

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Комсомольский-на-Амуре государственный университет»

Колледж

УТВЕРЖДАЮ
Проректор по УВР и ОВ
Т.Е. Наливайко
_____ 2021 г.



ПРОГРАММА КВАЛИФИКАЦИОННОГО ЭКЗАМЕНА

по профессиональному модулю ПМ.3
«Участие в интеграции программных модулей»

по специальности среднего профессионального образования
09.02.03 – «Программирование в компьютерных системах»
(базовая подготовка)

на базе основного общего образования

Форма обучения

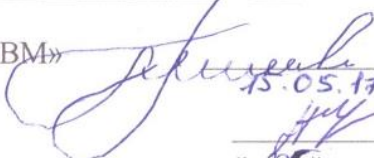
очная

Комсомольск-на-Амуре, 2021

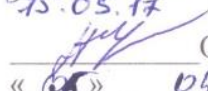
Программа квалификационного экзамена по ПМ 03 «Участие в интеграции программных модулей» разработан на основе Федерального государственного образовательного стандарта по специальности среднего профессионального образования 09.02.03 - «Программирование в компьютерных системах», утвержденного приказом Министерства образования и науки Российской Федерации от 28.07.2014 № 804.

Комплект контрольно-оценочных средств рассмотрен и утвержден на заседании кафедры «МОП ЭВМ»

Протокол № 13 от «15» 05 2017г.

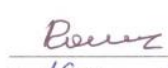
Заведующий кафедрой «МОП ЭВМ»  В.А. Тихомиров

Разработчик КОС

 О.Н. Попенко
«05» 05 2017 г.

СОГЛАСОВАНО

Декан факультета довузовской подготовки

 И.В. Конырева
«19» 05 2017г.

Начальник учебно-методического управления

 Е.Е. Поздеева
«29» 05 2017г.

Рецензент

руководитель проекта по реализации ФЦП, ПАО «Амурский судостроительный завод»



 В.А. Ханов
«11» 05 2017г.

СОДЕРЖАНИЕ

1	Паспорт комплекта контрольно-оценочных средств	4
2	Результаты освоения профессионального модуля, подлежащие проверке	5
3	Оценка освоения профессионального модуля:	14
	3.1 Формы и методы оценивания	14
	3.2 Типовые задания для оценки освоения профессионального модуля	16
4	Контрольно-оценочные материалы для итоговой аттестации по профессионального модулю	37
5	Критерии и шкалы для интегрированной оценки уровня сформированности компетенций.....	48
6	Список рекомендуемой литературы и источников информации	49

1 Паспорт комплекта контрольно-оценочных средств

В результате освоения ПМ «Участие в интеграции программных модулей» обучающийся должен обладать предусмотренными ФГОС по специальности СПО 09.02.03 - «Программирование в компьютерных системах» следующими знаниями, умениями, которые формируют общие и профессиональные компетенции.

Общие компетенции:

ОК 1 - Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.

ОК 2 - Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.

ОК 3 - Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.

ОК 4 - Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.

ОК 5 - Использовать информационно-коммуникационные технологии в профессиональной деятельности.

ОК 6 - Работать в коллективе и команде, эффективно общаться с коллегами, руководством, потребителями.

ОК 7 - Брать на себя ответственность за работу членов команды (подчиненных), результат выполнения заданий.

ОК 8 - Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.

ОК 9 - Ориентироваться в условиях частой смены технологий в профессиональной деятельности.

Профессиональные компетенции:

ПК 3.1 – анализировать проектную и техническую документацию на уровне взаимодействия компонент программного обеспечения;

ПК 3.2 – выполнять интеграцию модулей в программную систему;

ПК 3.3 – выполнять отладку программного продукта с использованием специализированных программных средств;

ПК 3.4 – осуществлять разработку тестовых наборов и тестовых сценариев;

ПК 3.5 – производить инспектирование компонент программного продукта на предмет соответствия стандартам кодирования;

ПК 3.6 – разрабатывать технологическую документацию.

Формой промежуточной аттестации по модулю является экзамен.

2 Результаты освоения профессионального модуля, подлежащие проверке

В результате текущей аттестации по модулю осуществляется комплексная проверка следующих умений (У) и знаний (З), а также динамика формирования общих (ОК) и профессиональных (ПК) компетенций.

Таблица 1

Результаты обучения	Показатели оценки результата	Формы контроля
ОК 1 - Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.	Знать: З1 – модели процесса разработки программного обеспечения; З2 – основные принципы процесса разработки программного обеспечения; З3 – основные подходы к интегрированию программных модулей; З4 – основные методы и средства эффективной разработки; З5 – основы верификации и аттестации программного обеспечения; З6 – концепции и реализации программных процессов; З7 – принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения; З8 – методы организации работы в коллективах разработчиков программного обеспечения; З9 – основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерения характеристик и параметров программ, программных систем и комплексов; З10 – стандарты качества программного обеспечения; З11 – методы и средства разработки программной документации.	Устный опрос, самостоятельная работа, лабораторная работа, экзамен
	Уметь: У1 – владеть основными методологиями процессов разработки программного обеспечения; У2 – использовать методы для получения кода с заданной функциональностью и степенью качества.	Самостоятельная работа, лабораторная работа, экзамен
ОК 2 - Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.	Знать: З1 – модели процесса разработки программного обеспечения; З2 – основные принципы процесса разработки программного обеспечения; З3 – основные подходы к интегрированию программных модулей; З4 – основные методы и средства эффективной разработки; З5 – основы верификации и аттестации программного обеспечения; З6 – концепции и реализации программных процессов;	Устный опрос, самостоятельная работа, лабораторная работа, экзамен

	37 – принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения; 38 – методы организации работы в коллективах разработчиков программного обеспечения; 39 – основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерения характеристик и параметров программ, программных систем и комплексов; 310 – стандарты качества программного обеспечения; 311 – методы и средства разработки программной документации.	
	Уметь: У1 – владеть основными методологиями процессов разработки программного обеспечения; У2 – использовать методы для получения кода с заданной функциональностью и степенью качества.	Самостоятельная работа, лабораторная работа, экзамен
ОК 3 - Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.	Знать: 31 – модели процесса разработки программного обеспечения; 32 – основные принципы процесса разработки программного обеспечения; 33 – основные подходы к интегрированию программных модулей; 34 – основные методы и средства эффективной разработки; 35 – основы верификации и аттестации программного обеспечения; 36 – концепции и реализации программных процессов; 37 – принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения; 38 – методы организации работы в коллективах разработчиков программного обеспечения; 39 – основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерения характеристик и параметров программ, программных систем и комплексов; 310 – стандарты качества программного обеспечения; 311 – методы и средства разработки программной документации.	Устный опрос, самостоятельная работа, лабораторная работа, экзамен
	Уметь: У1 – владеть основными методологиями процессов разработки программного обеспечения; У2 – использовать методы для получения кода с заданной функциональностью и степенью качества.	Самостоятельная работа, лабораторная работа, экзамен
ОК 4 - Осуществлять поиск и использование информации, необходимой	Знать: 31 – модели процесса разработки программного обеспечения; 32 – основные принципы процесса разработки про-	Устный опрос, самостоятельная работа, лабораторная ра-

димой для эффективного выполнения профессиональных задач, профессионального и личностного развития.	граммного обеспечения; 33 – основные подходы к интегрированию программных модулей; 34 – основные методы и средства эффективной разработки; 35 – основы верификации и аттестации программного обеспечения; 36 – концепции и реализации программных процессов; 37 – принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения; 38 – методы организации работы в коллективах разработчиков программного обеспечения; 39 – основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерения характеристик и параметров программ, программных систем и комплексов; 310 – стандарты качества программного обеспечения; 311 – методы и средства разработки программной документации.	бота, экзамен
	Уметь: У1 – владеть основными методологиями процессов разработки программного обеспечения; У2 – использовать методы для получения кода с заданной функциональностью и степенью качества.	Самостоятельная работа, лабораторная работа, экзамен
ОК 5 - Использовать информационно-коммуникационные технологии в профессиональной деятельности.	Знать: 31 – модели процесса разработки программного обеспечения; 32 – основные принципы процесса разработки программного обеспечения; 33 – основные подходы к интегрированию программных модулей; 34 – основные методы и средства эффективной разработки; 35 – основы верификации и аттестации программного обеспечения; 36 – концепции и реализации программных процессов; 37 – принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения; 38 – методы организации работы в коллективах разработчиков программного обеспечения; 39 – основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерения характеристик и параметров программ, программных систем и комплексов; 310 – стандарты качества программного обеспечения; 311 – методы и средства разработки программной документации.	Устный опрос, самостоятельная работа, лабораторная работа, экзамен

	Уметь: У1 – владеть основными методологиями процессов разработки программного обеспечения; У2 – использовать методы для получения кода с заданной функциональностью и степенью качества.	Самостоятельная работа, лабораторная работа, экзамен
ОК 6 - Работать в коллективе и команде, эффективно общаться с коллегами, руководством, потребителями.	Знать: 31 – модели процесса разработки программного обеспечения; 32 – основные принципы процесса разработки программного обеспечения; 33 – основные подходы к интегрированию программных модулей; 34 – основные методы и средства эффективной разработки; 35 – основы верификации и аттестации программного обеспечения; 36 – концепции и реализации программных процессов; 37 – принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения; 38 – методы организации работы в коллективах разработчиков программного обеспечения; 39 – основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерения характеристик и параметров программ, программных систем и комплексов; 310 – стандарты качества программного обеспечения; 311 – методы и средства разработки программной документации.	Устный опрос, самостоятельная работа, лабораторная работа, экзамен
	Уметь: У1 – владеть основными методологиями процессов разработки программного обеспечения; У2 – использовать методы для получения кода с заданной функциональностью и степенью качества.	Самостоятельная работа, лабораторная работа, экзамен
ОК 7 - Брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий.	Знать: 31 – модели процесса разработки программного обеспечения; 32 – основные принципы процесса разработки программного обеспечения; 33 – основные подходы к интегрированию программных модулей; 34 – основные методы и средства эффективной разработки; 35 – основы верификации и аттестации программного обеспечения; 36 – концепции и реализации программных процессов; 37 – принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения; 38 – методы организации работы в коллективах разработчиков программного обеспечения;	Устный опрос, самостоятельная работа, лабораторная работа, экзамен

	39 – основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерения характеристик и параметров программ, программных систем и комплексов; 310 – стандарты качества программного обеспечения; 311 – методы и средства разработки программной документации.	
	Уметь: У1 – владеть основными методологиями процессов разработки программного обеспечения; У2 – использовать методы для получения кода с заданной функциональностью и степенью качества.	Самостоятельная работа, лабораторная работа, экзамен
ОК 8 - Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.	Знать: 31 – модели процесса разработки программного обеспечения; 32 – основные принципы процесса разработки программного обеспечения; 33 – основные подходы к интегрированию программных модулей; 34 – основные методы и средства эффективной разработки; 35 – основы верификации и аттестации программного обеспечения; 36 – концепции и реализации программных процессов; 37 – принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения; 38 – методы организации работы в коллективах разработчиков программного обеспечения; 39 – основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерения характеристик и параметров программ, программных систем и комплексов; 310 – стандарты качества программного обеспечения; 311 – методы и средства разработки программной документации.	Устный опрос, самостоятельная работа, лабораторная работа, экзамен
	Уметь: У1 – владеть основными методологиями процессов разработки программного обеспечения; У2 – использовать методы для получения кода с заданной функциональностью и степенью качества.	Самостоятельная работа, лабораторная работа, экзамен
ОК 9 - Ориентироваться в условиях частой смены технологий в профессиональной деятельности.	Знать: 31 – модели процесса разработки программного обеспечения; 32 – основные принципы процесса разработки программного обеспечения; 33 – основные подходы к интегрированию программных модулей; 34 – основные методы и средства эффективной разработки;	Устный опрос, самостоятельная работа, лабораторная работа, экзамен

	35 – основы верификации и аттестации программно-го обеспечения; 36 – концепции и реализации программных процессов; 37 – принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения; 38 – методы организации работы в коллективах разработчиков программного обеспечения; 39 – основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерения характеристик и параметров программ, программных систем и комплексов; 310 – стандарты качества программного обеспечения; 311 – методы и средства разработки программной документации.	
	Уметь: У1 – владеть основными методологиями процессов разработки программного обеспечения; У2 – использовать методы для получения кода с заданной функциональностью и степенью качества.	Самостоятельная работа, лабораторная работа, экзамен
ПК 3.1 - Анализировать проектную и техническую документацию на уровне взаимодействия компонент программного обеспечения.	Знать: 31 – модели процесса разработки программного обеспечения; 32 – основные принципы процесса разработки программного обеспечения; 33 – основные подходы к интегрированию программных модулей; 34 – основные методы и средства эффективной разработки; 35 – основы верификации и аттестации программно-го обеспечения; 36 – концепции и реализации программных процессов; 37 – принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения; 38 – методы организации работы в коллективах разработчиков программного обеспечения; 39 – основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерения характеристик и параметров программ, программных систем и комплексов; 310 – стандарты качества программного обеспечения; 311 – методы и средства разработки программной документации.	Устный опрос, самостоятельная работа, лабораторная работа, экзамен
	Уметь: У1 – владеть основными методологиями процессов разработки программного обеспечения; У2 – использовать методы для получения кода с заданной функциональностью и степенью качества.	Самостоятельная работа, лабораторная работа, экзамен

