/2018 учебный год

УЧЕБНОЙ ДИСЦИПЛИНЫ

«ВВЕДЕНИЕ В СПЕЦИАЛЬНОСТЬ»

Направление подготовки:

09.03.04 «Программная инженерия»

Профиль подготовки:

«Разработка программно-информационных систем»

квалификация (степень) выпускника:

бакалавр

Форма обучения:

очная

Рыбница 2017

Рабочая программа дисциплины «*Введение в специальность*» /сост.: Л.А. Балан, Е.С. Гарбузняк. – Рыбница: филиал ПГУ им. Т.Г. Шевченко в г. Рыбнице, 2017. – 20 с.

РАБОЧАЯ ПРОГРАММА ПРЕДНАЗНАЧЕНА ДЛЯ ПРЕПОДАВАНИЯ ДИСЦИПЛИНЫ ВАРИАТИВНОЙ ЧАСТИ БЛОКА ДИСЦИПЛИН (МОДУЛЕЙ) СТУДЕНТАМ ОЧНОЙ ФОРМЫ ОБУЧЕНИЯ ПО НАПРАВЛЕНИЮ ПОДГОТОВКИ 09.03.04 – «ПРОГРАММНАЯ ИНЖЕНЕРИЯ», ПРОФИЛЮ ПОДГОТОВКИ «РАЗРАБОТКА ПРОГРАММНО-ИНФОРМАЦИОННЫХ СИСТЕМ».

Рабочая программа составлена с учетом Федерального Государственного образовательного стандарта высшего образования по направлению подготовки 09.03.04 – «Программная инженерия», утвержденного приказом Министерства образования и науки Российской Федерации от 12 марта 2015 г. № 229.

Составители:



Л.А. Балан, доцент



Е.С. Гарбузняк, ст. преподаватель

1. Цели и задачи освоения дисциплины

Цель освоения дисциплины «Введение в специальность» – дать систематизированное представление о современном комплексе задач, методов и стандартах программной инженерии, создании и эволюции сложных, многоверсионных, тиражируемых программных продуктах высокого качества, необходимых для практического использования на последующих этапах обучения и в профессиональной сфере деятельности будущего специалиста.

Задачами дисциплины являются:

- получение сведений о существующих уровнях образования, структуре и содержании государственного образовательного стандарта и учебного плана направления;
- приобретение знаний об основных концепциях, принципах и направлениях развития программной инженерии;
- получение сведений об основных этапах жизненного цикла промышленной разработки и области применения прикладных программных продуктов;
- получение знаний о понятии качественной программы и стандартизации в области информационных технологий;
- формирование навыков самостоятельной работы с первоисточниками технической информации;
- формирование умений технически грамотно формулировать вопросы по разработке программно-информационных систем.

2. Место дисциплины в структуре ООП ВО

Дисциплина «Введение в специальность» является обязательной дисциплиной вариативной части (Б1.В.ОД.10) блока дисциплин (модулей) подготовки студентов по направлению 09.03.04 – «Программная инженерия», профилю подготовки «Разработка программно-информационных систем».

Для освоения дисциплины «Введение в специальность» студенты используют знания, умения, навыки, способы деятельности и установки, полученные и сформированные в ходе изучения дисциплины «Теоретическая информатика», а также дисциплины «Основы программирования», изучаемой параллельно.

Изучение дисциплины «Введение в специальность» является базой для дальнейшего освоения студентами дисциплин основной образовательной программы по направлению.

3. Требования к результатам освоения дисциплины «Введение в специальность»

Изучение дисциплины направлено на формирование следующих компетенций:

Код компетенции	Формулировка компетенции
ОК-7	способность к самоорганизации и самообразованию
ОПК-3	готовность применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов
ОПК-4	способность осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий
ПК-1	готовность применять основные методы и инструменты разработки программного обеспечения

ПК-3	владение навыками использования различных технологий разработки программного обеспечения
ПК-4	владение концепциями и атрибутами качества программного обеспечения (надежности, безопасности, удобства использования), в том числе роли людей, процессов, методов, инструментов и технологий обеспечения качества
ПК-5	владение стандартами и моделями жизненного цикла
ПК-7	владение методами управления процессами разработки требований, оценки рисков, приобретения, проектирования, конструирования, тестирования, эволюции и сопровождения
ПК-8	владение основами групповой динамики, психологии и профессионального поведения, специфичных для программной инженерии
ПК-15	способность готовить презентации, оформлять научно-технические отчеты по результатам выполненной работы, публиковать результаты исследований в виде статей и докладов на научно-технических конференциях
ПК-23	владение навыками проведения практических занятий с пользователями программных систем

В результате освоения дисциплины студент должен:

3.1. Знать:

- структуру и содержание государственного образовательного стандарта и учебного плана направления;
- основы моделирования и анализа программных систем, разработки, выявления, спецификации и управления требованиями;
- концепции и стратегии архитектурного проектирования и конструирования программного продукта;
- концепции эволюционного развития программного обеспечения;
- стандарты качества программного продукта и процессов его обеспечения;
- основные стандарты информационно-коммуникационных систем и технологий.

