

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Белгородский государственный технологический университет
им. В. Г. Шухова

СТАНДАРТИЗАЦИЯ И ЛИЦЕНЗИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Методические указания к выполнению лабораторных работ
и индивидуальных домашних заданий для студентов всех форм
обучения направлений подготовки 09.03.02 – Информационные
системы и технологии, 09.03.03 – Прикладная информатика

Белгород
2018

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Белгородский государственный технологический университет
им. В. Г. Шухова

Кафедра информационных технологий

Утверждено
научно-методическим советом
университета

СТАНДАРТИЗАЦИЯ И ЛИЦЕНЗИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Методические указания к выполнению лабораторных работ
и индивидуальных домашних заданий для студентов всех форм
обучения направлений подготовки 09.03.02 – Информационные
системы и технологии, 09.03.03 – Прикладная информатика

Белгород
2018

УДК 004:006(07)

ББК 30ця7

С764

Составитель канд. техн. наук, доц. *М.А. Косоногова*

Рецензент канд. техн. наук, доц. *Е.С. Черноситова*

С764 Стандартизация и лицензирование программного обеспечения: метод. указания к выполнению лабораторных работ и индивидуальных домашних заданий / сост. М.А. Косоногова. - Белгород: Изд-во БГТУ, 2018. – 40 с.

В методические указания включен материал по основным стандартам, используемым для упорядочения процессов жизненного цикла программной продукции. Издание содержит требования, рекомендации и задания для выполнения лабораторных работ и ИДЗ по дисциплине «Стандартизация и лицензирование программного обеспечения».

Методические указания предназначены для студентов всех форм обучения направлений подготовки 09.03.02 – Информационные системы и технологии, 09.03.03 – Прикладная информатика.

Издание публикуется в авторской редакции.

УДК 004:006(07)

ББК 30ця7

©Белгородский государственный
технологический университет
(БГТУ) им. В.Г. Шухова, 2018

ВВЕДЕНИЕ

Методические указания охватывают дисциплину «Стандартизация и лицензирование программного обеспечения». Дисциплина входит в состав федеральной компоненты цикла профессиональных дисциплин направлений бакалавриата «Информационные системы и технологии» и «Прикладная информатика». Целью изучения данной дисциплины является ознакомление с ролью и местом стандартов в организации жизненного цикла информационных систем и программной продукции. Организовать коллективную работу, спроектировать систему и ее интерфейс, подготовить программную документацию, реализовать и протестировать, оценить качество программной продукции и процесса ее разработки – использование стандартов позволяет внести упорядоченность и согласованность в эти процессы. Выполнение студентами лабораторных работ базируется на адаптации действующих стандартов к их программным продуктам, разработанным в рамках курсового проектирования. В методических целях приводятся шаблоны выполнения и оформления лабораторной работы, а также контрольные вопросы для ее защиты. Издание также содержит индивидуальное домашнее задание, которое предполагает использование стандартов документирования программной продукции.

Задачи по внедрению стандартов в деятельность ИТ-компании возникают тогда, когда руководство осознает потребность в повышении зрелости компании с точки зрения способности производить качественную программную продукцию. В этом случае использование стандартов вносит упорядоченность и согласованность на множестве процессов жизненного цикла информационных систем и программных продуктов. Без стандартов, устанавливающих характеристики и метрики качества программной продукции, невозможно перейти к точному планированию проекта и контролю качества его результатов. Грамотное внедрение стандартов в деятельность ИТ-компании может обеспечить:

- повышение конкурентоспособности ИТ-продукции (работ, услуг);
- улучшение технико-экономических показателей проектов программных средств;
- получение дополнительных экономических преимуществ (увеличение тиражей продукции и длительности жизненного цикла);
- обеспечение переносимости и масштабируемости прикладных программ.

Все лабораторные работы имеют относительно самостоятельный сюжет и располагаются в последовательности, соответствующей основным процессам жизненного цикла ИТ-продукции. Итогом лабораторных работ должен стать пакет отчетов, разработанных на основе стандартов в области ИТ и ИКТ и описывающих разные стадии разработки: общая структура жизненного цикла, оценка качества и тестирование программной продукции, управление программными активами.

Индивидуальное домашнее задание дополняет комплекс лабораторных работ и позволяет приобрести навыки документирования программной продукции и процесса ее разработки.

Все отчёты о выполнении работ оформляются в унифицированном печатном виде на листах формата А4.

Каждый отчёт должен содержать:

1. Заголовок лабораторной работы.
2. Цель работы.
3. Задание к лабораторной работе.
4. Ход выполнения работы и полученные результаты.
5. Выводы по сделанной работе.