ПК 3.2 - Выполнять интеграцию модулей в программную систему.	Знать: 31 – модели процесса разработки программного обеспечения; 32 – основные принципы процесса разработки программного обеспечения; 33 – основные подходы к интегрированию программных модулей; 34 – основные методы и средства эффективной разработки; 35 – основы верификации и аттестации программного обеспечения; 36 – концепции и реализации программных процессов; 37 – принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения; 38 – методы организации работы в коллективах разработчиков программного обеспечения; 39 – основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерения характеристик и параметров программ, программных систем и комплексов; 310 – стандарты качества программного обеспечения; 311 – методы и средства разработки программной документации. Уметь: У1 – владеть основными методологиями процессов разработки программного обеспечения; У2 – использовать методы для получения кода с заданной функциональностью и степенью качества.	Устный опрос, самостоятельная работа, лабораторная работа, экзамен Самостоятельная работа, лабораторная работа, экзамен
ПК 3.3 - Выполнять отладку программного продукта с использованием специализированных программных средств.	Знать: 31 – модели процесса разработки программного обеспечения; 32 – основные принципы процесса разработки программного обеспечения; 33 – основные подходы к интегрированию программных модулей; 34 – основные методы и средства эффективной разработки; 35 – основы верификации и аттестации программного обеспечения; 36 – концепции и реализации программных процессов; 37 – принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения; 38 – методы организации работы в коллективах разработчиков программного обеспечения; 39 – основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерения характеристик и параметров программ, программных систем и комплексов;	Устный опрос, самостоятельная работа, лабораторная работа, экзамен

	310 – стандарты качества программного обеспечения; 311 – методы и средства разработки программной документации.	
	Уметь: У1 – владеть основными методологиями процессов разработки программного обеспечения; У2 – использовать методы для получения кода с заданной функциональностью и степенью качества.	Самостоятельная работа, лабораторная работа, экзамен
ПК 3.4 - Осуществлять разработку тестовых наборов и тестовых сценариев.	Знать: 31 – модели процесса разработки программного обеспечения; 32 – основные принципы процесса разработки программного обеспечения; 33 – основные подходы к интегрированию программных модулей; 34 – основные методы и средства эффективной разработки; 35 – основы верификации и аттестации программного обеспечения; 36 – концепции и реализации программных процессов; 37 – принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения; 38 – методы организации работы в коллективах разработчиков программного обеспечения; 39 – основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерения характеристик и параметров программ, программных систем и комплексов; 310 – стандарты качества программного обеспечения; 311 – методы и средства разработки программной документации.	Устный опрос, самостоятельная работа, лабораторная работа, экзамен
	Уметь: У1 – владеть основными методологиями процессов разработки программного обеспечения; У2 – использовать методы для получения кода с заданной функциональностью и степенью качества.	Самостоятельная работа, лабораторная работа, экзамен
ПК 3.5 - Производить инспектирование компонент программного продукта на предмет соответствия стандартам кодирования.	Знать: 31 – модели процесса разработки программного обеспечения; 32 – основные принципы процесса разработки программного обеспечения; 33 – основные подходы к интегрированию программных модулей; 34 – основные методы и средства эффективной разработки; 35 – основы верификации и аттестации программного обеспечения; 36 – концепции и реализации программных процессов; 37 – принципы построения, структуры и приемы работы с инструментальными средствами, поддер-	Устный опрос, самостоятельная работа, лабораторная работа, экзамен

	живающими создание программного обеспечения; 38 – методы организации работы в коллективах разработчиков программного обеспечения; 39 – основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерения характеристик и параметров программ, программных систем и комплексов; 310 – стандарты качества программного обеспечения; 311 – методы и средства разработки программной документации.	
	Уметь: У1 – владеть основными методологиями процессов разработки программного обеспечения; У2 – использовать методы для получения кода с заданной функциональностью и степенью качества.	Самостоятельная работа, лабораторная работа, экзамен
ПК 3.6 - Разрабатывать технологическую документацию.	Знать: 31 – модели процесса разработки программного обеспечения; 32 – основные принципы процесса разработки программного обеспечения; 33 – основные подходы к интегрированию программных модулей; 34 – основные методы и средства эффективной разработки; 35 – основы верификации и аттестации программного обеспечения; 36 – концепции и реализации программных процессов; 37 – принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения; 38 – методы организации работы в коллективах разработчиков программного обеспечения; 39 – основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерения характеристик и параметров программ, программных систем и комплексов; 310 – стандарты качества программного обеспечения; 311 – методы и средства разработки программной документации.	Устный опрос, самостоятельная работа, лабораторная работа, экзамен
	Уметь: У1 – владеть основными методологиями процессов разработки программного обеспечения; У2 – использовать методы для получения кода с заданной функциональностью и степенью качества.	Самостоятельная работа, лабораторная работа, экзамен

3 Оценка освоения дисциплины

3.1 Формы и методы оценивания

Предметом оценки служат умения и знания, предусмотренные ФГОС СПО по специальности, направленные на формирование общих и профессиональных компетенций.

В результате промежуточной аттестации по профессиональному модулю осуществляется комплексная проверка умений и знаний, а также динамика формирования общих и профессиональных компетенций (таблица 2).

Самостоятельная подготовка студентов по дисциплине предполагает следующие формы:

- работа с информационными источниками (поиск дополнительной информации по заданной теме в различных информационных источниках, таких как учебники, справочники, журналы и интернет);
- творческие задания (выполнение различных заданий, связанных с развитием творческого воображения и умением применять свои знания в самостоятельной работе);
- подготовка презентационных материалов (поиск и систематизация заданной информации, ее обработка, оформление в виде презентации и выступление в виде защиты перед группой);
- оформление лабораторных работ, отчетов и подготовка к их защите.

При оценивании лабораторной и самостоятельной работы студента учитывается:

- качество выполнения практической части работы;
- качество оформления отчета по работе;

Качество устных ответов на контрольные вопросы при защите работы.

Элемент учебной дисциплины	Формы и методы контроля					
	Текущий контроль		Рубежный контроль (другие формы контроля)		Промежуточная Аттестация(экзамен)	
	Форма контроля	Проверяемые ПК, ОК, У, З	Форма контроля	Проверяемые ПК, ОК, У, З	Форма контроля	Проверяемые ПК, ОК, У, З
Раздел 1 Технология разработки программного обеспечения	Самостоятельная работа Домашнее задание Проектно-исследовательская деятельность Лабораторные работы	ПК 3.1-3.6 ОК 1-9 У 1-2 З 1-11	Тестирование	ПК 3.1-3.6 ОК 1-9 У 1-2 З 1-11	Дифференцированный зачет	ПК 3.1-3.6 ОК 1-9 У 1-2 З 1-11

Элемент учебной дисциплины	Формы и методы контроля					
	Текущий контроль		Рубежный контроль (другие формы контроля)		Промежуточная Аттестация(экзамен)	
	Форма контроля	Проверяемые ПК, ОК, У, З	Форма контроля	Проверяемые ПК, ОК, У, З	Форма контроля	Проверяемые ПК, ОК, У, З
Раздел 2 Инструментальные средства разработки программного обеспечения	Самостоятельная работа Домашнее задание Проектно-исследовательская деятельность Лабораторные работы	ПК 3.1-3.6 ОК 1-9 У 1-2 З 1-11	Тестирование	ПК 3.1-3.6 ОК 1-9 У 1-2 З 1-11	Дифференцированный зачет	ПК 3.1-3.6 ОК 1-9 У 1-2 З 1-11
Раздел 3 Документирование и сертификация	Самостоятельная работа Домашнее задание Проектно-исследовательская деятельность Лабораторная работа	ПК 3.1-3.6 ОК 1-9 У 1-2 З 1-11	Тестирование	ПК 3.1-3.6 ОК 1-9 У 1-2 З 1-11	Дифференцированный зачет	ПК 3.1-3.6 ОК 1-9 У 1-2 З 1-11

3.2 Типовые задания для оценки освоения профессионального модуля

1) Входное тестирование

- 1) Какие программы можно отнести к системному программному обеспечению:
 - 1 операционные системы;**
 - 2 прикладные программы;
 - 3 игровые программы.
- 2) Какие программы можно отнести к системному ПО:
 - 1 драйверы;**
 - 2 текстовые редакторы;
 - 3 электронные таблицы;
 - 4 графические редакторы.
- 3) Специфические особенности ПО как продукта:
 - 1 продажа по ценам ниже себестоимости (лицензирование);**
 - 2 низкие материальные затраты при создании программ;**
 - 3 возможность создание программ небольшие коллективом или даже одним человеком;**
 - 4 разнообразие решаемых задач с помощью программных средств.**
- 4) Какие программы можно отнести к системному ПО:
 - 1 программа расчета заработной платы;
 - 2 электронные таблицы;
 - 3 СУБД (системы управления базами данных).**
- 5) Какие программы нельзя отнести к системному ПО:
 - 1 игровые программы;**
 - 2 компиляторы языков программирования;
 - 3 операционные системы;
 - 4 системы управления базами данных.
- 6) Какие программы можно отнести к прикладному программному обеспечению:
 - 1 электронные таблицы;**
 - 2 таблицы решений;
 - 3 СУБД (системы управления базами данных).
- 7) Какие программы можно отнести к прикладному ПО:
 - 1 программа расчета заработной платы;**
 - 2 диспетчер программ;
 - 3 программа «Проводник» (Explorer).
- 8) Какие программы нельзя отнести к прикладному ПО:
 - 1 компиляторы и (или) интерпретаторы;**
 - 2 текстовые и (или) графические редакторы;
 - 3 электронные таблицы.
- 9) Можно ли отнести операционную систему к программному обеспечению:
 - 1 да;**
 - 2 нет.
- 10) Можно ли отнести операционную систему к прикладному программному обеспечению:
 - 1 да;**
 - 2 нет.**
- 11) Специфические особенности ПО как продукта:
 - 1 низкие затраты при дублировании;**
 - 2 универсальность;
 - 3 простота эксплуатации;
 - 4 наличие поддержки (сопровождения) со стороны разработчика.

- 12) Какие программы можно отнести к системному ПО:
- 1 **утилиты;**
 - 2 экономические программы;
 - 3 статистические программы;
 - 4 мультимедийные программы.
- 13) Этап, занимающий наибольшее время, в жизненном цикле программы:
- 1 **сопровождение;**
 - 2 проектирование;
 - 3 тестирование;
 - 4 программирование;
 - 5 формулировка требований.
- 14) Этап, занимающий наибольшее время, при разработке программы:
- 1 **тестирование;**
 - 2 сопровождение;
 - 3 проектирование;
 - 4 программирование;
 - 5 формулировка требований.
- 15) Первый этап в жизненном цикле программы:
- 1 **формулирование требований;**
 - 2 анализ требований;
 - 3 проектирование;
 - 4 автономное тестирование;
 - 5 комплексное тестирование.
- 16) Один из необязательных этапов жизненного цикла программы:
- 1 **оптимизация;**
 - 2 проектирование;
 - 3 тестирование;
 - 4 программирование;
 - 5 анализ требований.
- 17) Самый большой этап в жизненном цикле программы:
- 1 **эксплуатация;**
 - 2 изучение предметной области;
 - 3 программирование;
 - 4 тестирование;
 - 5 корректировка ошибок.
- 18) Какой этап выполняется раньше:
- 1 **отладка;**
 - 2 тестирование.
- 19) Какой этап выполняется раньше:
- 1 отладка;
 - 2 оптимизация;
 - 3 **программирование;**
 - 4 тестирование.
- 20) Что выполняется раньше:
- 1 **компиляция;**
 - 2 отладка;
 - 3 компоновка;
 - 4 тестирование.
- 21) Что выполняется раньше:
- 1 **проектирование;**
 - 2 программирование;
 - 3 отладка;

- 4 тестирование.
- 22) В стадии разработки программы не входит:
 - 1 автоматизация программирования;**
 - 2 постановка задачи;
 - 3 составление спецификаций;
 - 4 эскизный проект;
 - 5 тестирование.
- 23) Самый важный критерий качества программы:
 - 1 работоспособность;**
 - 2 надежность;
 - 3 эффективность;
 - 4 быстродействие;
 - 5 простота эксплуатации.
- 24) Способы оценки качества:
 - 1 сравнение с аналогами;**
 - 2 наличие документации;
 - 3 оптимизация программы;
 - 4 структурирование алгоритма.
- 25) Существует ли связь между эффективностью и оптимизацией программы:
 - 1 да;**
 - 2 нет.
- 26) Наиболее важный критерий качества:
 - 1 надежность;**
 - 2 быстродействие;
 - 3 удобство в эксплуатации;
 - 4 удобный интерфейс;
 - 5 эффективность.
- 27) Способы оценки надежности:
 - 1 тестирование;**
 - 2 сравнение с аналогами;
 - 3 трассировка;
 - 4 оптимизация.
- 28) Повышает ли качество программ оптимизация:
 - 1 да;**
 - 2 нет.
- 29) Существует ли связь между надежностью и быстродействием:
 - 1 нет;**
 - 2 да.
- 30) В каких единицах можно измерить надежность:
 - 1 отказов/час;**
 - 2 км/час;
 - 3 Кбайт/сек;
 - 4 операций/сек.
- 31) В каких единицах можно измерить быстродействие:
 - 1 отказов/час;
 - 2 км/час;
 - 3 Кбайт/сек;
 - 4 операций/сек.**
- 32) Что относится к этапу программирования:
 - 1 написание кода программы;**
 - 2 разработка интерфейса;
 - 3 работоспособность;