3.2. Уметь:

- использовать основную документацию по стандартизации деятельности инженера-программиста;
- различать этапы жизненного цикла программного продукта;
- различать подходы к инженерному проектированию в конкретных предметных областях.

3.3. Владеть:

- пониманием социальной значимости своей будущей профессии, обладанием высокой мотивацией к выполнению профессиональной деятельности;
- навыками разработки программной документации;
- навыками персональной и командной разработки программного обеспечения;
- навыками самостоятельного анализа новых тенденций и концепций программной инженерии.

4. Структура и содержание дисциплины (модуля)

Рабочая программа учебной дисциплины рассчитана на 1 семестр. Трудоемкость составляет 2 зачетные единицы, 72 часа. В том числе 18 часов отводится на лекционные занятия, 18 часов – на практические занятия, 36 часов – на самостоятельную работу.

4.1. Распределение трудоемкости в з.е./часах по видам аудиторной и самостоятельной работы студентов по семестрам

Семестр	Трудоем- кость, з.е./часы	Количество часов					Форма итогового контроля
		В том числе					
		Аудиторных				Самост. работы	
		Всего	Лекций	Лаб. раб.	Практич. зан		
2	2/72	36	18	—	18	36	зачет
Итого:	2/72	36	18	—	18	36	зачет

4.2. Распределение видов учебной работы и их трудоемкости по разделам дисциплины

№ раз- дела	Наименование разделов	Количество часов				
		Всего	Аудиторная работа			Внеауд. работа (СР)
			Л	ПЗ	ЛР	
1	Введение в программную инженерию. Жизненный цикл программного обеспечения	40	10	10	–	20
2	Инженерия приложений и инженерия предметной области	16	4	4	–	8
3	Методы управления проектом, риском и конфигурацией	16	4	4	–	8
Итого:		72	18	18	–	36

4.3. Тематический план по видам учебной деятельности

Лекции

№ п/п	Номер раздела дисциплины	Объем часов	Тема лекции	Учебно- наглядные пособия
1	1	2	Основные положения федерального государственного образовательного стандарта по направлению 09.03.04 – «Программная инженерия» и профессиональных стандартов в области IT-индустрии	Компьютерные презентации
2		2	Модели жизненного цикла для разработки программных систем	Компьютерные презентации
3		2	Определение понятий и видов требований при разработке программных систем	Компьютерные презентации
4		2	Методы проектирования ПО	Компьютерные презентации
5		2	Методы системного, теоретического программирования	Компьютерные презентации
6	2	2	Инженерия повторно используемого компонента. Методы тестирования программ	Компьютерные презентации
7		2	Модель качества ПО. Модели оценки и надежности	Компьютерные презентации

8	3	2	Теория управления в программной инженерии	Компьютерные презентации
9		2	Инструменты создания программных систем	Компьютерные презентации
Итого:		18		

Практические (семинарские) занятия

№ п/п	Номер раздела дисциплины	Объем часов	Тема практического занятия	Учебно- наглядные пособия
1	1	2	Особенности подготовки бакалавров по направлению 09.03.04 – «Программная инженерия» и ее соответствие профессиональным стандартам в области IT-индустрии	Раздаточный материал
2		2	Практика использования моделей жизненного цикла для разработки программных систем	Раздаточный материал
3		2	Обработка требований при разработке программных систем	Раздаточный материал
4		2	Значимость и методы проектирования ПО	Раздаточный материал
5		2	Методы системного, теоретического программирования	Раздаточный материал
6	2	2	Области применения инженерии повторно используемого компонента. Подходы к тестированию программ	Раздаточный материал
7		2	Модель качества ПО. Модели оценки и надежности	Раздаточный материал
8	3	2	Теория управления в программной инженерии	Раздаточный материал
9		2	Инструменты создания программных систем	Раздаточный материал
Итого:		18		

Лабораторные работы

Не предусмотрены учебным планом

Самостоятельная работа студента

Раздел дисциплины	№ п/п	Тема и вид СРС	Трудоемкость (в часах)
1	1	Характеристика стандартов разработки автоматизированных систем (АС). Стандарты разработки документации. <i>Работа с литературой</i>	2
	2	Стандарты программной инженерии. <i>Работа с литературой</i>	2
	3	Жизненный цикл компонентной разработки ПС. <i>Конспектирование</i>	2