Лабораторная работа №1

ЗНАКОМСТВО С ИНФОРМАЦИОННО-ПОИСКОВЫМИ СИСТЕМАМИ НОРМАТИВОВ

Цель работы: получение практических навыков по начальной работе с системами поиска стандартов; знакомство с интерфейсом специализированной информационно-поисковой системы стандартов NormaCS и типовой структурой стандартов.

Задания к лабораторной работе №1

Задания выполняются в информационно-поисковой системе нормативов NormaCS. При описании хода выполнения работы указывайте, с помощью каких инструментов NormaCS выполнялось соответствующее задание.

В процессе выполнения работы используйте руководство пользователя NormaCS и презентации по этой системе (входят в комплект лабораторных работ). Для работы дома подойдет сетевая lite-версия, которую можно скачать с официального сайта по адресу <http://www.normacs.ru/lite.jsp>.

1. Приведите сведения об основных элементах интерфейса системы NormaCS (в описание можно включить скриншоты).
2. Занесите в отчет полное наименование стандартов в сфере информационных технологий следующих категорий и ссылки на них: ГОСТ, ГОСТ Р, ОСТ, ГОСТ Р ИСО/МЭК. По каждой категории достаточно привести 2-3 документа.

Пример описания хода работы: Для поиска стандартов категории ГОСТ в сфере ИТ на вкладке «Поиск» были указаны следующие параметры: «Индекс» - ГОСТ, «Область поиска» - Классификатор ISO → Информационные технологии. Машины конторские. Результаты поиска по сформированному запросу:

- 2.1. ГОСТ 28470-90 Система технического обслуживания и ремонта технических средств вычислительной техники и информатики. Виды и методы технического обслуживания и ремонта ([normacs://normacs.ru/SIU?dob=42736.000347&dol=42765.596030](http://normacs.ru/SIU?dob=42736.000347&dol=42765.596030)).
- 2.2. ...
3. Стандарты, найденные при выполнении п.2, добавьте в панель избранных документов в NormaCS.

4. Найдите формулировки терминов согласно варианту (см. табл.1). Результаты оформите в виде таблицы со столбцами «Термин», «Определение», «Источник»*, «Предметная область»**.

* полное наименование нормативного документа, из которого взято определение;

** сфера деятельности, в которой данная трактовка термина уместна.

Таблица 1. Поиск терминов

№ вар.	Термин
1.	Национальная стандартизация; Вредоносная программа
2.	Стандартизация; Программа для ЭВМ
3.	Технический комитет (по стандартизации); Программа для ЭВМ с открытым кодом
4.	Уровень стандартизации; Система управления базами данных
5.	Документ по стандартизации; Язык программирования
6.	Международная стандартизация; Компиляция
7.	Национальный стандарт; Верификация программы
8.	Предварительный национальный стандарт; Операционная система
9.	Официальное опубликование; Мобильная программа
10.	ГОСТ; Программа вычислительной машины
11.	Область стандартизации; Библиотека программ
12.	Национальный орган Российской Федерации по стандартизации; Реляционная модель данных
13.	Уровень стандартизации; Генератор программ
14.	Стандартизация; Жизненный цикл программного обеспечения
15.	Нормативная база качества услуги; Исходная программа
16.	Официальное опубликование; Канал связи
17.	Национальная стандартизация; Локальная сеть
18.	Аттестация качества продукции; Магистральный кабель (локальной вычислительной сети)
19.	Предварительный национальный стандарт; Шлюз (локальной вычислительной сети)
20.	Предварительная информация о стандарте; Обработка информации

5. Заполните таблицу реквизитов (см. табл.2) для стандарта согласно варианту (см. табл.3). Всю нужную информацию можно получить из карточки документа.

При описании хода выполнения работы предварительно укажите, по каким параметрам Вы нашли стандарт.

Таблица 2. Структура информации о стандарте

Реквизит	Описание
Название	
Индекс (категория)	
Код	

Реквизит	Описание
Статус	
Кем утвержден	
Номер нормативного документа об утверждении	
Дата утверждения	
Дата ввода в действие	
Область применения	
Исходящие ссылки (документ ссылается на...)	
Входящие ссылки (на документ ссылаются...)	
Взамен	
Чем заменен	
Оглавление	
Изменения	
Публикации	
Классификаторы	
Комментарий	