- 4 анализ требований.
- 33) Последовательность этапов программирования:
1 компилирование, компоновка, отладка;
2 компоновка, отладка, компилирование;
3 отладка, компилирование, компоновка;
4 компилирование, отладка, компоновка.
- 34) Инструментальные средства программирования:
1 компиляторы, интерпретаторы;
2 СУБД (системы управления базами данных);
3 BIOS (базовая система ввода-вывода);
4 ОС (операционные системы).
- 35) На языке программирования составляется:
1 исходный код;
2 исполняемый код;
3 объектный код;
4 алгоритм.
- 36) Правила, которым должна следовать программа это:
1 алгоритм;
2 структура;
3 спецификация;
4 состав информации.
- 37) Можно ли внутри цикла поместить еще один цикл:
1 да;
2 нет.
- 38) Можно ли внутри условного оператора поместить еще одно условие:
1 да;
2 нет.
- 39) Можно ли одно большое (длинное) выражение разбить на несколько выражений:
1 да;
2 нет.
- 40) Если имеется стандартная функция, нужно ли писать собственную:
1 нет;
2 да.
- 41) Доступ, при котором записи файла читаются в физической последовательности, называется:
1 последовательным;
2 прямым;
3 простым;
4 основным.
- 42) Доступ, при котором записи файла обрабатываются в произвольной последовательности, называется:
1 прямым;
2 последовательным;
3 простым;
4 основным.
- 43) Методы программирования (укажите НЕ верный ответ):
1 логическое;
2 структурное;
3 модульное.
- 44) Что выполняется раньше:
1 разработка алгоритма;

- 2 выбор языка программирования;
 - 3 написание исходного кода;
 - 4 компиляция.
- 45) Можно ли переменным присваивать произвольные идентификаторы:
- 1 **да;**
 - 2 нет.
- 46) Найдите НЕ правильное условие для создания имен:
- 1 **имена могут содержать пробелы;**
 - 2 длинное имя можно сократить;
 - 3 из имени лучше выбрасывать гласные;
 - 4 можно использовать большие буквы.
- 47) Какие символы не допускаются в именах переменных:
- 1 **пробелы;**
 - 2 цифры;
 - 3 подчеркивание
- 48) Можно ли использовать имена, которые уже были использованы в другой программе (модуле):
- 1 **да;**
 - 2 нет.
- 49) Можно ли ставить знак подчеркивания в начале имени:
- 1 **да, но не рекомендуется;**
 - 2 да, без ограничений;
 - 3 нет.
- 50) Как называется способ составления имен переменных, когда в начале имени сообщается тип переменной:
- 1 прямым указанием;
 - 2 **венгерской нотацией;**
 - 3 структурным программированием;
 - 4 поляризацией.
- 51) Можно ли писать комментарии в отдельной строке:
- 1 **да;**
 - 2 нет.
- 52) Транслируются ли комментарии:
- 1 да;
 - 2 **нет.**
- 53) Наличие комментариев позволяет:
- 1 **быстрее найти ошибки в программе;**
 - 2 быстрее писать программы;
 - 3 быстрее выполнять программы.
- 54) Наличие комментариев позволяет:
- 1 **легче разобраться в программе;**
 - 2 применять сложные структуры;
 - 3 увеличить быстродействие.
- 55) Наличие комментариев позволяет:
- 1 **улучшить читаемость программы;**
 - 2 улучшить эксплуатацию программы;
 - 3 повысить надежность программы.
- 56) Что определяет выбор языка программирования:
- 1 **область приложения;**
 - 2 знание языка;
 - 3 наличие дополнительных библиотек.
- 57) Возможно ли комбинирование языков программирования в рамках одной зада-

чи:

- 1 да;
- 2 нет.

58) Допустимо ли комбинирование языков программирования в рамках одной задачи :

- 1 да;
- 2 нет.

59) Для каких задач характерно использование большого количества исходных данных, выполнение операций поиска, группировки:

- 1 для экономических задач;
- 2 для системных задач;
- 3 для инженерных задач.

60) Для каких задач характерен большой объем вычислений, использование сложного математического аппарата:

- 1 для инженерных задач;
- 2 для системных задач;
- 3 для экономических задач.

61) На каком этапе производится выбор языка программирования:

- 1 проектирование;
- 2 программирование;
- 3 отладка;
- 4 тестирование.

62) Можно ли использовать комбинацию языков программирования в рамках одного проекта:

- 1 да;
- 2 нет.

63) Для решения экономических задач характерно применение:

- 1 СУБД (систем управления базами данных);
- 2 языков высокого уровня;
- 3 языков низкого уровня;
- 4 применение сложных математических расчетов.

64) Для решения инженерных задач характерно применение:

- 1 САПР (систем автоматизированного проектирования);
- 2 СУБД (систем управления базами данных);
- 3 ОС (операционных систем).

65) Причины синтаксических ошибок:

- 1 плохое знание языка программирования;
- 2 ошибки в исходных данных;
- 3 ошибки, допущенные на более ранних этапах;
- 4 неправильное применение процедуры тестирования.

66) Когда можно обнаружить синтаксические ошибки:

- 1 при компиляции;
- 2 при отладке;
- 3 при тестировании;
- 4 на этапе проектирования;
- 5 при эксплуатации.

67) Ошибки компоновки заключаются в том, что:

- 1 указано внешнее имя, но не объявлено;
- 2 неправильно использовано зарезервированное слово;
- 3 составлено неверное выражение;
- 4 указан неверный тип переменной.

68) Могут ли проявиться ошибки при изменении условий эксплуатации:

- 1 да;
2 нет.
- 69) Могут ли проявиться ошибки при изменении в предметной области:
1 да;
2 нет.
- 70) Возможно ли программирование с защитой от ошибок:
1 да;
2 нет.
- 71) Есть ли недостатки программирования с защитой от ошибок:
1 да;
2 нет.
- 72) Защитное программирование это:
1 **встраивание в программу отладочных средств;**
2 создание задач защищенных от копирования;
3 разделение доступа в программе;
4 использование паролей;
5 оформление авторских прав на программу.
- 73) Вид ошибки с неправильным написанием служебных слов (операторов):
1 **синтаксическая;**
2 семантическая;
3 логическая;
4 символьная.
- 74) Вид ошибки с неправильным использованием служебных слов (операторов):
1 **семантическая;**
2 синтаксическая;
3 логическая;
4 символьная.
- 75) Ошибки при написании программы бывают:
1 **синтаксические;**
2 орфографические;
3 лексические;
4 фонетические;
5 морфологические.
- 76) Процедура поиска ошибки, когда известно, что она есть это:
1 **отладка;**
2 тестирование;
3 компоновка;
4 транзакция;
5 трансляция.
- 77) Программа для просмотра значений переменных при выполнении программы:
1 **отладчик;**
2 компилятор;
3 интерпретатор;
4 трассировка;
5 тестирование.
- 78) Отладка – это:
1 **процедура поиска ошибок, когда известно, что ошибка есть;**
2 определение списка параметров;
3 правило вызова процедур (функций);
4 составление блок-схемы алгоритма.
- 79) Когда программист может проследить последовательность выполнения команд программы:

- 1 при трассировке;
 - 2 при тестировании;
 - 3 при компиляции;
 - 4 при выполнении программы;
 - 5 при компоновке.
- 80) На каком этапе создания программы могут появиться синтаксические ошибки:
- 1 программирование;
 - 2 проектирование;
 - 3 анализ требований;
 - 4 тестирование.
- 81) Когда приступают к тестированию программы:
- 1 когда программа уже закончена;
 - 2 после постановки задачи;
 - 3 на этапе программирования;
 - 4 на этапе проектирования;
 - 5 после составления спецификаций,
- 82) Тестирование бывает:
- 1 автономное;
 - 2 инструментальное;
 - 3 визуальное;
 - 4 алгоритмическое.
- 83) Тестирование бывает:
- 1 комплексное;
 - 2 инструментальное;
 - 3 визуальное;
 - 4 алгоритмическое.
- 84) Существует ли различие между отладкой и тестированием:
- 1 да;
 - 2 нет.
- 85) При комплексном тестировании проверяются:
- 1 согласованность работы отдельных частей программы;
 - 2 правильность работы отдельных частей программы;
 - 3 быстродействие программы;
 - 4 эффективность программы.
- 86) Чему нужно уделять больше времени, чтобы получить хорошую программу:
- 1 тестированию;
 - 2 программированию;
 - 3 отладке;
 - 4 проектированию.
- 87) Процесс исполнения программы с целью обнаружения ошибок:
- 1 тестирование;
 - 2 кодирование;
 - 3 сопровождение;
 - 4 проектирование.
- 88) Автономное тестирование это:
- 1 тестирование отдельных частей программы;
 - 2 инструментальное средство отладки;
 - 3 составление блок-схем;
 - 4 пошаговая проверка выполнения программы.
- 89) Трассировка это:
- 1 проверка пошагового выполнения программы;
 - 2 тестирование исходного кода;

- 3 отладка модуля;
- 4 составление блок-схемы алгоритма.
- 90) Локализация ошибки:
 - 1 определение места возникновения ошибки;**
 - 2 определение причин ошибки;
 - 3 обнаружение причин ошибки;
 - 4 исправление ошибки.
- 91) Назначение тестирования:
 - 1 повышение надежности программы;**
 - 2 обнаружение ошибок;
 - 3 повышение эффективности программы;
 - 4 улучшение эксплуатационных характеристик;
 - 5 приведение программы к структурированному виду.
- 92) Назначение отладки:
 - 1 поиск причин существующих ошибок;**
 - 2 поиск возможных ошибок;
 - 3 составление спецификаций;
 - 4 разработка алгоритма.
- 93) Инструментальные средства отладки (НЕ правильный ответ):
 - 1 компиляторы;**
 - 2 отладчики;
 - 3 трассировка.
- 94) Отладка программ это:
 - 1 локализация и исправление ошибок;**
 - 2 алгоритмизация программирования;
 - 3 компиляция и компоновка.
- 95) Что выполняется раньше, автономная или комплексная отладка:
 - 1 автономная;**
 - 2 комплексная.
- 96) Что выполняется раньше, отладка или тестирование:
 - 1 отладка;**
 - 2 тестирование.
- 97) Что такое автоматизация программирования:
 - 1 создание исходного кода программными средствами;**
 - 2 создание исходного кода при помощи компилятора;
 - 3 создание исходного кода без разработки алгоритма.
- 98) В чем сущность автоматизации программирования:
 - 1 создание программы без написания ее текста;**
 - 2 получение готовой программы без выполнения компоновки;
 - 3 в отсутствии компиляции.
- 99) Возможна ли автоматизация программирования:
 - 1 да;**
 - 2 нет.
- 100) Создание исполняемого кода программы без написания исходного кода называется:
 - 1 составлением спецификаций;
 - 2 отладкой;
 - 3 проектированием.
 - 4 автоматизацией программирования;**
- 101) Одно из преимуществ автоматизации программирования:
 - 1 наглядное программирование с визуальным контролем;**
 - 2 получение стандартной программы;

- 3 создание программы с оптимальным кодом.
- 102) Один из методов автоматизации программирования:
- 1 структурное программирование;
 - 2 модульное программирование;
 - 3 визуальное программирование;**
 - 4 объектно-ориентированное программирование.
- 103) Влияет ли автоматизация программирования на эффективность программы:
- 1 нет;**
 - 2 да
- 104) Автоматизация программирования позволяет:
- 1 повысить надежность программы;
 - 2 сократить время разработки программы;**
 - 3 повысить быстродействие программы.
- 105) Позволяет ли автоматизация программирования всегда создавать эффективные программы:
- 1 да.
 - 2 нет;**
- 106) Позволяет ли автоматизация программирования всегда создавать надежные программы:
- 1 нет;**
 - 2 да.
- 107) Недостаток автоматизации программирования:
- 1 низкое быстродействие;
 - 2 большой размер программы;**
 - 3 сложность программы.
- 108) Возможны ли ошибки при автоматизации программирования:
- 1 да;**
 - 2 нет.
- 109) Нужно ли выполнять тестирование при автоматизации программирования:
- 1 да;**
 - 2 нет.
- 110) Выполняется ли процедура компиляции при автоматизации программирования:
- 1 да;**
 - 2 нет.
- 111) Что легко поддается автоматизации:
- 1 интерфейс;**
 - 2 работа с файлами;
 - 3 сложные логические задачи;
 - 4 алгоритмизация.
- 112) Относится ли визуальное программирование к средствам автоматизации:
- 1 да;**
 - 2 нет.
- 113) Нахождение наилучшего варианта из множества возможных:
- 1 оптимизация;**
 - 2 тестирование;
 - 3 автоматизация;
 - 4 отладка;
 - 5 сопровождение.
- 114) Что такое оптимизация программ:
- 1 улучшение работы существующей программы;**
 - 2 создание удобного интерфейса пользователя;