2	4	Функциональные и нефункциональные, пользовательские и системные требования. <i>Работа с литературой</i>	2
	5	Документирование, формирование и анализ требований. Аттестация требований. Управление требованиями. <i>Работа с литературой</i>	2
	6	Модели системного окружения. Поведенческие модели. Модели данных. Объектные модели. Инструментальные CASE-средства. <i>Работа с литературой</i>	2
	7	Технологии быстрого прототипирования. Прототипирование пользовательских интерфейсов. Спецификация интерфейсов. Спецификации поведения систем. <i>Подготовка сообщения</i>	2
	8	Архитектурное проектирование. Структурирование системы. Модели управления. Модульная декомпозиция. <i>Подготовка сообщения</i>	2
	9	Процессы системного проектирования программных средств. Структурное проектирование сложных программных средств. <i>Подготовка сообщения</i>	2
	10	Проектирование программных модулей и компонентов. Проектирование и оценивание интерфейса пользователя. <i>Работа с литературой</i>	2
	11	Принципы верификации и аттестации программ. Инспектирование программных систем. <i>Конспектирование</i>	2
	12	Тестирование дефектов, сборки и интерфейсов. Процессы и средства тестирования программных компонентов. Технологические этапы и стратегии систематического тестирования программ. <i>Работа с литературой</i>	2
3	13	Наследуемые системы. Сопровождение программного обеспечения. Эволюция системной архитектуры. Реинжиниринг программного обеспечения. Управление конфигурациями, изменениями, версиями и выпусками. <i>Работа с литературой</i>	2
	14	Функциональная надежность. Работоспособность и безотказность. Безопасность и защищенность. Критические программные системы. Особенности дефектов, ошибок и рисков в программном обеспечении. <i>Работа с литературой</i>	2
	15	Управление разработкой ПО: управление персоналом, групповая работа, подбор и удержание персонала, модели оценки уровня развития персонала. <i>Подготовка сообщения</i>	2
	16	Оценка стоимости программного проекта. Управление качеством: стандарты, планирование, контроль, измерение. Совершенство производства ПО. Анализ и моделирование производства. <i>Работа с литературой</i>	2

	17	Качество ПО: функциональная надежность, работоспособность и безотказность, безопасность и защищенность. <i>Конспектирование</i>	2
	18	Критические программные системы. Особенности дефектов, ошибок и рисков в программном обеспечении. <i>Работа с литературой</i>	2
Итого:			36

5. Примерная тематика курсовых проектов (работ)

Курсовые проекты (работы) не предусмотрены учебным планом.

6. Образовательные технологии

Семестр	Вид занятия (Л, ПР, ЛР)	Используемые интерактивные образовательные технологии	Количество часов
2	Л	интерактивные лекции с элементами мозгового штурма, дискуссии, проблемные лекции	10
	ПР	кейс-анализ, экспресс-опросы, проектная работа	6
Итого:			16

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

Для оценки качества усвоения курса используются следующие формы контроля:

- **текущий** – контроль выполнения практических заданий;
- **рубежный** – коллоквиумы, контрольные работы по разделам;
- **итоговый** осуществляется посредством тестирования, зачета.

Контроль самостоятельной работы студентов осуществляется с помощью ответов на практических занятиях, коллоквиумах, консультациях, ответов на тестирование.

Перечень вопросов для проведения экспресс-опросов

Раздел 1:

1. Назовите цели и задачи программной инженерии.
2. Назовите признаки зрелой профессии, присущие программной инженерии.
3. Назовите области знаний SWEBOK инженерии разработки ПО.
4. Приведите базовые понятия SWEBOK.
5. Определите цели и задачи области инженерии – управление проектом.
6. Определите цели и задачи области инженерии – управление качеством.
7. Дайте определение жизненного цикла разработки программного обеспечения.
8. Назовите три основные группы процессов жизненного цикла и перечислите процессы каждой из групп.
9. Назовите дополнительные процессы ЖЦ и перечислите их.
10. Дайте характеристику организационных процессов ЖЦ.
11. Какой международный стандарт определяет перечень и содержание процессов ЖЦ программного продукта?

12. Все ли процессы, указанные в стандарте, должны быть выполнены при каждой разработке программного обеспечения. Дает ли стандарт такие возможности, которые могут быть актуальными для конкретного случая?

13. Какие разделы ядра знаний и стандарта наиболее необходимы при разработке программных систем.

14. Охарактеризуйте понятие модели ЖЦ и назовите их виды.

15. Дайте характеристику каскадной модели.

16. Определите отличительную особенность спиральной модели ЖЦ.

17. Какие общие черты имеют инкрементная и эволюционная модели?

18. Дайте перечень процессов ЖЦ стандарта и назовите их назначение.

19. Как построить новую модель ЖЦ на основе стандарта?

20. Дайте классификацию процессов ЖЦ стандарта.

21. Назовите процессы управления проектом.

22. Назовите процессы управления качеством.

23. Проведите сравнительную оценку модели процессов ЖЦ стандарта 12207 и областей – процессов ядра знаний SWEBOK.

24. Как называется этап ЖЦ разработки ПО, на котором фиксируется контракт между заказчиком и исполнителем разработки?

25. Назовите действующих лиц процесса формирования требований.

26. Назовите источники сведений о требованиях.