Таблица 3. Примеры стандартов из области ИТ

№ вар.	Полное наименование стандарта
1.	ГОСТ ИСО/МЭК 15459-2-2008 Автоматическая идентификация. Идентификаторы уникальные международные. Часть 2. Порядок регистрации
2.	ГОСТ Р 34.31-96 Информационная технология. Микропроцессорные системы. Интерфейс Фьючебас+. Спецификации физического уровня
3.	ГОСТ Р 53626-2009 Информационно-коммуникационные технологии в образовании. Технические средства обучения. Общие положения
4.	ГОСТ Р 53115-2008 Защита информации. Испытание технических средств обработки информации на соответствие требованиям защищенности от несанкционированного доступа. Методы и средства
5.	ГОСТ Р ИСО 21549-5-2010 Информатизация здоровья. Структура данных на пластиковой карте пациента. Часть 5. Идентификационные данные
6.	ГОСТ 19.301-79 Единая система программной документации. Программа и методика испытаний. Требования к содержанию и оформлению
7.	ГОСТ 14289-88 Средства вычислительной техники. Клавиатуры. Расположение клавиш и символов, функции управляющих клавиш
8.	ГОСТ 19.402-78 Единая система программной документации. Описание программы
9.	ГОСТ 23411-84 Машины электронные контрольно-регистрирующие. Общие технические условия
10.	ГОСТ 26765.52-87 Интерфейс магистральный последовательный системы электронных модулей. Общие требования
11.	ГОСТ Р ИСО 10303-21-99 Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Часть 21. Методы реализации. Кодирование открытым текстом структуры обмена
12.	ГОСТ 34.961-91 Системы обработки информации. Взаимосвязь открытых систем. Протокол транспортного уровня в режиме с установлением соединения
13.	ГОСТ ISO/IEC 15420-2010 Автоматическая идентификация. Кодирование штриховое. Спецификация символики штрихового кода EAN/UPC
14.	IEC/TR 60825-5:2003 Безопасность лазерной аппаратуры. Часть 5. Контрольный

№ вар.	Полное наименование стандарта
	перечень к МЭК 60825-1 для изготовителей
15.	ГОСТ Р ИСО/ТО 8509-95 Системы обработки информации. Взаимосвязь открытых систем. Соглашения по услугам
16.	ГОСТ Р 43.4.1-2011 Информационное обеспечение техники и операторской деятельности. Система «человек-информация»
17.	ГОСТ Р ИСО/МЭК 14764-2002 Информационная технология. Сопровождение программных средств
18.	ГОСТ Р ИСО/МЭК 15910-2002 Информационная технология. Процесс создания документации пользователя программного средства
19.	ГОСТ Р ИСО/МЭК 17826-2015 Информационные технологии. Интерфейс управления облачными данными (CDMI)
20.	ГОСТ Р ИСО/МЭК ТО 10032-2007 Эталонная модель управления данными

6. Освойте поиск стандартов по условиям с помощью панели расширенного поиска. Изучите список полей, доступных при расширенном поиске. Найдите документ(ы) в соответствии с поисковым запросом (см. табл.4). Укажите, какие поля из панели расширенного поиска Вами использовались, а также приведите список найденных нормативных документов.

Таблица 4.. Поисковые запросы

№ вар.	Запрос
1, 12	Найти все документы серии ГОСТ 34, принятые в 90-х годах
2, 13	Найти все документы серии ГОСТ 19, принятые в 1977-м году
3, 14	Найти все документы серии ГОСТ 34, принятые после 01.01.1977 года
4, 15	Найти стандарт по языку Бейсик
5, 16	Найти стандарты ГОСТ Р, вступившие в силу в 2016 году
6, 17	Найти стандарты ГОСТ Р, которые не используются с 2015 года
7, 18	Найти национальные стандарты, относящиеся к области телекоммуникаций, аудио- и видеотехники (волоконно-оптическая связь)
8, 19	Найти национальные стандарты, относящиеся к области языков, используемых в информационных технологиях
9, 20	Найти стандарты, наименование которых начинается с «системы обработки информации»
10, 21	Найти нормативные документы, утвержденные Правительством Санкт-Петербурга
11, 22	Найти стандарты, разработанные техническим комитетом «ТК 22 Информационные технологии»

7. Найдите в системе стандарты, определяющие трудовые функции конкретного вида профессиональной деятельности в сфере ИТ (возможные виды профессиональной деятельности: администратор баз данных, архитектор программного обеспечения, программист, систем-

ный аналитик, специалист по информационным системам, специалист по информационным ресурсам, технический писатель, специалист по тестированию в области информационных технологий). В отчет занесите наименования найденных стандартов и способ поиска.

Контрольные вопросы

1. Что такое стандарт?
2. Каков главный принцип стандартизации?
3. Какие категории стандартов выделяют?
4. Перечислите государственные и негосударственные организации, участвующие в деятельности по стандартизации.
5. Какие разновидности нормативных документов существуют?
6. Классифицируйте стандарты в зависимости от возникновения.
7. Классифицируйте стандарты в зависимости от масштаба.
8. Полное наименование стандарта состоит из
9. Что понимается под областью распространения стандарта?
10. Перечислите реквизиты стандартов.
11. Зачем используются классификаторы стандартов?
12. Каково назначение информационно-поисковых систем нормативов?
13. Что такое профессиональные стандарты?
14. Какие организации могут принимать участие в разработке профессиональных стандартов?