- 3 разработка модульной конструкции программы;
- 4 применение методов объектно-ориентированного программирования.
- 115) Критерии оптимизации:
 - 1 **время выполнения или размер требуемой памяти;**
 - 2 размер программы и ее эффективность;
 - 3 независимость модулей;
 - 4 качество программы, ее надежность.
- 116) Критерии оптимизации:
 - 1 **эффективность использования ресурсов;**
 - 2 структурирование алгоритма;
 - 3 структурирование программы.
- 117) Возможна ли оптимизация программ без участия программиста:
 - 1 **да;**
 - 2 нет.
- 118) Возможна ли оптимизация циклов:
 - 1 **да;**
 - 2 нет.
- 119) В чем заключается оптимизация условных выражений:
 - 1 **в изменении порядка следования элементов выражения;**
 - 2 в использовании простых логических выражений;
 - 3 в использовании сложных логических выражений;
 - 4 в использовании операций AND, OR и NOT.
- 120) Оптимизация циклов заключается в:
 - 1 **уменьшении количества повторений тела цикла;**
 - 2 просмотре задачи с другой стороны;
 - 3 упрощение задачи за счет включения логических операций.
- 121) Оптимизация программы это:
 - 1 **модификация;**
 - 2 отладка;
 - 3 повышение сложности программы;
 - 4 уменьшение сложности программы.
- 122) Критерии оптимизации программы:
 - 1 **быстродействие или размер программы;**
 - 2 быстродействие и размер программы;
 - 3 надежность или эффективность;
 - 4 надежность и эффективность.
- 123) Результат оптимизации программы:
 - 1 **эффективность;**
 - 2 надежность;
 - 3 машино-независимость;
 - 4 мобильность.
- 124) Сущность оптимизации циклов:
 - 1 **сокращение количества повторений выполнения тела цикла;**
 - 2 сокращение тела цикла;
 - 3 представление циклов в виде блок-схем;
 - 4 трассировка циклов;
 - 5 поиск ошибок в циклах.
- 125) В чем сущность модульного программирования:
 - 1 **в разбиении программы на отдельные функционально независимые ча-**
 - сти;**
 - 2 в разбиении программы на отдельные равные части;
 - 3 в разбиение программы на процедуры и функции;

- 126) Можно ли сочетать модульное и структурное программирование:
1 да;
2 нет.
- 127) Может ли модуль включать несколько процедур или функций:
1 да;
2 нет.
- 128) Рекомендуемые размеры модулей:
1 **небольшие;**
2 большие;
3 равные;
4 фиксированной длины.
- 129) В чем заключается независимость модуля:
1 **в написании, отладке и тестировании независимо от остальных модулей;**
2 в разработке и написании независимо от других модулей;
3 в независимости от работы основной программы.
- 130) При модульном программировании желательно, чтобы модуль имел:
1 большой размер;
2 **небольшой размер;**
3 фиксированный размер;
4 любой размер.
- 131) Модульное программирование это:
1 **разбиение программы на отдельные части;**
2 структурирование;
3 использование стандартных процедур и функций.
- 132) Можно ли использовать оператор GO TO в модульных программах:
1 **можно;**
2 нельзя.
- 133) Разрешается ли использование циклов при модульном программировании:
1 да;
2 нет.
- 134) Разрешается ли использование условных операторов при модульном программировании:
1 да;
2 нет.
- 135) Сократится ли размер программы, если ее написать в виде набора модулей:
1 **нет;**
2 да.
- 136) Достоинство модульного программирования:
1 **создание программы по частям в произвольном порядке;**
2 не требует компоновки;
3 всегда дает эффективные программы;
4 снижает количество ошибок.
- 137) Недосток модульного программирования:
1 увеличивает трудоемкость программирования;
2 **усложняет процедуру комплексного тестирования;**
3 снижает быстродействие программы;
4 не позволяет выполнять оптимизацию программы.
- 138) Достоинство модульного программирования:
1 **возможность приступить к тестированию до завершения написания всей программы;**
2 не требует комплексного тестирования;
3 уменьшает размер программы;

- 4 повышает надежность программы.
- 139) Допустимо ли использование оператора GO TO при структурном программировании:
- 1 **нет;**
 - 2 да.
- 140) Можно ли сочетать структурное программирование с модульным:
- 1 **можно;**
 - 2 нельзя;
 - 3 только в особых случаях.
- 141) Любую ли программу можно привести к структурированному виду:
- 1 **любую;**
 - 2 не все;
 - 3 нельзя.
- 142) Можно ли использовать оператор GO TO в структурированных программах:
- 1 можно;
 - 2 **нельзя;**
 - 3 только в особых случаях.
- 143) Возможно, ли преобразовать неструктурированную программу к структурному виду:
- 1 **да;**
 - 2 нет.
- 144) Возможно ли программирование без оператора GO TO:
- 1 **да;**
 - 2 нет.
- 145) При структурном программировании задача выполняется:
- 1 **поэтапным разбиением на более легкие задачи;**
 - 2 без участия программиста;
 - 3 объединением отдельных модулей программы.
- 146) Разрешается ли использование оператора GO TO при структурном программировании:
- 1 **нет;**
 - 2 да;
 - 3 иногда.
- 147) Разрешается ли использование циклов при структурном программировании:
- 1 **да;**
 - 2 нет.
- 148) Разрешается ли использование оператора IF при структурном программировании:
- 1 **да;**
 - 2 нет.
- 149) Программирование без GO TO применяется. при:
- 1 **структурном программировании;**
 - 2 модульном программировании;
 - 3 объектно-ориентированном программировании;
 - 4 все ответы верные.
- 150) Достоинство структурного программирования:
- 1 **можно приступить к комплексному тестированию на раннем этапе разработки;**
 - 2 можно приступить к автономному тестированию на раннем этапе разработки;
 - 3 нет необходимости выполнять тестирование;
 - 4 можно пренебречь отладкой.
- 151) Достоинство структурного программирования:

- 1 облегчает работу над большими и сложными проектами;
 - 2 повышает быстродействие программы;
 - 3 снижает затраты на программирование.
- 152) Недостаток структурного программирования:
- 1 увеличивает размер программы;
 - 2 снижает эффективность;
 - 3 уменьшает количество ошибок;
 - 4 не требует отладки.
- 153) Повышает ли читабельность программ структурное кодирование:
- 1 да;
 - 2 нет.
- 154) Разрешается ли использование циклов при объектно-ориентированном программировании:
- 1 да;
 - 2 нет.
- 155) Разрешается ли использование оператора IF при объектно-ориентированном программировании:
- 1 да;
 - 2 нет.
- 156) Предусматривает ли объектно-ориентированное программирование использование стандартных процедур и функций:
- 1 да;
 - 2 нет.
- 157) Можно ли сочетать объектно-ориентированное и структурное программирование
- 1 можно;
 - 2 нельзя.
- 158) Можно ли сочетать объектно-ориентированное и модульное программирование:
- 1 можно;
 - 2 нельзя.
- 159) Что такое объект, в объектно-ориентированное программировании:
- 1 тип данных;
 - 2 структура данных;
 - 3 событие;
 - 4 обработка событий;
 - 5 использование стандартных процедур.
- 160) Инкапсуляция это:
- 1 определение новых типов данных;
 - 2 определение новых структур данных;
 - 3 объединение переменных, процедур и функций в одно целое;
 - 4 разделение переменных, процедур и функций;
 - 5 применение стандартных процедур и функций.
- 161) Наследование это:
- 1 передача свойств экземплярам;
 - 2 передача свойств предкам;
 - 3 передача свойств потомкам;
 - 4 передача событий потомкам.
- 162) Полиморфизм это:
- 1 изменение поведения потомков, имеющих общих предков;
 - 2 передача свойств по наследству;
 - 3 изменение поведения потомков на разные события;

- 4 изменение поведения экземпляров, имеющих общих предков;
- 163 Три "кита" объектно-ориентированного метода программирования:
- 1 предки, родители, потомки;
 - 2 полиморфизм, инкапсуляция, наследование;**
 - 3 свойства, события, методы;
 - 4 визуальные, не визуальные компоненты и запросы.
- 164 Какое утверждение верно:
- 1 предки наследуют свойства родителей;
 - 2 родители наследуют свойства потомков;
 - 3 потомки не могут иметь общих предков;
 - 4 потомки наследуют свойства родителей.**
- 165) Может ли дочерний элемент иметь двух родителей:
- 1 да;
 - 2 нет;**
 - 3 только для визуальных элементов;
 - 4 если их свойства совпадают.
- 166) Могут ли два визуальных компонента иметь общего предка:
- 1 да;**
 - 2 нет;
 - 3 если их свойства совпадают;
 - 4 если их методы совпадают.
- 167) Есть ли различие между объектом и экземпляром:
- 1 да;**
 - 2 нет;
 - 3 если у них общий предок.
- 168) Есть ли различие в поведении объекта и экземпляра того же типа:
- 1 да;
 - 2 если у них есть общий предок;
 - 3 нет;**
 - 4 если у них нет общего предков.
- 169) Изменение свойств, приводит к изменению поведения экземпляра:
- 1 нет;
 - 2 только для визуальных;
 - 3 только НЕ для визуальных ;
 - 4 да .**
- 170) Можно ли свойствам присваивать значения:
- 1 да (всегда);
 - 2 не всегда;**
 - 3 нет.
- 171) Можно ли переопределять методы:
- 1 да;**
 - 2 нет.
- 172) Можно ли переопределять свойства:
- 1 да;
 - 2 нет.**
- 173) Могут ли два различных объекта реагировать на событие по-разному:
- 1 да;**
 - 2 нет.
- 174) Могут ли два экземпляра одного объекта реагировать на событие по-разному:
- 1 да;**
 - 2 нет.
- 175) Какой методикой проектирования пользуются при структурном программировании?

вании:

- 1 сверху вниз;**
 - 2 снизу-вверх.
- 176) Какой этап проектирования может быть исключен:
 - 1 эскизный проект;**
 - 2 технический проект;
 - 3 рабочий проект.
- 177) Какие этапы проектирования можно объединять:
 - 1 технический и рабочий;**
 - 2 эскизный и рабочий;
 - 3 технический и эскизный.
- 178) Модульное программирование применимо при:
 - 1 проектировании сверху вниз;
 - 2 проектирование снизу-вверх;**
- 179) Процесс преобразования постановки задачи в план алгоритмического или вычислительного решения это:
 - 1 проектирование;**
 - 2 анализ требований;
 - 3 программирование;
 - 4 тестирование.
- 180) Составление спецификаций это:
 - 1 формализация задачи;**
 - 2 эскизный проект;
 - 3 поиск алгоритма;
 - 4 отладка.
- 181) Этап разработки программы, на котором дается характеристика области применения программы:
 - 1 техническое задание;**
 - 2 эскизный проект;
 - 3 технический проект;
 - 4 внедрение;
 - 5 рабочий проект.
- 182) Укажите правильную последовательность создания программы:
 - 1 формулирование задачи, анализ требований, проектирование, программирование;**
 - 2 анализ требований, проектирование, программирование, тестирование, отладка;
 - 3 анализ требований, программирование, проектирование, тестирование;
 - 4 анализ требований, проектирование, программирование, модификация, трассировка;
 - 5 формулирование задачи, анализ требований, программирование, проектирование, отладка.
- 183) Уточнение структуры входных и выходных данных, разработка алгоритмов, определение элементов интерфейса входят в:
 - 1 технический проект;**
 - 2 рабочий проект;
 - 3 эскизный проект.
- 184) Несуществующий метод проектирования:
 - 1 алгоритмическое;**
 - 2 нисходящее;
 - 3 восходящее.
- 185) Метод проектирования:

- 1 нисходящее;
 - 2 алгоритмическое;
 - 3 логическое;
 - 4 использование языков программирования;
 - 5 составление блок-схем.
- 186) Нисходящее проектирование это:
- 1 **последовательное уточнение (детализация);**
 - 2 составление блок-схем;
 - 3 разделение программы на отдельные участки (блоки);
 - 4 трассировка.
- 187) Признаки нисходящего программирования:
- 1 **последовательная детализация;**
 - 2 наличие оптимизации;
 - 3 наличие тестирования;
 - 4 автоматизация программирования.
- 188) Какой методикой пользуются при структурном программировании:
- 1 **сверху вниз;**
 - 2 снизу-вверх.
- 189) Проектирование сверху вниз это:
- 1 **последовательное разбиение общих задач на более мелкие;**
 - 2 составление из отдельных модулей большой программы.
- 190) Проектирование снизу-вверх это:
- 1 **составление из отдельных модулей большой программы;**
 - 2 последовательное разбиение общих задач на более мелкие.
- 191) Модульное программирование применимо при:
- 1 проектировании сверху вниз;
 - 2 проектировании снизу-вверх;
 - 3 **и в том, и другом случае;**
 - 4 ни в коем случае.
- 192) Какой методикой проектирования пользуются при структурном программировании:
- 1 **сверху вниз;**
 - 2 снизу-вверх.
- 193) В чем заключается иерархический подход в решении задачи:
- 1 **в последовательном разбиении задачи на более мелкие составные части;**
 - 2 в выделении основных и второстепенных элементов;
 - 3 в возможности параллельного выполнения отдельных частей задачи.
- 194) Какой метод проектирования соответствует иерархическому подходу в решении задачи:
- 1 **нисходящее (сверху вниз);**
 - 2 восходящее (снизу-вверх).
- 195) В каких единицах измеряются затраты на проектирование:
- 1 **в человеко-днях;**
 - 2 в долларах;
 - 3 в тенге;
 - 4 в килобайтах.
- 196) Зависит ли трудоемкость разработки от сложности алгоритма:
- 1 **да;**
 - 2 нет.
- 197) Зависит ли трудоемкость разработки от количества программистов:
- 1 да;
 - 2 **нет.**

- 198) Зависит ли трудоемкость разработки от языка или системы программирования:
- 1 да;**
 - 2 нет.
- 199) Зависит ли трудоемкость разработки от количества обрабатываемой информации:
- 1 да;
 - 2 нет.**
- 200) Зависит ли трудоемкость разработки от вида информации:
- 1 да;**
 - 2 нет.
- 201) Если вы приобрели программу законным путем, являетесь ли вы собственником программы:
- 1 нет;**
 - 2 да.
- 202) Если вы приобрели программы законным путем, имеете ли вы право вносить в нее изменения:
- 1 нет;**
 - 2 да
- 203) Если вы приобрели программы законным путем, имеете ли вы право продать ее:
- 1 да;**
 - 2 нет.
- 204) Кому принадлежит право собственности на ПО:
- 1 разработчику;**
 - 2 продавцу;
 - 3 покупателю.
- 205) Кому принадлежит авторское право на ПО:
- 1 разработчику;**
 - 2 продавцу;
 - 3 покупателю.
- 206) Что охраняется законом:
- 1 структура базы данных;
 - 2 содержание базы данных**

2) Самостоятельная работа

- работа с информационными источниками;
- подготовка презентационных материалов.