27. Какова последовательность шагов по использованию действующей системы в новой разработке?

28. Назовите категории классификации требований.

29. Цели и составляющие концептуального моделирования проблемы.

30. Что определяет онтология концептуального моделирования проблемы?

31. Объясните суть отношений, с помощью которых строятся понятия: обобщение, декомпозиция, абстракция, ассоциация.

32. Назовите элементы объектно-ориентированного моделирования программных систем.

33. В чем состоит принцип сокрытия информации?

34. Определите концепция модели сценариев для сбора требований.

35. Дайте пояснения для нотации диаграммы сценариев и базовых отношений в них.

36. Назовите основные типы объекты модели.

37. Приведите задачи трассировки требований.

38. Расскажите о принципах взаимоотношений между заказчиком и разработчиком требований к системе.

39. Определите задачи анализа предметной области и процессов проектирования архитектуры системы.

40. Сформулируйте задачи концептуального проектирования моделей ПрО.

41. Назовите продукты анализа домена в методе Шлаер и Меллора.

42. Назовите модели метода Шлаер и Меллора и их суть.

43. Какие еще модели ПрО Вы знаете?

44. Перечислите ключевые факторы, влияющие на проектирование интерфейсов.

45. Назовите примеры нефункциональных требований, которые требуется учитывать на стадии проектирования архитектуры.

46. Какие уровни выделяются в архитектуре системы?

47. Какие известны способы объединения объектов в подсистемы?

48. Назовите приемы обеспечения переноса системы в другую среду.

Раздел 2

1. Дайте характеристику структурного метода.

2. Приведите основные особенности и возможности объектно-ориентированного программирования.
3. Какие структуры имеются в языке UML для наглядного проектирования?
4. Приведите пути развития компонентного программирования.
5. Приведите базовые определения в компонентном программировании.
6. Определите основные понятия и этапы жизненного цикла компонентного программирования.
7. Определите основные элементы аспектно-ориентированного программирования.
8. Дайте характеристику инженерии ПрО.
9. Объекты генерирующего программирования и краткая их характеристика.
10. Представьте главные теоретические методы программирования.
11. Дайте определение формальных методов программирования.
12. Назовите формальные методы проверки правильности программ.
13. Какие процессы проверки зафиксированы в стандарте?
14. Какие объекты входят в доказательство правильности программ?
15. Назовите основные методы доказательства корректности программ и базис этих методов.
16. Определите типы логических операций, используемых при логическом доказательстве корректности программ.
17. В чем состоит отличие техники формального доказательства от символьного выполнения программ?
18. Сформулируйте основные задачи верификации и валидации программ.
19. В чем отличие верификации и валидации?
20. Определите процесс тестирования.
21. Назовите методы тестирования.
22. Объясните значения терминов «черный ящик», «белый ящик».
23. Назовите объекты тестирования и подходы к их тестированию.
24. Какая существует классификация типов ошибок в программах?
25. Определите основные этапы ЖЦ тестирования ПО.
26. Наведите классификацию тестов для проверки ПО.
27. Какие задачи выполняет группа тестирующих?
28. Какая организация работ проводится для проведения тестирования.
- Определите цели и задачи метода интеграции в программной инженерии.
29. Назовите системы, которые поддерживают процессы интеграции и преобразования данных.
30. Охарактеризуйте кратко современные системы взаимосвязи объектов – COM, CORBA, JAVA и др.
31. Назовите методы вызова компонентов в распределенных средах.
32. Какую роль выполняет брокер объектных запросов?
33. Определите проблему преобразования данных в ЯП.
34. Какие требуется провести преобразования передаваемых по сети данных от объекта в JAVA к объекту в C++ и обратно?
35. Определите проблемы преобразования данных, связанные с заменой одной БД на другую.
36. Какие методы переноса данных существуют?
37. Определите цели и задачи изменения ПС при проведении сопровождения.
38. Какие выполняются работы при сопровождении, когда вносятся изменения?
39. Дайте краткую характеристику проблем, возникающих при сопровождении системы.
40. Определите основные задачи реинженерии ПО.
41. Определите основные операции рефакторинга компонентов.
42. Определите основные операции реинженерии программных систем.

43. Определите понятие – качество ПО.
44. Назовите основные аспекты и уровни модели качества ПО.
45. Определите характеристики качества ПО и их назначение.
46. Какие методы используются при определении показателей качества?
47. Определите метрики программного продукта и их составляющие.
48. Какие стандарты в области качества ПО существуют?
49. Назовите основные цели и задачи системы управления качеством.
50. В чем суть инженерии качества?
51. Назовите содержание классификации моделей надежности.
52. Определите типы моделей надежности и их базис.
53. Какие данные необходимы для оценивания надежности ПО?