Лабораторная работа №2

ОПИСАНИЕ ЖИЗНЕННОГО ЦИКЛА ПРОГРАММНОГО СРЕДСТВА

Цель работы: обзорное изучение общесистемного стандарта ГОСТ Р ИСО/МЭК 12207-2010, регламентирующего жизненный цикл систем и программных средств.

Краткие теоретические сведения

Стандарт ГОСТ Р ИСО/МЭК 12207-2010 «Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных средств» описывает жизненный цикл системы и программного средства в терминах процессов. Процессы распределены на семь категорий: процессы соглашения, процессы организационного обеспечения проекта, процессы проекта, технические процессы, процессы реализации программных средств, процессы поддержки программных средств и процессы их повторного применения. Применение стандарта не предполагает обязательного использования всех возможных процессов для описания жизненного цикла. Конечный набор процессов зависит от масштаба системы и программных средств, входящих в ее состав, а также от предметной области.

Каждый процесс жизненного цикла описывается с помощью набора атрибутов: наименование, цель, выходы, деятельность, задачи.

Наименование – атрибут, указывающий на область применения процесса в рамках жизненного цикла системы или программного средства.

Цель – атрибут, описывающий конечные цели выполнения процесса жизненного цикла.

Выход – наблюдаемый результат, ожидаемый при успешном выполнении процесса.

Деятельность – перечень действий, используемых для достижения выходов процесса жизненного цикла.

Задачи – требования, рекомендации или допустимые действия, предназначенные для поддержки достижения выходов процесса.

Настоящий стандарт устанавливает строгую взаимосвязь между системой и применяемыми в ней программными средствами. Программное средство трактуется как единая часть общей системы, выполняющая определенные функции в данной системе. Поэтому требования к программным средствам отделяются от требований к системе.

Содержательную часть стандарта составляют разделы 4, 5, 6 и 7, описывающие, соответственно, термины и определения, порядок применения настоящего стандарта, процессы ЖЦ систем и процессы ЖЦ ПС. На использовании указанных разделов и должно строиться выполнение лабораторной работы.

Стандарт ИСО/МЭК 12207-2010 устанавливает требования для ряда процессов, приемлемых для использования в течение всего жизненного цикла программного продукта, услуги или системы. Документом допускается, что в отдельных проектах или организациях может отсутствовать потребность применять все процессы. В этом случае производится адаптация стандарта. При этом считается, что жизненный цикл программного продукта, услуги или системы находятся в адаптированном соответствии с требованиями стандарта. Иначе заявляется о полном соответствии нормативному документу. При любой из этих форм соответствия оформляется документ – заявление о соответствии. В случае полного соответствия в заявление заносится перечень процессов, которые удовлетворяют требованиям стандарта, иначе - адаптированный текст положений стандарта.

Задание к лабораторной работе №2

Опишите жизненный цикл программного средства, разработанного в рамках выполнения курсовой работы (например, по «Технологии программирования», или «Информатике», или «Языкам программирования»), в виде списка процессов (см. рис. 1, 2) на основе стандарта ГОСТ Р ИСО/МЭК 12207-2010.

Для всех выбранных процессов жизненного цикла укажите атрибуты (цель и выходы).

При этом рекомендуется произвести адаптацию стандарта. Адаптация может быть выполнена по нескольким направлениям. Во-первых, можно исключить процессы, не выполнявшиеся в рамках подготовки курсовой работы. Во-вторых, можно отобрать процессы жизненного цикла, требующие адаптации, и исключить отдельные выходы.

Пример описания выхода процессов жизненного цикла:

<наименование процесса>

*В результате успешного осуществления процесса реализации программного средства создана программная часть в виде файлов *.ras и *.exe.*

<наименование процесса>

В результате успешного осуществления процесса квалификационного тестирования программного средства получены следующие результаты тестирования:

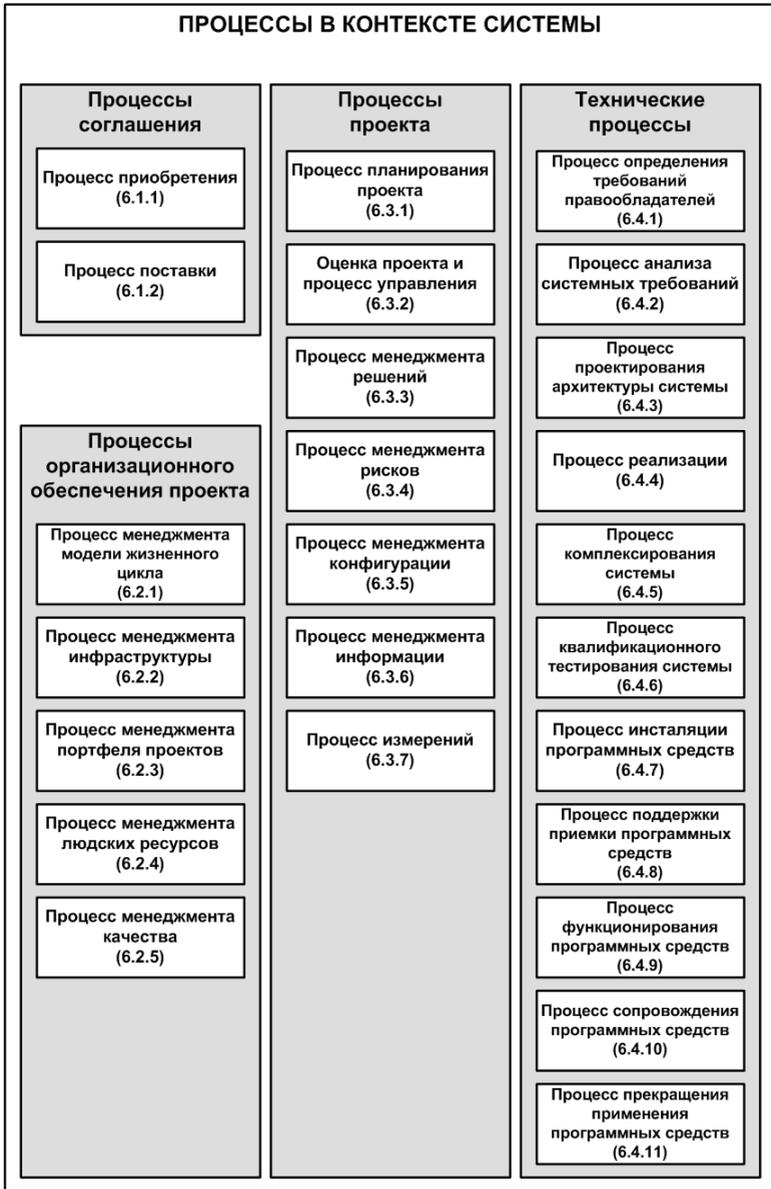


Рис. 1. Процессы жизненного цикла в контексте системы



Рис. 2. Специальные процессы жизненного цикла программных средств

Контрольные вопросы

1. Что такое общесистемный стандарт?
2. Что такое жизненный цикл программного средства?
3. Какие стандарты, регламентирующие жизненный цикл программных средств, Вам известны?
4. В терминах каких атрибутов описывается процесс жизненного цикла согласно ГОСТ Р ИСО/МЭК 12207-2010?
5. Что такое выход процесса жизненного цикла?
6. Какие группы процессов жизненного цикла устанавливает стандарт ГОСТ Р ИСО/МЭК 12207-2010?
7. Что такое процессы соглашения?
8. Что такое процессы организационного обеспечения проекта?
9. Что такое процессы проекта?
10. Что такое технические процессы?
11. Что такое процессы реализации программных средств?
12. Что такое процессы поддержки программных средств?
13. Что такое процессы повторного применения программных средств?
14. Почему стандарт ГОСТ Р ИСО/МЭК 12207-2010 устанавливает разделение процессов жизненного цикла на две категории - процессы в контексте системы и специальные процессы программных средств?
15. Что такое полное и адаптированное соответствие стандарту?
16. Что такое платформы класса ALM?
17. Какие программные платформы для управления жизненным циклом приложений Вам известны?
18. Каковы типовые функции платформ класса ALM?

Лабораторная работа №3

ОЦЕНКА КАЧЕСТВА И СЛОЖНОСТИ ПРОГРАММНОГО СРЕДСТВА

Цель работы: изучить некоторые методы оценки качества и сложности программных средств, применить их на практике.

Краткие теоретические сведения

Качество программной продукции по стандарту SQuaRE

Согласно стандартам серии ISO/IEC 25000 (SQuaRE) качество ИТ-продукции выражено как множество характеристик качества, которые, в свою очередь, детализируются подхарактеристиками. Для их понимания, указания и количественного оценивания необходимо идентифицировать свойства качества и элементы показателей качества, которые покрывают каждую характеристику/подхарактеристику.

Свойство качества – измеримый компонент качества, т.е. любой количественный признак или количественно оцениваемое взаимодействие программного обеспечения с его окружением.