3) Домашнее задание

- Нарисовать схему вспомогательных процессов жизненного цикла программного продукта;
- Составить график длительности этапов жизненного цикла программного продукта;
- Провести выбор модели жизненного цикла разработки программного продукта;
- Построить график структуры процесса и организации, занимающейся разработкой программных продуктов;
- Провести сбор метрик;
- Провести оценку рисков при выполнении программного проекта;
- Создание прототипа программного продукта;
- Графическое оформление проекта программного продукта;
- Создание программного продукта;

- Разработка тестов;
- Создание плана сопровождения программного продукта;
- Провести классификацию поставляемых программных продуктов;
- Проверить связь методов обеспечения надежности с этапами жизненного цикла разработки программного продукта;
- Провести сравнение различных настольных клиентских технологий и сценариев;
- Провести сравнение версий WPF;
- Провести оформление макета страницы;
- Составление конспекта различных стилей пользовательского интерфейса;
- Провести тестирование и составить отчет;
- Проверить данные между источником и целью привязки;
- Провести связывание коллекции объектов, сортировки, фильтрации;
- Составление списка асинхронных процессов;
- Виды поддержки пользователей с ограниченными возможностями;
- Разобрать виды графики;
- Составление таблиц элементов управления;
- Разобрать присоединенные свойства;
- Типы и источники дефектов и ошибок в комплексах программ;
- Формирование Руководства по документированию для сертификации системы менеджмента качества программных средств на основе стандарта ISO 10013.

4) Проектно-исследовательская деятельность

- CRM системы. Решаемый класс задач и методы их решения. Тенденции развития. Истории развития;
- ERP системы. Решаемый класс задач и методы их решения. Тенденции развития. Истории развития;
- Модель SEI CMM (определение уровня зрелости IT-компаний);
- Системы менеджмента качества в российских IT-компаниях;
- Методология RUP. Обзор;
- Методология экстремального программирования. Обзор;
- Сравнение технологии RUP и технологии экстремального программирования;
- Методология управления проектами MSF. Обзор;
- ARIS. Обзор методологии;
- Человеческий фактор в IT проектах;
- Разновидности IT проектов;
- Командообразование и IT проектах;
- Мотивация в IT проектах;
- Обзор программных средств для управления проектами;
- Реинжиниринг бизнес процессов. Проблемы и решение;
- Информационные системы в логистике;
- IT-аутсорсинг. Оценка видов деятельности. Современное состояние. Перспективы;
- IT-консалтинг. Оценка видов деятельности. Современное состояние. Перспективы;
- Оценка IT проектов. Проблемы и решения;
- Методики ROI и TCO. Обзор;
- Международные организации по управлению проектами. Сертификации менеджеров проектов. Обзор;
- Технология проектного офиса. Основной смысл. Сравнение с классической технологией управления проектами;
- Матричные структуры в организации;
- Стандарт ISO 10006:2003 «Системы менеджмента качества. Руководящие указания по управлению качеством в проектах». Обзор;

- Способы и методы разработки пользовательских интерфейсов;
- Microsoft Remoting в разработке распределительных приложений;
- SOAP и обмен сообщениями;
- Приемники сообщений и компоненты исполнения Microsoft Remoting;
- Средства автоматизированного проектирования и их использование при разработке корпоративных приложений;
- Требования к квалификации аудиторов по сертификации производства и систем качества программных продуктов;
- Подготовка и документирование организации процессов сертификации производства и системы качества предприятия;
- Организация сертификационных испытаний производства и систем качества программных средств.

5) Лабораторная работа

Раздел 1:

Лабораторная работа 01 «Жизненный цикл программного продукта».

Лабораторная работа 02 «Взаимосвязь между процессами жизненного цикла программного продукта».

Лабораторная работа 03 «Планирование длительности основных этапов».

Лабораторная работа 04 «Создание плана работы на основе каскадной модели».

Лабораторная работа 05 «Создание плана работы на основе V-образной модели».

Лабораторная работа 06 «Создание плана работы на основе модели прототипирования».

Лабораторная работа 07 «Создание плана работы на основе модели быстрой разработки приложений».

Лабораторная работа 08 «Создание плана работы на основе многопроходной модели».

Лабораторная работа 09 «Создание плана работы на основе спиральной модели».

Лабораторная работа 10 «Составление структуры СММ».

Лабораторная работа 11 «Составление структуры процесса и организации, занимающейся разработкой».

Лабораторная работа 12 «Составление метрик процесса».

Лабораторная работа 13 «Составление метрик проекта».

Лабораторная работа 14 «Составление метрик программного продукта».

Лабораторная работа 15 «Разработка плана по созданию программного продукта».

Лабораторная работа 16 «Составление спецификаций по требованиям заказчика».

Лабораторная работа 17 «Разработка проекта программного продукта».

Лабораторная работа 18 «Кодирование программного продукта».

Лабораторная работа 19 «Создание документации пользователя».

Лабораторная работа 20 «Тестирование».

Лабораторная работа 21 «Сопровождение программного продукта».

Лабораторная работа 22 «Поставка программного продукта».

Лабораторная работа 23 «Ошибки».

Лабораторная работа 24 «Составление плана обеспечения надежности».

Раздел 2:

Лабораторная работа 01 «Проектирование настольного клиентского приложения».

Лабораторная работа 02 «Применение новых возможностей в Visual Studio 2013 и в WPF версии 4».

Лабораторная работа 03 «Разработка макета страницы и использование элементов управления».

Лабораторная работа 04 «Создание согласованного пользовательского интерфейса».

Лабораторная работа 05 «Разработка наборов тестовых данных».

Лабораторная работа 06 «Представление данных во время разработки».

Лабораторная работа 07 «Создание представления данных «мастер-детали и настройка отображения данных с помощью шаблонов данных»».

Лабораторная работа 08 «Применение асинхронных процессов для реализации отзывчивого пользовательского интерфейса».

Лабораторная работа 09 «Интеграция локализации и помощи для пользователя».

Лабораторная работа 10 «Создание двумерной графики, отображения изображений в WPF приложения и добавление мультимедийного содержимого».

Лабораторная работа 11 «Создание и печать документов».

Лабораторная работа 12 «Разработка элементов управления».

Лабораторная работа 13 «Реализация в Expression Blend поведений, триггеров и действий».

Лабораторная работа 14 «Визуализация данных».

Раздел 3:

Лабораторная работа 01 «Сбор информации».

Лабораторная работа 02 «Основные понятия, цели и виды сертификации программных средств».

Лабораторная работа 03 «Стандартизация и сертификация как основа для обеспечения качества и безопасности программных продуктов».

Лабораторная работа 04 «Принципы промышленной сертификации и стандартизации процессов производства и продуктов».

Лабораторная работа 05 «Формирование назначения, функций и технического задания на проект системы».

Лабораторная работа 06 «Системные основы разработки требований и программным продуктом».

Лабораторная работа 07 «Принципы организации производства программных продуктов».

Лабораторная работа 08 «Процессы уравнивания проектами программных средств на основе модели – СММІ».

Лабораторная работа 09 «Менеджмент-административное управление обеспечением качества систем на основе стандартов ISO 9000:2000».

Лабораторная работа 10 «Подготовка стандартов жизненного цикла программных средств для производства».

Лабораторная работа 11 «Базовые стандарты жизненного цикла программных средств».

Лабораторная работа 12 «Руководства по применению базовых стандартов жизненного цикла программных средств».

Лабораторная работа 13 «Стандарты сопровождения и управления конфигурацией программных средств».

Лабораторная работа 14 «Подготовка технологии производства программных продуктов и системы качества предприятия к сертификации».

Лабораторная работа 15 «Адаптация для сертификации базовых стандартов управления производством ISO 12207 и системой качества ISO 90003 программных средств».

Лабораторная работа 16 «Формирование Руководства по планированию качества программных средств для сертификации на основе стандарта ISO 10005».

4 Контрольно-оценочные материалы для итоговой аттестации по профессиональному модулю

Оценка освоения МДК предусматривает проведение дифференцированного зачета.
Оценка освоения ПМ предусматривает проведение экзамена.

4.1 Методические материалы, определяющие процедуры оценивания знаний, умений, практического опыта, характеризующие этапы формирования компетенций

Вопросы к экзамену по модулю:

- 1) Понятие жизненного цикла программного продукта.
- 2) Основные процессы жизненного цикла программного продукта.
- 3) Вспомогательные процессы жизненного цикла программного продукта.
- 4) Организационные процессы жизненного цикла программного продукта.
- 5) Взаимосвязь между процессами жизненного цикла программного продукта.
- 6) Длительность основных этапов.
- 7) Характеристика основных этапов.
- 8) Понятие модели жизненного цикла разработки программного продукта.
- 9) Обзор существующих моделей.
- 10) Каскадная модель.
- 11) V-образная модель.
- 12) Модель прототипирования.
- 13) Модель быстрой разработки приложений.
- 14) Многопроходная модель.
- 15) Спиральная модель.
- 16) Вспомогательные процессы.
- 17) Кризис программирования и способ выхода из него.
- 18) Модель CMM-SEI.
- 19) Управление качеством разработки программного продукта с помощью системы стандартов ISO 9001.
- 20) Примерная структура процесса и организации, занимающейся разработкой программных продуктов.
- 21) Роль метрик в процессе разработки программных продуктов.
- 22) Метрики и модель CMM-SEI.
- 23) Парадигма Бейзили.
- 24) Набор основных метрических показателей.
- 25) Лабораторные работы.
- 26) Структура разделения работ по созданию программного продукта.
- 27) Оценка объемов и сложности программного продукта.
- 28) Оценка технических, нетехнических и финансовых ресурсов для выполнения программного проекта.
- 29) Оценка возможных рисков при выполнении программного продукта.
- 30) Составление временного графика выполнения программного проекта.
- 31) Общие сведения об управлении требованиями.
- 32) Цикл формирования требований.
- 33) Анализ и структурирование первичных требований заказчика.
- 34) Конструирование прототипа.
- 35) Составление спецификаций по требованиям заказчика.
- 36) Общая характеристика и компоненты проектирования.
- 37) Эволюция разработки программного продукта.
- 38) Структурное программирование.

- 39) Объектно-ориентированное проектирование.
- 40) Кодирование.
- 41) Тестирование.
- 42) Разработка справочной системы программного продукта.
- 43) Создание документации пользователя.
- 44) Создание версии и инсталляции программного продукта.
- 45) Общая характеристика тестирования и его цикл.
- 46) Виды тестирования.
- 47) Программные ошибки.
- 48) Тестирование документации.
- 49) Разработка и выполнение тестов.
- 50) Роль этапа сопровождения в жизненном цикле программного продукта.
- 51) Собираемые метрики, используемые инструменты и шаблон.
- 52) Общие сведения об управлении поставками.
- 53) Классификация поставляемых программных продуктов.
- 54) Действия, выполняемые при поставке программного продукта.
- 55) Используемые термины.
- 56) Основные понятия о надежности программных продуктов и методах ее обеспечения.
- 57) Методы обеспечения надежности на различных этапах жизненного цикла разработки программного продукта.
- 58) Прогнозирование ошибок.
- 59) Предотвращение ошибок.
- 60) Устранение ошибок.
- 61) Обеспечение отказоустойчивости.
- 62) Инструменты, обеспечивающие надежность программных продуктов.
- 63) План обеспечения надежности.
- 64) Назначение языка UML.
- 65) Общая структура языка UML.
- 66) Общие сведения о пакетах в языке UML.
- 67) Основные пакеты метамодели языка UML.
- 68) Специфика описания метамодели языка UML.
- 69) Особенности изображения диаграмм языка UML.
- 70) Описание часто используемых настольных клиентских технологий и сценариев, в которых разработчики их используют.
- 71) Определение области для миграции с Windows Forms на WPF.
- 72) Новые возможности в Visual Studio 2013 и WPF версии 4.
- 73) Разработка макета страницы и использование элементов управления.
- 74) Совместное использование ресурсов.
- 75) Способы совместного использования логических ресурсов в приложении.
- 76) Создание согласованного пользовательского интерфейса, используя стили.
- 77) Изменения внешнего вида элементов управления с помощью шаблонов.
- 78) Обработка событий и команд.
- 79) Введение в стратегии и пути модульного тестирования.
- 80) Отладка XAML с помощью визуализатора и отслеживание источников.
- 81) Разработка наборов тестовых данных.
- 82) Обеспечение обратной связи с пользователем для необработанных исключений.
- 83) Концепция и терминология привязки данных в WPF.
- 84) Привязка данных к элементам управления, реализация уведомления об изменении свойства.
- 85) Преобразование и проверка данных между источником и целью привязки.
- 86) Связывание коллекции объектов, сортировка, фильтрация и группировка дан-

ных с помощью представления коллекции.