Раздел 3:

1. Как решаются задачи менеджмента программного проекта?
2. Определите процесс планирования менеджмента проекта.
3. Определите понятие управления риском.
4. Объясните стратегию оценки стоимости продукта по Бозму.
5. Как решаются задачи менеджмента программного проекта?
6. Определите процесс планирования менеджмента проекта.
7. Определите понятие управления риском.
8. Объясните стратегию оценки стоимости продукта по Бозму.
9. Что понимается под процессом управления конфигурацией ПО?
10. Приведите основные задачи управления конфигурацией.
11. Дайте общую характеристику понятий идентификации, учета статуса.
12. Какие действия выполняются в процессе управления версиями ПО?
13. Сформулируйте основные задачи учета и аудита.

Дайте характеристику спецификации компонента.

14. Определите языковые средства описания компонентов.
15. Представьте объекты языка JAVA.
16. Определите методы интеграции объектов языка JAVA.
17. Определите основные характеристики объектов в системе CORBA.
18. Приведите структуру описания спецификации интерфейса в языке IDL.
19. Расскажите об особенностях описания объектов в системе COM.
20. Проведите сравнение средств CORBA и COM.
21. Для каких целей создано Rational Rose.
22. Назовите инструменты Rational Rose.
23. Дайте перечень диаграмм языка моделирования UML.
24. Определите процесс разработки ПС с помощью UML.
25. Для каких целей разработан метод MSF?
26. Назовите основные модели MSF.
27. Как решаются вопросы управления проектом и рисками в системе MSF.
28. Цели и задачи проектной группы в MSF.

Тематика рефератов

1. Лауреаты премии Тьюринга (по годам).
2. Эволюция сложных программных систем.
3. Методы документирования архитектуры.
4. Управление знаниями в процессе разработки программных систем.
5. CASE технологии разработки программных систем.
6. Модели программных систем.
7. Построение процесса разработки программных систем.

8. Бизнес аспекты разработки программных систем.
9. Модели ROI для оценки эффективности компаний-разработчиков ПО.
10. Человеческий фактор в разработке программного обеспечения.
11. Модели и методы оценки личностных характеристик исполнителей и команды в целом.
12. Оценка затрат программных проектов методом функциональных точек.
13. Регрессионная модель оценки затрат программных проектов COSIMO II.
14. Оценка программных проектов в модели SLIM.
15. Количественные методики оценки рисков программных проектов.
16. Метрические показатели в оценке программных проектов.
17. Модели структурного анализа программных проектов.
18. Модели объектно-ориентированного анализа программных проектов.
19. Метод определения точек тестирования, основанный на анализе цикломатической сложности Мак-Кейба.
20. Сравнительный анализ инструментов моделирования и трассировки программных требований.
21. Сравнительный анализ инструментов верификации программных проектов.
22. Сравнительный анализ инструментов оптимизации программных проектов.
23. Сравнительный анализ инструментов тестирования ПО (генераторы тестов, схемы выполнения тестов, оценка тестов, управление тестами).
24. Сравнительный анализ инструментов сопровождения ПО.
25. Системы моделирования процессов разработки ПО.
26. Среды разработки ПО, ориентированные на процессы.
27. Сравнительный анализ инструментов обеспечения качества ПО.
28. Сравнительный анализ инструментов управления конфигурацией ПО.
29. Инструменты планирования и отслеживания программных проектов.
30. Инструменты, реализующие поддержку инфраструктуры разработки.
31. Методы и технологии балансировки нагрузки.
32. Технологии Data Mining.
33. Технологии защиты авторских прав.
34. Проблемы исчерпания IPv4-адресов и особенности перехода на IPv6.
35. Парадигма SAAS.
36. Утечка исходных кодов коммерческого ПО: примеры и последствия.
37. Метод верификации ПО "Model checking".
38. командные методологии управления разработкой проектов.
39. Методы деперсонализации обрабатываемых данных для обеспечения их защиты.
40. Библиотеки информационных ресурсов по программной инженерии.
41. Проблемы ИТ аутсорсинга.
42. Автоматное программирование.
43. Краудсорсинг.
44. Социальные последствия отказов информационных систем.
45. Перевод/написание статьи на Википедии "Компьютинг".
46. Разработка приложений под iPhone и iPad.
47. Технологии эмуляции и виртуализации.
48. Использование параллелизма в программах.
49. Инновационные и патентноспособные технологии будущего.
50. Усовершенствование проактивного метода обнаружения вредоносного ПО.
51. Безопасность и защита личных данных в электронном обучении.
52. Вопросы безопасности в системах для удаленного проведения экзаменов.