Свойства качества измеряются посредством метода измерения. Метод измерения представляет собой логическую последовательность операций, используемых для количественного определения свойств относительно конкретной шкалы. Результат применения метода измерения называют элементом показателя качества (ЭПК).

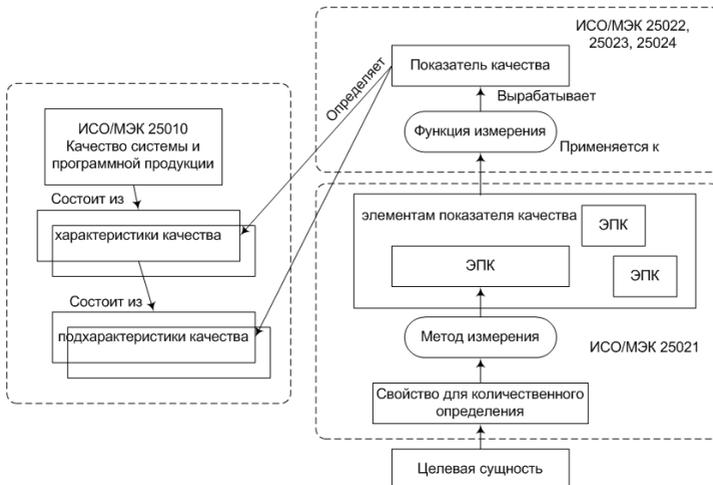


Рис. 3. Взаимосвязь между компонентами модели качества

Элемент показателя качества – показатель, определенный в терминах свойства и метода измерения для количественного определения этого свойства.

Показатель качества – показатель, получаемый как функция измерения не менее чем двух значений элементов показателя качества.

Целевая сущность (целевой объект) – результат работы или поведение системы, программного обеспечения или заинтересованных лиц.

Для облегчения понимания взаимосвязей внутри модели качества SQaRE рисунок 3 разобран на конкретном примере:



Рис. 4. Взаимосвязь между компонентами модели качества на примере

Стандарт ГОСТ Р ИСО/МЭК 25021–2014 «Элементы показателя качества» вводит четыре группы признаков для описания и проектирования любого ЭПК: признаки для идентификации; признаки для определения, что представляет собой ЭПК; признаки, необходимые при измерении ЭПК; признаки для управления применением ЭПК. Расшифровка каждого из признаков приведена в таблице 1 ГОСТ Р ИСО/МЭК 25021–2014. В таблице 2 того же документа дается пример описания ЭПК «Дефект кода» с помощью перечисленных групп признаков.

Одни и те же ЭПК могут быть связаны с разными показателями качества.

Таким образом, количественное определение ЭПК входит в общую оценку качества программной продукции.

Сложность программной продукции

Наиболее распространенным и доступным способом выявления сложности программного средства является использование метрик размера программ. Традиционной характеристикой размера программ является количество строк исходного текста. Под строкой понимается любой оператор программы. На этой характеристике основаны простейшие объемные метрики и метрики Холстеда.

К простейшим объемным метрикам относятся: L – число строк в программе, включая комментарии; S – число исполняемых операторов; U – число программных модулей: процедур, функции; S/U – средний размер программного модуля.

Основу метрик Холстеда составляют 4 измеряемые характеристики программы:

n_1 – число уникальных операторов программы;

n_2 – число уникальных операндов программы (словарь операндов);

N_1 – общее число операторов в программе;

N_2 – общее число операндов в программе.

Операнд – константа, переменная, функция и др. объект языка программирования, над которым производится действие.

Оператор – фраза алгоритмического языка, определяющая законченный этап обработки данных. В состав оператора могут входить ключевые слова, константы, переменные, знаки операций.

Оценки Холстеда вычисляются по формулам (1)÷(11).

Словарь программы:

$$n = n_1 + n_2. \quad (1)$$

Длина программы:

$$N = N_1 + N_2. \quad (2)$$

Объем программы (в битах):

$$V = N \cdot \log_2(n). \quad (3)$$

Длина теоретического словаря:

$$n^* = n_1^* + n_2^*. \quad (4)$$

Длина теоретического словаря расшифровывается как словарный запас, необходимый для написания программы, с учетом того, что необходимая функция уже реализована в данном языке и, следовательно,

программа сводится к вызову этой функции. Тогда за n_1^* принимается количество вызов пользовательских и стандартных процедур/функций, за n_2^* – количество передаваемых параметров.

Потенциальный объем программы:

$$V^* = n^* \cdot \log_2(n^*). \quad (5)$$

Теоретическая длина программы:

$$N^{\wedge} = n_1 \cdot \log_2(n_1) + n_2 \cdot \log_2(n_2). \quad (6)$$

Для стилистически корректных программ отклонение в оценке теоретической длины N^{\wedge} от реальной N не превышает 10%.