87) Создание представления данных «мастер-детали и настройка отображения данных с помощью шаблонов данных».

88) Применение асинхронных процессов для реализации отзывчивого пользовательского интерфейса.

89) Функции локализации и глобализации, локализации WPF приложений.

90) Поддержка пользователей с ограниченными возможностями.

91) Создание двумерной графики, отображение в WPF приложения и добавление мультимедийного.

92) Разработка элементов управления.

93) Управление внешним видом элемента управления с помощью визуальных состояний.

94) Интеграция технологий WPF и Windows Forms.

95) Реализация присоединенных свойств.

96) Реализация в Expression Blend поведений, триггеров и действий.

97) Реализация в интерфейсе пользователя drag-and-drop.

98) Применение анимации в пользовательском интерфейсе с помощью триггеров.

99) Визуализация данных.

100) Введение в Документирование и сертификацию

101) Цели и основные принципы сертификации качества производственных предприятий и программных продуктов.

102) Системные требования, типы и источники дефектов и ошибок в комплексах программ.

103) Базовые стандарты сертификации управления производством программных продуктов.

104) Стандарты жизненного цикла программных средств для сертификации систем качества предприятий.

105) Подготовка производства программных средств и системы качества предприятия к сертификации.

106) Сертификация процессов производства программных продуктов и систем качества предприятий.

4.2 Контрольно-оценочные материалы для итоговой аттестации по профессиональному модулю «Участие в интеграции программных модулей»

1. ПАСПОРТ

Назначение:

КОС (контрольно-оценочные средства) предназначен для контроля и оценки результатов освоения ПМ 03 Участие в интеграции программных модулей по специальности СПО 09.02.03 «Программирование в компьютерных системах».

Умения

У 1 – владеть основными методологиями процессов разработки программного обеспечения;

У 2 – использовать методы для получения кода с заданной функциональностью и степенью качества.

Знания

З 1 – модели процесса разработки программного обеспечения;

З 2 – основные принципы процесса разработки программного обеспечения;

З 3 – основные подходы к интегрированию программных модулей;

З 4 – основные методы и средства эффективной разработки;

З 5 – основы верификации и аттестации программного обеспечения;

З 6 – концепции и реализации программных процессов;

З 7 – принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения;

З 8 – методы организации работы в коллективах разработчиков программного обеспечения;

З 9 – основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерения характеристик и параметров программ, программных систем и комплексов;

З 10 – стандарты качества программного обеспечения;

З 11 – методы и средства разработки программной документации.

Общие компетенции

ОК 1 - Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.

ОК 2 - Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.

ОК 3 - Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.

ОК 4 - Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.

ОК 5 - Использовать информационно-коммуникационные технологии в профессиональной деятельности.

ОК 6 - Работать в коллективе и команде, эффективно общаться с коллегами, руководством, потребителями.

ОК 7 - Брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий.

ОК 8 - Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.

ОК 9 - Ориентироваться в условиях частой смены технологий в профессиональной деятельности.

Профессиональные компетенции

ПК 3.1 – анализировать проектную и техническую документацию на уровне взаимодействия компонент программного обеспечения;

ПК 3.2 – выполнять интеграцию модулей в программную систему;

ПК 3.3 – выполнять отладку программного продукта с использованием специализированных программных средств;

ПК 3.4 – осуществлять разработку тестовых наборов и тестовых сценариев;

ПК 3.5 – производить инспектирование компонент программного продукта на предмет соответствия стандартам кодирования;

ПК 3.6 – разрабатывать технологическую документацию.

II. ЗАДАНИЕ ДЛЯ ЭКЗАМЕНУЮЩЕГОСЯ. Вариант № 1

Вариант 1

Инструкция для обучающихся

Внимательно прочитайте задание.

Время выполнения задания – 0,75 часа

Задание

Теоретическая часть:

1) В чем заключается сертификация процессов производства программных продуктов и систем качества предприятий?

2) Что такое спиральная модель?

3) В чем заключается парадигма Бейзили?

4) Каковы виды тестирования?

III. ПАКЕТ ЭКЗАМЕНАТОРА

III а. УСЛОВИЯ

Количество вариантов задания для экзаменуемого – 26.

Время выполнения задания – 0,75 часа.

Оборудование:

Компьютер.

Программное обеспечение:

- операционная система;
- файловый менеджер (в составе операционной системы или др.);
- электронные средства образовательного назначения:
 - Microsoft® Office Professional Plus 2007 Russian;
 - Microsoft® Office Standard 2013 Russian.

Эталоны ответов на теоретические вопросы:

1) В стандарте ISO 19011 сформулированы общие требования и рекомендации к оценке компетентности аудиторов производства и менеджмента систем качества продукции, которые целесообразно применять при сертификации программных средств. В них входят личные качества и способность применять знания и навыки при сертификационных испытаниях процессов производства.

Требования к квалификации аудиторов по сертификации производства и систем менеджмента качества предприятия:

- общие требования к личным характеристикам, к компетентности, знаниям и навыкам аудиторов производства и систем качества программных средств;
- требования к квалификации аудиторов по сертификации производства программных средств:
 - понимание принципов работы проверяемого предприятия и производства программных средств;
 - факторы, необходимые при выборе главных аудиторов и аудиторов для выполнения конкретных заданий сертификации производства программных средств;
 - обязанности главного аудитора качества производства программных средств;
 - обязанности аудиторов качества производства программных средств.

Для того, чтобы аудиторы имели возможность выбора и систематического проведения аудита надлежащим образом, они должны быть готовы к выполнению следующих действий:

- применению принципов, процедур и методов аудита;
- результативному планированию и организации работ;
- проведению аудита в течение установленного срока;
- сбору данных посредством результативного опроса, выслушивания, наблюдений и анализа документов, записей и данных;
- проверке точности собранных данных;
- подтверждению достаточности и приемлемости свидетельств аудита для подкрепления выводов аудита и заключений;
- оценке факторов, влияющих на достоверность выводов и заключений по результатам аудита;
- подготовке отчетов по аудиту;
- сохранению конфиденциальности данных.

Для применения критериев аудита, знания и навыки в этой области аудиторов должны охватывать:

- стандарты по системе менеджмента качества, применяемые процедуры или другие документы по системам менеджмента, используемые в качестве критериев аудита;
- применение ссылочных документов к различным ситуациям при аудите;
- системы информации и методы санкционирования доступа, обеспечения безопас-

ности, рассылки и управления документами, данными и записями.

Для понимания принципов работы проверяемого предприятия и производства знания и навыки аудиторов в этой области должны охватывать:

- размеры, структуру, функции предприятия и взаимосвязи подразделений внутри него;
- общие бизнес - процессы производства продукта и соответствующую терминологию;
- культурные и социальные обычаи проверяемого предприятия;
- применяемые законы, технические регламенты и другие требования, относящиеся к предмету производства;
- контракты и договоры предприятия;
- международные соглашения и конвенции.

Руководители аудиторских групп должны иметь дополнительные знания и навыки по руководству аудитом для результативного и эффективного его проведения. Руководитель аудиторской группы должен быть подготовлен к выполнению следующих действий:

- планированию аудита и результативное использование ресурсов в ходе аудита;
- представлению аудиторской группы при взаимодействии с заказчиком аудита и проверяемой организацией;
- организации и направлению работы членов аудиторской группы;
- руководству аудиторской группой для получения заключения по результатам аудита;
- предупреждению и разрешению конфликтов;
- подготовке и завершению отчета(акта) по аудиту.

Аудиторы производства и системы менеджмента качества должны обладать знаниями и навыками в следующих областях:

- принципы менеджмента качества программных средств и их применение,
- терминологию в отрасли производства программных продуктов;
- технические характеристики процессов и продукции производства;
- процессы и практику производства программных средств и в области экономики.

У аудиторов должны быть необходимое образование, опыт работы, обучения аудиту и опыт его проведения. Практический опыт работы должен быть в технической сфере, сфере управления или профессиональной области, включая опыт принятия решений, разрешения проблем и обмена информацией с другим управленческим или специальным персоналом, сотрудниками того же уровня, потребителями и/или другими заинтересованными сторонами.

При выборе главных аудиторов и аудиторов для выполнения конкретных заданий по сертификации производства программных средств необходимо, чтобы уровень их подготовки соответствовал следующим требованиям:

- знание состава и содержания профиля стандартов системы качества программных средств, в соответствии, с которыми должны быть проведены испытания;
- наличие профессиональных квалификационных оценок или технической экспертизы аудиторов в определенной прикладной области;
- достаточный размер и состав аудиторской группы;
- навыки управления группой аудиторов;
- личные качества, необходимые для общения с проверяемыми производствами и специалистами;
- отсутствие реального или предполагаемого столкновения интересов.

Службы управления Программами сертификации производства должна постоянно оценивать деятельность своих аудиторов и экспертов либо путем наблюдения за проведением испытаний, либо другими средствами. Получаемая информация должна использоваться для улучшения выбора аудиторов и их деятельности и для выявления их неправильных действий. Проверки, проводимые разными аудиторами, должны приводить к

аналогичным выводам, когда одна и та же процедура проверяется в одинаковых условиях. Должны быть установлены методы для оценки и сравнения деятельности аудиторов с целью достижения согласованности в их деятельности: семинары по обучению; сравнения деятельности аудиторов; ротация аудиторов из одной группы в другую.

Главный аудитор является, в конечном счете, ответственным за проведение всех этапов сертификационных испытаний. Следует, что-бы он обладал способностями и опытом в управленческой деятельности и имел достаточную компетенцию для принятия решений, связанных с проведением проверок и любых наблюдений. Кроме того, в обязанности главного аудитора входит:

- участие в выборе других членов группы для сертификации;
- составление Программы проведения проверок, подготовка рабочих документов и инструктирование бригады по сертификационным испытаниям процессов и/или продуктов;
- определение требований к деятельности, связанной с сертификацией включая квалификационные требования к аудиторам;
- оценка документов, относящихся к деятельности в области существующей системы качества с целью установления их адекватности производству;
- представление группы по испытаниям руководству со стороны проверяемого производства;
- информирование проверяемого производственного коллектива о наличии значительных несоответствий и о существенных затруднениях в ходе проверок системы качества;
- в четкой и ясной форме предоставление отчетов о результатах сертификационных испытаний.

Необходимо, чтобы аудиторы сохраняли беспристрастность и не были подвержены влияниям, которые могли бы неблагоприятно отразиться на их объективности. Следует, чтобы все лица и все организации, занимающиеся проверкой, уважали независимость и честное отношение к делу аудиторов, а также поддерживали их.

В обязанности аудиторов входит:

- передача и разъяснение проверяемым производственным коллективам требований к испытаниям;
- планирование и эффективное выполнение возложенных на них обязанностей;
- запись результатов наблюдений и подготовка отчетов о результатах испытаний;
- контроль за эффективностью выполнения корректирующих действий после проведения проверок;
- обеспечение сохранности документации, относящейся к испытаниям для ее предоставления по требованию, обеспечение ее конфиденциальности, обработка секретной информации с соблюдением требований инструкций;
- сотрудничество с главным аудитором и оказание помощи в его деятельности.

Необходимо, чтобы аудиторы: зарекомендовали себя объективными; собирали и анализировали достоверные и достаточно полные данные для подготовки выводов, относящихся к проверяемой системе качества; могли бы установить, что процедуры, документы и другая информация, описывающая или обеспечивающая требуемые компоненты системы качества, являются известными, понятными, имеющимися в распоряжении и используемыми коллективом проверяемого предприятия.

Управление программами сертификационных испытаний должно осуществляться главными аудиторами, обладающими практическими знаниями в области методов проверки конкретных систем качества производства программных средств. Служба управления сертификацией должна нанимать аудиторов, квалификация которых соответствует рекомендациям стандарта ISO 19011. Эти аудиторы должны быть утверждены оценочной комиссией (органом сертификации), приемлемой с точки зрения службы управления программами испытаний и соответствующей рекомендациям, представленным в этом стан-

дарте.

2) Для преодоления проблем, связанных с использованием многопроходной модели, в середине 1980-х годов была предложена спиральная модель жизненного цикла. Ее принципиальная особенность заключается в том, что прикладной ПП создается не сразу, как в случае каскадного подхода, а по частям с использованием метода прототипирования. Под прототипом понимается действующий программный компонент, реализующий отдельные функции и внешние интерфейсы разрабатываемого ПП. Создание прототипов осуществляется за несколько итераций, или витков спирали. Каждая итерация соответствует созданию фрагмента, или версии ПП, на ней уточняются цели и характеристики проекта, оценивается качество полученных результатов и планируются работы следующей итерации. На каждой итерации производится тщательная оценка риска превышения сроков и стоимости проекта с целью определения необходимости выполнения еще одной итерации, степени полноты и точности понимания требований к системе, а также целесообразности прекращения проекта. Спиральная модель избавляет пользователей и разработчиков ПП от полного и точного формулирования требований к системе на начальной стадии, поскольку они уточняются на каждой итерации. Таким образом, углубляются и последовательно конкретизируются детали проекта и в результате выбирается обоснованный вариант, который доводится до реализации.