53. Метрики ПО.
54. Интернет-протоколы: принципы работы и области применения.
55. Обфускация.
56. ПО в системах "умный дом".
57. Моделирование рисков разработки ПО.
58. Semantic web.
59. Частное облако.
60. Парадигмы программирования.
61. Концепция жизни в окружении цифровых устройств.
62. Кодекс этики разработчиков ПО.
63. Облачные технологии.
64. Автоматизация тестирования.
65. Беспроводные децентрализованные самоорганизующиеся сети мобильных устройств.
66. Разработка и эксплуатация безопасных систем электронного обучения.
67. Управление авторским правом на цифровой контент в электронном обучении.
68. "Псевдо-анонимность" и "несвязуемость".
69. Разработка платформонезависимых программ.
70. Методы и технологии профилирования пользователей.
71. Интернет будущего.
72. Модели ценообразования на рынке программных продуктов.
73. Программирование роботов на Microsoft Robotics Studio.
74. Обеспечение доверия, конфиденциальности и профессиональной этики в Web 2.0.

**Образец теста для проведения итогового контроля
по итогам освоения дисциплины, а также для контроля самостоятельной работы**

Указания: Напишите Вашу фамилию, номер группы и дату. Для ответа на вопрос с выбором варианта ответа достаточно написать номер вопроса и рядом букву, обозначающую правильный вариант из предложенных в тексте ответов на вопрос. Если Вы считаете правильными несколько вариантов ответов, то запишите через запятую соответствующие литеры букв.

Время: 45 мин

1. Укажите объекты профессиональной деятельности инженеров-программистов:
 - a) проект разработки программного продукта;
 - b) процессы организационной жизни общества;
 - c) создаваемое ПО;
 - d) элементы общественной культуры;
 - e) процессы жизненного цикла программного продукта;
 - f) структурные подразделения предприятий;
 - g) методы и инструменты разработки программного продукта.
2. Содействие заказчику в оценке и выборе вариантов ПО является задачей _____ деятельности.
 - a) проектной;
 - b) аналитической;
 - c) организационно-управленческой;
 - d) педагогической;
 - e) технологической;
 - f) научно-исследовательской;

- g) сервисно-эксплуатационной;
 - h) производственной.
3. Алгоритмизация является задачей такой сферы программной инженерии как:
- a) разработка ПО;
 - b) проектирование ПО;
 - c) сопровождение;
 - d) процесс разработки ПО;
 - e) документирование;
 - f) качество ПО;
 - g) анализ и планирование.
4. Спиральная модель относится к _____ стратегиям конструирования ПО.
- a) водопадным (однократным);
 - b) инкрементным;
 - c) эволюционным;
 - d) смешанным.
5. Повышение качества программных систем можно обеспечить с помощью:
- a) верификации;
 - b) рефакторинга;
 - c) интегрирования компонентов.
6. Укажите те варианты, которые влияют на успешность программного проекта:
- a) решаемая задача;
 - b) стандарты документирования;
 - c) заказчик;
 - d) команда разработки;
 - e) количество предполагаемых пользователей;
 - f) инфраструктура;
 - g) методология проектирования ПО.
7. Укажите методологию, в которой не предусмотрены промежуточные версии ПО:
- a) RUP;
 - b) RAD;
 - c) водопадная;
 - d) экстремальное программирование;
 - e) макетирование.
8. Внесение изменений в водопадной модели проектирования ПО возможно на этапе:
- a) проектирование;
 - b) разработка;
 - c) анализ и планирование;
 - d) эксплуатация и сопровождение;
 - e) тестирование.
9. Укажите методологию проектирования ПО среди водопадных, в которой возможен возврат только к предыдущему этапу:
- a) классическая итерационная;
 - b) каскадная;
 - c) строгая каскадная.

10. Укажите правильный русский аналог аббревиатуры RAD как методологии проектирования ПО:
- быстрая разработка приложений;
 - унифицированный процесс проектирования;
 - экстремальное программирование.
11. К артефактам стадии конструирования методологии RUP относятся:
- спецификация основных проектных требований;
 - начальный словарь проекта;
 - дополнительные требования;
 - начальный план развития;
 - описание текущей реализации;
 - начальная модель Use Case;
 - описание программной архитектуры;
 - начальная оценка риска;
 - альфа-, бета-версии программного продукта;
 - действующий архитектурный макет;
 - проектный план с этапами и итерациями;
 - основная модель Use Case;
 - план разработки всего проекта;
 - руководство пользователя;
 - переработанный список элементов риска.
12. Гибкие технологии разработки программного обеспечения ориентированы на _____ команды разработчиков.
- большие;
 - небольшие.
13. Укажите, что важнее в гибких технологиях проектирования ПО:
- сотрудничество с заказчиком;
 - согласование условий контракта.
14. Укажите agile технологии разработки ПО:
- водопадная;
 - спиральная;
 - Scrum;
 - RUP;
 - макетирование;
 - экстремальное программирование;
 - инкрементная.
15. Укажите, в каком методе XP выполняется изменение программы с целью упрощения добавления новой возможности:
- игра в планирование;
 - метафора;
 - простой дизайн;
 - рефакторинг;
 - парное программирование;
 - коллективное владение.
16. Укажите, какое из указанных свойств требований указывает на то, что должна быть возможность их изменения:

- a) корректность;
- b) однозначность;
- c) приоритизация;
- d) модифицируемость;
- e) полнота;
- f) непротиворечивость;
- g) проверяемость;
- h) отслеживаемость.