Уровень качества программирования (уровень программы):

$$L = V^*/V. \quad (7)$$

Для идеальной программы $L=1$, для реальной – всегда $L<1$. Холстед предлагает аппроксимировать эту оценку выражением, включающим только фактические характеристики программы:

$$L^{\wedge} = (2 \cdot n_2) / (n_1 \cdot N_2). \quad (8)$$

Интеллектуальное содержание конкретного алгоритма:

$$I = L^{\wedge} \cdot V. \quad (9)$$

Число требуемых элементарных решений при написании программы:

$$E = N^{\wedge} \cdot \log_2(n/L). \quad (10)$$

Затраты на восприятие готовой программы:

$$E' = N \cdot \log_2(n/L). \quad (11)$$

Задание к лабораторной работе №3

Часть 1. Оценка качества программного средства

1.1. Выберите программную продукцию. Приведите краткую информацию об оцениваемом программном средстве (название, разработчик, год создания, язык программирования).

1.2. Выберите 2-3 характеристики/подхарактеристики качества из стандарта ГОСТ Р ИСО/МЭК 25010–2015 «Модели качества систем и

программных продуктов». Термины и определения характеристик и подхарактеристик качества приведены в п.4 указанного стандарта.

1.3. Для выбранных характеристик/подхарактеристик по Приложению А стандарта ГОСТ Р ИСО/МЭК 25021–2014 «Элементы показателя качества» подберите 2-3 ЭПК. На примере оцениваемой программной продукции проведите измерение этих ЭПК. В отчет занесите описание метода измерения и полученные количественные значения.

Часть 2. Оценка сложности программного средства

2.1. Для оцениваемой программной продукции определите простейшие объемные метрики. Результаты оформите в виде таблицы:

Таблица 5. Значение объемных метрик

Объемные метрики			
L	S	U	S/U
...

2.2. Определите элементы и характеристики программы для метрик Холстеда. Результаты оформите в следующем виде (привести формулы и промежуточные расчеты):

$$\begin{aligned}
 n_1 &= \dots; n_2 = \dots; n = n_1 + n_2 = \dots; N_1 = \dots; N_2 = \dots; N = \dots; \\
 V &= \dots; n^* = \dots; V^* = \dots; N^\wedge = \dots; L = \dots; L^\wedge = \dots; \\
 I &= \dots; \\
 E &= \dots; \\
 E' &= \dots
 \end{aligned}$$

Таблица 6. Значение метрик Холстеда

Метрики Холстеда									
n	N	V	V*	N^	L	L^	I	E	E'
...						

2.3. Проанализируйте рассчитанные значения метрик. Сделайте вывод о стилистической корректности программы, качестве программирования, интеллектуальном содержании алгоритма, заложенного в программу, затратах на восприятие готовой программы.

2.4. Найдите информацию о других метриках программного кода.

Контрольные вопросы

1. Что такое менеджмент качества?
2. Какие стандарты относятся к оценке качества программных средств?
3. Что такое качество программного средства?
4. Опишите структуру серии стандартов SQuaRE.
5. Что такое модель качества программной продукции?
6. Что такое характеристика качества?
7. Что такое свойство качества?
8. Зачем в модель качества программной продукции вводятся элементы показателя качества?
9. Приведите примеры элементов показателя качества.
10. Опишите взаимосвязи между компонентами модели качества SQuaRE.
11. Перечислите методы определения показателей качества ПС исходя из способов получения информации о них.
12. Перечислите методы определения показателей качества ПС исходя из источников получения информации о них.
13. От чего зависит качество программной продукции?
14. Перечислите модели качества процесса разработки программной продукции.
15. Чем отличаются модели CMM и SPICE?
16. Какие метрики используются для оценки сложности программной продукции?

Лабораторная работа №4

ОЦЕНКА ТЕСТОВОГО ПОКРЫТИЯ

Цель работы: изучить способы измерения покрытия исходного кода тестами и провести обзор инструментов автоматизации тестирования.

Краткие теоретические сведения

Тестовое покрытие – степень, выраженная в процентах, в которой некоторые атрибуты или их комбинации были проверены контрольными примерами.

Выделяют следующие виды тестового покрытия: *покрытие требований* (requirements coverage), *покрытие кода* (code coverage), *покрытие на базе анализа потока управления* (control flow coverage).