Разработка итерациями отражает объективно существующий спиральный цикл создания системы, позволяя переходить на следующую стадию, не дожидаясь полного завершения работы на текущей стадии, поскольку при итеративном способе разработки недостающую работу можно выполнить на следующей итерации. Главная задача такой разработки – как можно быстрее показать пользователям системы работоспособный продукт, тем самым активизируя процесс уточнения и дополнения требований.

Спиральная модель не исключает использования каскадного подхода на завершающих стадиях проекта в тех случаях, когда требования к системе оказываются полностью определенными.

Основная проблема спирального цикла – определение момента перехода на следующую стадию. Для ее решения необходимо ввести временные ограничения на каждую из стадий жизненного цикла. Переход осуществляется в соответствии с планом, даже если не вся запланированная работа закончена. План составляется на основе статистических данных, полученных в предыдущих проектах, и личного опыта разработчиков.

Спиральная модель обладает следующими достоинствами:

- заказчик имеет возможность увидеть разрабатываемый ПП на ранних стадиях разработки;
- заказчики принимают активное участие в разработке ПП;
- в модели воплощены преимущества каскадной и многопроходной модели.

Недостатки спиральной модели:

- усложненная структура;
- Спираль может продолжаться до бесконечности, так как каждая ответная реакция заказчика может породить новый цикл.

В качестве модели жизненного цикла разработки программного продукта большое распространение получила улучшенная спиральная модель. В отличие от ранее рассмотренной спиральной модели эта модель использует каскадный подход на завершающих этапах разработки ПП.

Использование спиральной модели целесообразно, если существует хотя бы одна из следующих причин:

- целесообразно создание прототипа;
- организация обладает навыками, требуемыми для адаптации модели;
- требуется выполнять проекты со средней и высокой степенями риска;
- заказчики не уверены в своих потребностях;
- требования слишком сложные;

- проект очень большой.

3) Парадигма доктора Виктора Бейзили «цель, вопрос, метрика» является широко применяемым и хорошо зарекомендовавшим себя подходом к определению метрик, наиболее подходящим для контроля над проектом и формирования решения. Подход с применением GQM предполагает, что цель устанавливается еще до выбора метрик. В. Бейзили разработал методику, в основе которой лежит процесс, состоящий из семи этапов. Благодаря этому появилась возможность систематического применения измерений при разработке ПП. Первые три из них имеют определяющее значение для работы в рамках парадигмы GQM.

Этап 1 – Определение набора целей. Под этим понимается формирование набора корпоративных целей, относящихся к работе подразделения, или проектных целей, связанных с разработкой, сопровождением, повышением производительности и качественных показателей продукта или процесса. Часто определение целей выполняется при решении проблем методом «мозгового штурма», а также в процессе получения требований от каждого из заинтересованных лиц. Эти цели можно располагать с учетом приоритетов, а также группировать их по областям усовершенствований ПП.

Этап 2 – Формирование набора вопросов, характеризующих цели. Каждая цель позволяет сформулировать вопрос, ответ на который определяет, будет ли эта цель достигнута. Данные вопросы характеризуют цели, позволяют оценить прогресс в их достижении, прогнозируют момент достижения или являются мотивацией для оценки продвижения к цели.

Этап 3 – Определение метрических показателей, необходимых для ответа на вопросы. На этом этапе определяют, что необходимо измерить для получения адекватного ответа на поставленные вопросы.

Этап 4 – Разработка механизмов сбора данных.

Этап 5 – Сбор, подтверждение и анализ данных в реальном времени для поддержки обратной связи между корректирующими действиями и проектами.

Этап 6 – Анализ данных с использованием подпрограммы для оценки соответствия целям и рекомендации для дальнейшего совершенствования.

Этап 7 – Поддержка обратной связи организатора проекта с его участниками.

4) Модульно тестирование. Этот вид тестирования представляет собой процесс проверки отдельных программ процедур и подпрограмм, входящих в состав программ или программных систем. Модульное тестирование производится непосредственным разработчиком и позволяет проверять все внутренние структуры и потоки данных в каждом модуле. Этот вид тестирования является частью этапа разработки. При модульном тестировании выполняется набор тестов, определяемый разработчиком так, чтобы охват тестированием каждого модуля был не менее 70...75%.

Элементами модульного тестирования являются:

- синтаксическая проверка – проверка с использованием некоторого инструментального средства для выявления синтаксических ошибок в программном коде;
- проверка соответствия стандартам кодирования – проверка кода на соответствие стандартам кодирования компании;
- технический обзор программного кода.

После успешного завершения модульного тестирования все измененные модули и наборы тестов сохраняются в базе данных проекта.

Интеграционное тестирование. Этот вид тестирования проводится для проверки совместной работы отдельных модулей и предшествует тестированию всей системы как единого целого. В ходе интеграционного тестирования проверяются связи между модулями, их совместимость и функциональность. Оно осуществляется независимым тестировщиком и входит в состав этапа тестирования.

Элементами интеграционного тестирования являются:

- проверка функциональности – проверка соответствия отдельных функций, вы-

полняемых совокупностями модулей, функциям, заданным в спецификациях требований;

- проверка промежуточных результатов – проверка всех промежуточных результатов и файлов на наличие и корректность;
- проверка интеграции – проверка того, что модули передают друг другу информацию корректно.

Ошибки, выявленные в ходе интеграционного тестирования, заносятся в базу данных ошибок. Результаты интеграционного тестирования включаются в отчет о ходе тестирования при завершении цикла тестирования.

Системное тестирование. Этот вид тестирования предназначен для проверки программной системы в целом, ее организации и функционирования на соответствие спецификациям требований заказчика. Его проводит независимый тестировщик после успешного интеграционного тестирования.

Элементами системного тестирования являются:

- граничное тестирование – тестирование в граничных условиях;
- прогоночное тестирование – тестирование всех функциональных характеристик реальной работы системы;
- целевое тестирование – тестирование на целевой платформе;
- проверка документации – проверка пользовательской документации на корректность;
- другие тесты, определяемые тестировщиком.

Ошибки, выявленные при системном тестировании, заносятся в базу данных проекта. Результаты системного тестирования в отчет о ходе тестирования.

Выходное тестирование. Это завершающий этап тестирования, на котором проверяется готовность ПП к поставке заказчику. Данный вид тестирования проводит независимый тестировщик. Элементами выходного тестирования являются:

- проверка инсталляции – проверка на ясность и корректность инструкций по инсталляции;
- проверка документации – проверка того, что вся необходимая документация полностью подготовлена и готова к передаче заказчику.

Ошибки, выявленные при выходном тестировании, заносятся в базу данных проекта. При успешном завершении выходного тестирования ПП поставляется заказчику вместе с отчетом о результатах тестирования.

Приемочное тестирование. Этот вид тестирования проводится организацией, отвечающей за инсталляцию, сопровождение программной системы и обучение конечного пользователя.

Экзаменационная ведомость.

III.6. КРИТЕРИИ ОЦЕНКИ

Критерии оценки:

Отметка «5» - выставляется в том случае, когда в ответе обучающегося полно и верно раскрыто основное содержание вопроса, соблюдена логическая последовательность элементов ответа; общие положения конкретизируются фактами, обосновываются аргументами.

Отметка «4» - выставляется в том случае, когда в ответе обучающегося содержится верное освещение темы вопроса, но отсутствует полнота раскрытия; соблюдена логика изложения.

Отметка «3» - выставляется в том случае, когда в ответе обучающегося приведены отдельные несистематизированные положения, отсутствует конкретизация их фактами или частично приведены отдельные верные факты

Образец экзаменационного билета

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Комсомольский-на-Амуре государственный университет»

20__ / 20__ учебный год

Экзамен
по ПМ 03 Участие в интеграции программных модулей
Билет № 1

ЗАДАНИЕ ДЛЯ ЭКЗАМЕНУЮЩЕГОСЯ

Оцениваемые компетенции: ОК1-9, ПК3.1-3.6

Вариант № 1

Теоретическая часть:

- 1) В чем заключается сертификация процессов производства про-граммных продуктов и систем качества предприятий?
 - 2) Что такое спиральная модель?
 - 3) В чем заключается парадигма Бейзили?
 - 4) Каковы виды тестирования?
- Вы можете воспользоваться конспектом лекций.
Максимальное время выполнения задания – 45 мин.

Утвержден на заседании кафедры «_____»,
протокол № ____ от «___» _____ 20__ г.

Зав. кафедрой «_____» _____ В.А. Тихомиров
(подпись)

5 Критерии и шкалы для интегрированной оценки уровня сформированности компетенций

Индикаторы компетенции	неудовлетворительно	удовлетворительно	хорошо	отлично
Полнота знаний	Уровень знаний ниже минимальных требований. Имели место грубые ошибки.	Минимально допустимый уровень знаний. Допущено много негрубых ошибок.	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки, без ошибок.
Наличие умений	При решении стандартных задач не продемонстрированы основные умения. Имели место грубые ошибки.	Продemonстрированы основные умения. Решены типовые задачи с негрубыми ошибками. Выполнены все задания но не в полном объеме.	Продemonстрированы все основные умения. Решены все основные задачи с негрубыми ошибками. Выполнены все задания, в полном объеме, но некоторые с недочетами.	Продemonстрированы все основные умения, решены все основные задачи с отдельными незначительными недочетами, выполнены все задания в полном объеме.
Характеристика сформированности компетенции	Компетенция в полной мере не сформирована. Имеющихся знаний, умений, навыков недостаточно для решения практических (профессиональных) задач. Требуется повторное обучение	Сформированность компетенции соответствует минимальным требованиям. Имеющихся знаний, умений, навыков в целом достаточно для решения практических (профессиональных) задач, но требуется дополнительная практика по большинству практических задач.	Сформированность компетенции в целом соответствует требованиям, но есть недочеты. Имеющихся знаний, умений, навыков и мотивации в целом достаточно для решения практических (профессиональных) задач, но требуется дополнительная практика по некоторым профессиональным задачам.	Сформированность компетенции полностью соответствует требованиям. Имеющихся знаний, умений, навыков и мотивации в полной мере достаточно для решения сложных практических (профессиональных) задач.
Уровень сформированности компетенций	Низкий	Ниже среднего	Средний	Высокий

6 Список рекомендуемой литературы и источников информации

Основная литература:

1 Исаченко О.В. Программное обеспечение компьютерных сетей [Электронный ресурс]: учебное пособие / О.В. Исаченко. - М.: НИЦ ИНФРА-М, 2018. - 117 с. //ZNANIUM.COM: электронно-библиотечная система. Режим доступа: <http://www.znanium.com/catalog.php>, ограниченный. – Загл. с экрана.

2 Федорова Г.Н. Разработка, внедрение и адаптация программного обеспечения отраслевой направленности [Электронный ресурс]: учебное пособие для сред. проф. образования / Федорова Г.Н. - М.:КУРС, НИЦ ИНФРА-М, 2018. - 336 с. //ZNANIUM.COM: электронно-библиотечная система. – Режим доступа: <http://www.znanium.com/catalog.php>, ограниченный. – Загл. с экрана.

3 Голицына, О.Л. Языки программирования [Электронный ресурс]: учеб. пособие для сред. проф. образования / О.Л. Голицына, Т.Л. Партыка, И.И. Попов. – М.: ФОРУМ: ИНФРА-М, 2018. – 399 с. //ZNANIUM.COM: электронно-библиотечная система. Режим доступа: <http://www.znanium.com/catalog.php>, ограниченный. – Загл. с экрана.

4 Липаев, В.В. Документирование сложных программных комплексов [Электронный ресурс]: электронное дополнение к учебному пособию «Программная инженерия сложных заказных программных продуктов» / В.В. Липаев. – Саратов: Вузовское образование, 2015. – 115 с. //IPRbooks: электронно-библиотечная система. Режим доступа: <http://www.iprbookshoop.ru/27294.html>, ограниченный. – Загл. с экрана.

5 Вичугова А.А. Инструментальные средства разработки компьютерных систем и комплексов [Электронный ресурс]: учебное пособие для сред. проф. образования / А.А. Вичугова. — Саратов: Профобразование, 2017. — 135 с. //IPRbooks: электронно-библиотечная система. – Режим доступа: <http://www.iprbookshoop.ru/66387.html>, ограниченный. – Загл. с экрана.

6 Синявская, С.В. Стандартизация и сертификация радиоэлектронной и вычислительной техники [Электронный ресурс]: учебное пособие / С.В. Синявская. – Минск: РИПО, 2015. – 324 с. //IPRbooks: электронно-библиотечная система. Режим доступа: <http://www.iprbookshoop.ru/67741.html>, ограниченный. – Загл. с экрана.

7 Ананьева, Т.Н. Стандартизация, сертификация и управление качеством программного обеспечения [Электронный ресурс]: учебное пособие / С.В. Синявская. – Минск: РИПО, 2015. – 324 с. //IPRbooks: электронно-библиотечная система. Режим доступа: <http://www.iprbookshoop.ru/67741.html>, ограниченный. – Загл. с экрана.

8 Терехов, А.Н. Технология программирования [Электронный ресурс]: учебное пособие / А.Н. Терехов. – Москва, Саратов: ИНТУИТ, 2017. – 152 с. //IPRbookshoop.ru/67370.html, ограниченный. – Загл. с экрана.