17. Укажите способы выявления требований:

- a) исследования;
- b) моделирование;
- c) интервью;
- d) семинары;
- e) проектирование;
- f) создание прототипов;
- g) создание вариантов использования Use Case.

18. Наблюдение за проектом осуществляется с помощью срезов по:

- a) задачам;
- b) требованиям;
- c) прототипам;
- d) сотрудникам;
- e) дефектам;
- f) рискам;
- g) критическому пути.

19. Неправильные версии исходных файлов относятся к проблемам с:

- a) исходными текстами;
- b) подключаемыми файлами;
- c) используемыми библиотеками;
- d) процедурами сборки;
- e) средствами сборки;
- f) системной средой и аппаратной платформой сборки.

20. Укажите характеристику качества ПО, подхарактеристиками которого являются пригодность, корректность, способность к взаимодействию, согласованность и защищенность.

- a) функциональность;
- b) надежность;
- c) практичность;
- d) эффективность;
- e) сопровождаемость;
- f) мобильность.

Бланк ответов к тестовым заданиям

Номер задания	Ответ	Номер задания	Ответ
1	a,c,e,g	11	e, i, n
2	b	12	b
3	b	13	a
4	c	14	c, f

5	b	15	d
6	a, c, d, f, g	16	d
7	c	17	a, c, d, f, g
8	d	18	a, d, e, g
9	c	19	a
10	a	20	a

При тестировании все верные ответы берутся за 100%, тогда оценка выставляется в соответствии с таблицей:

Процент выполнения задания	Оценка
85% и более	5 (отлично)
70-84%	4 (хорошо)
50-69%	3 (удовлетворительно)
менее 50%	2 (неудовлетворительно)

Примерные вопросы к зачету

1. Жизненный цикл программных систем.
2. Системные основы современных технологий программной инженерии.
3. Модель профиля стандартов жизненного цикла программных систем.
4. Управление программными проектами в системе – СММІ.
5. Стандарты административного управления качеством программных систем.
6. Стандарты открытых систем, регламентирующие структуру и интерфейсы программного обеспечения.
7. Процессы системного проектирования программного обеспечения.
8. Структурное проектирование программных систем.
9. Проектирование программных модулей и компонентов.
10. Технико-экономическое обоснование программных проектов.
11. Разработка требований к программным системам.
12. Структура документов, отражающих требования к программным системам.
13. Планирование жизненного цикла программных систем.
14. Планирование процессов управления качеством программных систем.
15. Объектно-ориентированное проектирование программных систем.
16. Ресурсы для обеспечения жизненного цикла программных систем.
17. Причины и свойства дефектов, ошибок и модификаций в программных системах.
18. Риски в жизненном цикле программных систем.
19. Риски при формировании требований к характеристикам программных систем.
20. Факторы, определяющие качество программных систем.
21. Свойства и атрибуты качества функциональных возможностей программных систем.
22. Принципы верификации и тестирования программ.
23. Процессы и средства тестирования программных компонентов.
24. Технологические этапы и стратегии систематического тестирования программ.
25. Процессы тестирования структуры программных компонентов.
26. Организация и методы сопровождения программных систем.
27. Задачи и процессы переноса программ и данных на иные платформы.
28. Процессы управления конфигурацией программных систем.
29. Технологическое обеспечение при сопровождении и управлении конфигурацией программных систем.
30. Организация документирования программных систем.
31. Формирование требований к документации программных систем.
32. Планирование документирования проектов программных систем.
33. Процессы сертификации в жизненном цикле программных продуктов.

34. Организация сертификации программных продуктов.
35. Документирование процессов и результатов сертификации программных продуктов.

8. Учебно-методическое и информационное обеспечение дисциплины

8.1. Основная литература

1. Гвоздева В.А. Введение в специальность программиста: учебник. – 2-е изд., испр. и доп. – М.: ИД «ФОРУМ»: ИНФРА-М, 2013. – 208 с.
2. Введение в инфокоммуникационные технологии: учебное пособие / Л.Г. Гагарина, А.М. Баин, Г.А. Кузнецов, Е.М. Портнов, Я.О. Теплова. – М.: ИД «ФОРУМ»: ИНФРА-М, 2013. – 336 с.
3. Гецци К., Джазайери М., Мандртоли Д. Основы инженерии программного обеспечения. – 2-е издание. – СПб.: БХВ-Петербург, 2005. – 832 с.
4. Карпенко С.Н. Введение в программную инженерию. Учебно-методические материалы по программе повышения квалификации «Информационные технологии и компьютерное моделирование в прикладной математике». – Нижний Новгород, 2007.
5. Лаврищева Е.М., Петрухин В.А. Методы и средства инженерии программного обеспечения: учебник. – Москва, 2006.
6. Липаев В.В. Программная инженерия. Методологические основы. Учебник. М.: ТЕИС, 2006. – 608 с.