Дополнительная литература:

1 Гагарина Л. Г. Технология разработки программного обеспечения: Учеб. пос. / Л.Г. Гагарина, Е.В. Кокорева, Б.Д. Виснадул; Под ред. проф. Л.Г. Гагариной - М.: ИД ФОРУМ: НИЦ Инфра-М, 2013. - 400 с.: ил.- (Высшее обр.) [Электронный ресурс] Режим доступа: www.znanium.com/ ЭБС ИНФРА-М

2 Рудаков А.В. Технология разработки программных продуктов [Текст]: учебник. - М.: - ОИЦ «Академия», 2013-208с.

3 Рудаков А.В. Технология разработки программных продуктов [Текст]: практикум. - М.: - ОИЦ «Академия», 2014-192с.

4 Вичугова А.А. Инструментальные средства информационных систем: Учебное пособие / Вичугова А.А. - Томск: Изд-во Томского политех. университета, 2015. - 136 с. [Электронный ресурс] Режим доступа: www.znanium.com/ ЭБС ИНФРА-М

5 Дубовой Н.Д. Основы метрологии, стандартизации и сертификации: Учебное пособие / Н.Д. Дубовой, Е.М. Портнов. - М.: ИД ФОРУМ: НИЦ ИНФРА-М, 2014. - 256 с.: ил. (Профессиональное образование) [Электронный ресурс] Режим доступа:

6 ГОСТ 24.103-84. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Автоматизированные системы управления. Общие положения

7 ГОСТ 24.104-85 Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Автоматизированные системы управления. Общие требования

8 ГОСТ 24.202-80. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документа «Технико-экономическое обоснование»

9 ГОСТ 24.203-80 Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию общесистемных документов

10 ГОСТ 24.204-80. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документа «Описание постановки задачи»

11 ГОСТ 24.205-80 Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов по информационному обеспечению

12 ГОСТ 24.206-80 Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов по техническому обеспечению

13 ГОСТ 24.207-80 Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов по программному обеспечению

14 ГОСТ 24.208-80 Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов стадии «Ввод в эксплуатацию»

15 ГОСТ 24.209-80 Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов по организационному обеспечению

16 ГОСТ 24.210-82 Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов по функциональной части

17 ГОСТ 24.211-82 Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документа «Описание алгоритма»

18 ГОСТ 24.301-80 Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Общие требования к выполнению текстовых документов

19 ГОСТ 24.302-80 Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Общие требования к выполнению схем

20 ГОСТ 24.304-82 Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к выполнению чертежей

21 ГОСТ 24.703-85 Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Типовые проектные решения. Основные положения

22 ГОСТ 34.201-89. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Виды, комплектность и обозначение документов при создании автоматизированных систем

23 ГОСТ 34.320- 96 Информационные технологии. Система стандартов по базам данных. Концепции и терминология для концептуальной схемы и информационной базы

24 ГОСТ 34.321- 96 Информационные технологии. Система стандартов по базам данных. Эталонная модель управления данными

25 ГОСТ 34.601 – 90 Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания.

26 ГОСТ 34.602-89. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Техническое задание на создание автоматизированной системы

27 ГОСТ 34.603-92. Комплекс стандартов на автоматизированные системы. Авто-

матризованные системы. Виды испытаний автоматизированных систем

28 ГОСТ 6.01.1-87. Единая система классификации и кодирования технико-экономической информации

29 Стандарт ISO/IEC 12207:1995 «Information Technology — Software Life Cycle Processes» (информационные технологии – жизненный цикл программного обеспечения), ГОСТ Р ИСО/МЭК 12207-99.

30 ГОСТ Р ИСО/МЭК 15288-2005. Системная инженерия. Процессы жизненного цикла систем

31 ГОСТ Р ИСО/МЭК ТО 16326-2002. Программная инженерия. Руководство по применению ГОСТ Р ИСО/МЭК 12207 при управлении проектом

32 ISO 10014. Управление качеством — Указания по получению финансовых и экономических выгод.

33 Гвоздева Т.В., Баллод Б.А. Проектирование информационных систем. Серия Высшее образование. М.: Феникс, 2009. – 512 с.

34 Грекул В.И., Денищенко Г.Н., Коровкина Н.Л. Управление внедрением информационных систем М.: БИНОМ. Лаборатория знаний, Интернет-университет информационных технологий - ИНТУИТ.ру, 2008. – 200 с.

35 Грекул В.И., Денищенко Г.Н., Коровкина Н.Л. Проектирование информационных систем Интернет-университет информационных технологий -2-е изд. – М.: Бином. Лаборатория знаний Интуит Серия: Основы информационных технологий, 2008. – 300 с.

36 Лодон Дж., Лодон К. Управление информационными системами. Спб.: Питер, 2005. – 280 с.

37 Соловьев И.В., Майоров А.А. Проектирование информационных систем. М.: Академический проект, 2009. – 400 с.

38 Арлоу Дж., Нейштадт А. UML 2 и Унифицированный процесс: Практический объектно-ориентированный анализ и проектирование (пер. с англ. Шатохиной Н.). 2-е изд., М.: Символ Плюс, 2007. – 624 с.

39 Буч Г. Коналлен Д. Максимчук Р.А. Хьюстон К. Энгл М. Янг Б. Объектно-ориентированный анализ и проектирование с примерами приложений. – 3-е изд. М.: Вильямс, 2008. – 720 с.

40 Васильев Р. Б., Калянов Г. Н и др. Управление развитием информационных систем. – М.: Горячая Линия-Телеком, 2009 – 350 с.

41 Данилин А., Слюсаренко А. Архитектура и стратегия. "Инь" и "янь" информационных технологий Интернет-университет информационных технологий – ИНТУИТ.ру, 2009. – 506 с.

42 Информационные системы в экономике. Под ред. Титоренко Г.А. 2-е изд., перераб. и доп. - М.: Юнити-Дана, 2008. — 463 с.

43 Информационные системы и технологии в экономике и управлении. 3- изд.: Учебник / Под ред. проф. В.В. Трофимова. – М.: Высшее образование, 2009. – 528 с.

44 Мельников В. Защита информации в компьютерных системах. – М.: Финансы и статистика, Электроинформ, 2007.

45 Пирогов В.Ю. Информационные системы и базы данных. Организация и проектирование. Серия Учебная литература для вузов. СПб.: БХВ-Петербург, 2009. – 528 с.

46 Зверева В.П. Технические средства информатизации: учебник / В.П. Зверева, А.В. Назаров. – М.: КУРС: ИНФРА-М, 2017. – 248 с.

47 Федорова Г.Н. Разработка, внедрение и адаптация программного обеспечения отраслевой направленности: Учебное пособие. / Федорова Г.Н. - М.:КУРС, НИЦ ИНФРА-М, 2018. - 336 с

48 Шишмарев В.Ю. Метрология, стандартизация, сертификация, техническое регулирование и документооборот: Учебник / В.Ю. Шишмарев. — М.: КУРС: ИНФРА-М, 2017. — 312 с.

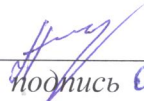
Лист изменений и дополнений

в комплект контрольно-оценочных средств учебного модуля «Участие в интеграции программных модулей» по специальности 09.02.03 «Программирование в компьютерных системах»

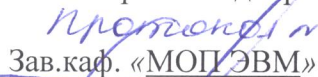
<i>№ изменения, дата изменения; номер страницы с изменением</i>	
БЫЛО	СТАЛО
1. Министерство образования и науки Российской Федерации – стр.1. 2. «Комсомольский-на-Амуре государственный технический университет» - стр.1	1. Министерство науки и высшего образования Российской Федерации – стр.1. 2. «Комсомольский-на-Амуре государственный университет» - стр.1. 3. Исаченко О.В. Программное обеспечение компьютерных сетей [Электронный ресурс]: учебное пособие / О.В. Исаченко. - М.: НИЦ ИНФРА-М, 2018. - 117 с. //ZNANIUM.COM: электронно-библиотечная система. Режим доступа: http://www.znanium.com/catalog.php, ограниченный. – Загл. с экрана. – стр.49. 4. Федорова Г.Н. Разработка, внедрение и адаптация программного обеспечения отраслевой направленности [Электронный ресурс]: учебное пособие для сред. проф. образования / Федорова Г.Н. - М.:КУРС, НИЦ ИНФРА-М, 2018. - 336 с. //ZNANIUM.COM: электронно-библиотечная система. – Режим доступа: http://www.znanium.com/catalog.php, ограниченный. – Загл. с экрана. – стр.49. 5. Голицына, О.Л. Языки программирования [Электронный ресурс]: учеб. пособие для сред. проф. образования / О.Л. Голицына, Т.Л. Партыка, И.И. Попов. – М.: ФОРУМ: ИНФРА-М, 2018. – 399 с. //ZNANIUM.COM: электронно-библиотечная система. Режим доступа: http://www.znanium.com/catalog.php, ограниченный. – Загл. с экрана. – стр.49. 6. 47 Федорова Г.Н. Разработка, внедрение и адаптация программного обеспечения отраслевой направленности: Учебное пособие. / Федорова Г.Н. - М.:КУРС, НИЦ ИНФРА-М, 2018. - 336 с – стр.51.

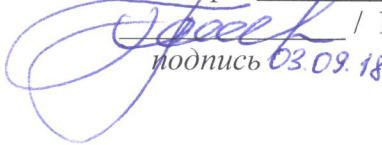
Основание:

1. Постановление Правительства РФ от 15.06.2018 №682 «Об утверждении Положения о Министерстве науки и высшего образования Российской Федерации и признании утратившими силу некоторых актов Правительства Российской Федерации.
2. Приказ Минобрнауки России от 3 октября 2017г. № 997 «О переименовании федерального государственного бюджетного образовательного учреждения высшего образования «Комсомольский-на-Амуре государственный технический университет» и его филиала и о внесении изменений в устав федерального государственного бюджетного образовательного учреждения высшего образования «Комсомольский-на-Амуре государственный технический университет»
3. Приобретение новой литературы.

 / О.Н. Попенко
подпись 03.09.18 Инициалы, фамилия внесшего изменения

Рассмотрено и одобрено на заседании кафедры «МОП ЭВМ»

 / В.А. Тихомиров/
Зав.каф. «МОП ЭВМ»

 / В.А. Тихомиров/
подпись 03.09.18 Инициалы, фамилия

Лист изменений и дополнений

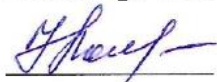
в комплект контрольно-оценочных средств учебного модуля
«Участие в интеграции программных модулей»
по специальности среднего профессионального образования
09.02.03 «Программирование в компьютерных системах»
на 2021-2022 учебный год

<i>№ изменения, дата изменения</i>
1 Титульный лист, изменено «Факультет довузовской подготовки» на «Колледж» Основание: Приказ ректора университета № 421-«О» от 30.11.2020 «О создании Колледжа»

Рассмотрено и одобрено на заседании кафедры «Общепрофессиональных и специальных дисциплин»

Протокол № 9 «05» мая 2021 г.

Зав.каф. «Общепрофессиональных и специальных дисциплин»



/ Н.С. Ломакина

РЕЦЕНЗИЯ

на комплект оценочных средств по профессиональному модулю ПМ.03 «Участие в интеграции программных модулей»,

составленные преподавателем КНАГТУ Попенко Ольгой Николаевной

Представленный на рецензию комплект оценочных средств по профессиональному модулю ПМ.03 «Участие в интеграции программных модулей» разработан в соответствии с требованиями Федерального государственного образовательного стандарта среднего профессионального образования для специальности 09.02.03 «Программирование в компьютерных системах») для очной формы обучения.

В структуре комплекта оценочных средств представлены следующие элементы: паспорт комплекта оценочных средств; комплект оценочных средств для проведения текущего контроля и промежуточной аттестации по междисциплинарным курсам МДК.03.01 «Технология разработки программного обеспечения», МДК.03.02 «Инструментальные средства разработки программного обеспечения», МДК.03.03 «Документирование и сертификация»; комплект оценочных средств для проведения промежуточной аттестации по профессиональному модулю.

Представленные оценочные средства по ПМ.03 позволяют стимулировать познавательную активность обучающихся за счет разнообразных форм заданий, разного уровня сложности, наличия материалов само- и взаимооценки.

Задания, представленные в комплекте оценочных средств, максимально приближены к условиям будущей профессиональной деятельности обучающихся. В программу квалификационного экзамена для контроля и оценки результатов освоения модуля ПМ.03 включено 106 экзаменационных вопросов.

Представленный комплект оценочных средств, по нашему мнению, позволяет объективно оценить уровень знаний, умений, сформированность практического опыта, общих и профессиональных компетенций обучающихся и их соответствие требованиям ФГОС СПО по данной специальности.

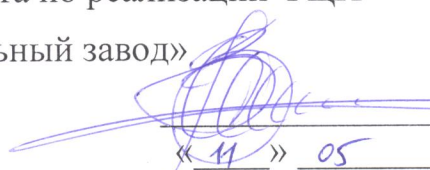
Рецензируемый комплект оценочных средств рекомендуется для использования в качестве диагностического инструментария при реализации профессионального модуля ПМ.01 «Участие в интеграции программных модулей».

Рецензент:

Рецензент: руководитель проекта по реализации ФЦП

ПАО «Амурский судостроительный завод»



 В.А. Ханов
« 11 » 05 2017 г.