8.2. Дополнительная литература

1. Батоврин В.К. Толковый словарь по системной и программной инженерии. – М.: ДМК Пресс, 2012. – 280 с.
2. Вигерс К.И. Разработка требований к ПО. – М., 2004.
3. Фатрелл Р.Т., Шафер Д.Ф., Шафер Л.И. Управление программными проектами. Достижение оптимального качества при минимуме затрат. – М.: Издательский дом «Вильямс», 2004. – 1136 с.
4. Чернецки К., Айзенекер У. Порождающее программирование. Методы, инструменты, применение. СПб.: Издательский дом «Питер», 2005.
5. Эммерих В. Конструирование распределенных объектов. Методы и средства программирования интероперабельных объектов в архитектурах OMG/CORBA, Microsoft COM и Java RMI. – М.: Мир, 2002.
6. Единая система программной документации. М.: ИПК Издательство стандартов, 2001.

8.3. Программное обеспечение и Интернет-ресурсы

Программное обеспечение:

1. Пакет Microsoft Office – офисное приложение.
2. Браузеры Интернет.
3. Инструментальные среды разработки программ.

Интернет-ресурсы:

1. Интуит. Национальный открытый университет [Электронный ресурс]. – Режим доступа: <http://www.intuit.ru>.
2. Образовательный портал ПГУ им. Т.Г. Шевченко [Электронный ресурс]. – Режим доступа: <http://moodle.spsu.ru>.

8.4. Методические указания и материалы по видам занятий

Методические указания по выполнению индивидуальных заданий предоставляются студентам в виде теоретических предпосылок (в электронном и печатном виде) к практическим работам.

9. Материально-техническое обеспечение дисциплины

Для осуществления образовательного процесса по дисциплине «Введение в специальность» необходима лекционная аудитория, оборудованная мультимедийными средствами для проведения

лекций-визуализаций. Для проведения практических занятий используется компьютерная аудитория, оснащенная 10 компьютерами, объединенными локальной сетью, с доступом в Интернет.

10. Методические рекомендации по организации изучения дисциплины

Рабочая учебная программа по дисциплине «Введение в специальность» составлена в соответствии с требованиями Федерального Государственного образовательного стандарта ВО по направлению 09.03.04 – «Программная инженерия» и учебного плана по профилю подготовки «Разработка программно-информационных систем».

Изучение дисциплины «Введение в специальность» включает лекционные и практические занятия. Во время выполнения практических заданий в учебной аудитории студент может консультироваться с преподавателем, определять наиболее эффективные методы решения поставленных задач. Если какая-то часть задания остается не выполненной, студент может продолжить ее выполнение во время внеаудиторной самостоятельной работы.

Работа с информационными источниками считается одним из основных видов самостоятельной работы. Для приобщения обучаемых к поиску, к исследовательской работе, для развития их творческого потенциала следует по возможности избегать прямого руководства работой обучающихся при выполнении ими тех или иных заданий, чаще выступать в роли консультанта, эксперта, коллеги-исследователя.

Текущий контроль усвоения знаний по дисциплине предполагает использование разных форм контроля, в том числе проверку практических заданий. Итоговый контроль может осуществляться в форме зачета, теста. Вопросы к зачету и образец тестовых заданий приведены. Выполнение практических и модульных контрольных работ является необходимым условием для сдачи зачета.

11. Технологическая карта дисциплины

Курс: I группа: РФ17ДР62ПИ семестр: II

Преподаватель-лектор: Гарбузняк Елена Сергеевна

Преподаватель, ведущий практические занятия: Гарбузняк Елена Сергеевна

Кафедра информатики и программной инженерии

Наименование дисциплины / курса	Уровень//ступень образования (бакалавриат, специалитет, магистратура)	Статус дисциплины в рабочем учебном плане (А, Б, В, Г)	Количество зачетных единиц / кредитов	
Введение в специальность	бакалавриат	В	2	
Смежные дисциплины по учебному плану:				
«Теоретическая информатика», «Основы программирования»				
ВВОДНЫЙ МОДУЛЬ				
(входной рейтинг-контроль, проверка «остаточных» знаний по смежным дисциплинам)				
Тема, задание или мероприятие входного контроля	Виды текущей аттестации	Аудиторная или внеаудиторная	Минимальное количество баллов	Максимальное количество баллов
Итого:				
БАЗОВЫЙ МОДУЛЬ				
(проверка знаний и умений по дисциплине)				
Тема, задание или мероприятие текущего контроля	Виды текущей аттестации	Аудиторная или внеаудиторная	Минимальное количество баллов	Максимальное количество баллов
Итого:				
ДОПОЛНИТЕЛЬНЫЙ МОДУЛЬ				

18. *Математическое моделирование по организационным вопросам*

[Signature]

Etas-

Amesbury

П.А. Тягульская, доцент