Покрытие требований – оценка покрытия тестами функциональных и нефункциональных требований к программному продукту. Для измерения этого показателя необходимо: проанализировать множество требований к продукту и разбить их на отдельные пункты; связать каждый пункт с тестовыми случаями, проверяющими его. Связи между пунктами требований и конкретными тестовыми случаями образуют *матрицу трассировки*. В общем случае покрытие требований T_{cov} вычисляется как

$$T_{cov} = (L_{cov}/L_{total}) \cdot 100\%, \quad (12)$$

L_{cov} – количество требований, проверяемых тестовыми случаями; L_{total} – общее количество требований.

Поскольку рассмотренный вид тестового покрытия не учитывает конечную реализацию алгоритмов, то покрытие требований может оставить неохваченными некоторые фрагменты исходного кода.

Тестовый случай (test case) – структура, описывающая совокупность действий, конкретных условий и параметров, необходимых для проверки функционального требования или его части. Тестовый случай имеет унифицированное описание, включающее список действий, ожидаемый от них результат и результат теста. Методики разработки тестовых случаев на основе имеющихся требований рассматриваются в тест дизайне.

Тест дизайн – первоначальный этап тестирования программной продукции, включающий планирование и проектирование тестовых случаев.

С методикой практического применения тест дизайна и примерами тестовых случаев можно ознакомиться по ссылке http://www.protesting.ru/testing/testdesign_practice.html.

Покрываемость кода T_{cov} – второй вид тестового покрытия – является простейшей метрикой и определяется исходя из формулы

$$T_{cov} = (L_{ic}/L_{code}) \cdot 100\%, \quad (13)$$

L_{ic} – число выполненных строк исходного кода; L_{code} – общее число строк.

В качестве примера на рисунке 5 приведен фрагмент кода, возвращающего период в развитии человека на основе пола и возраста. Для вычисления покрытия кода взято два теста:

Исходные данные: "female", 17; результат содержит "girl";

Исходные данные: "male", 16; результат содержит "teenager".

При имеющихся входных данных $T_{code} = (6 / 20) \cdot 100\% = 30\%$.

```

1 public String describePersonAge(String gender, int age) {
2     if (gender.equals("male")) {
3         if (age > 12 && age < 20) {
4             return "A teenager";
5         } else if (age < 25) {
6             return "A young man";
7         } else if (age > 25 && age < 40) {
8             return "A man in his prime";
9         }
10    } else if (gender.equals("female")) {
11        if (age < 20) {
12            return "A girl";
13        } else if (age > 20 && age < 40) {
14            return "A young woman in childbearing age";
15        } else {
16            return "An older woman";
17        }
18    }
19    return "A " + gender + " in undefined age";
20 }

```

Рис. 5. Пример 30% покрытия кода

Более детальную информацию о тестовом покрытии можно получить на основе *тестирования потоков управления* (control flow testing).

Тестирование потоков управления – техника структурного тестирования (тестирования методом «белого ящика»), суть которой в определении путей выполнения исходного кода и проектировании тестовых случаев для их [путей] покрытия. Данный вид тестирования использует *граф потока управления* – множество всех возможных путей исполнения программы, представленное в виде ориентированного графа.

Для тестирования потоков управления вводится семь уровней тестового покрытия (см. табл. 7).

Таблица 7. Уровни тестового покрытия

Уровень	Наименование	Примечание
0	"Test whatever you test, users will test the rest"	"Тестируйте, что вы хотите, остальное протестируют пользователи"
1	Покрытие операторов	Каждый оператор выполнен как минимум один раз
2	Покрытие альтернатив (ветвей)	Проверяется, в какой степени покрыты альтернативные проходы в программе. Т.е. каждый узел с ветвлением (альтернатива) выполнен как минимум один раз (хотя бы раз проверяется каждая точка ветвления алгоритма)
3	Покрытие условий	Каждый вариант условия (true и false) должен быть выполнен минимум один раз
4	Покрытие условий альтернатив	Суммарное применение уровней со второго по третий
5	Покрытие множественных условий	Суммарное применение уровней со второго по четвертый. Т.е. оценивается покрытие всех возможных комбинаций условий
6	Покрытие бесконечного числа путей	Задается максимальная глубина оценки покрытия при наличии длинного цикла или рекурсии
7	Покрытие путей	Проверке подлежат все пути без исключения Самый высокий уровень покрытия, используя который можно сказать, что программа на 100% покрыта тестами

Величина покрытия на любом уровне определяется как отношение проверенных контрольными примерами атрибутов (альтернатив, условий, условий альтернатив, множественных условий или путей) к их общему числу, т.е. аналогично формулам (12) и (13).

На псевдопримерах рассмотрим ключевые уровни покрытия.

Пример 1

Варианты входных данных

if a then

 x++

a = true

else

 x--

a = false

endif

Для покрытия операторов (уровень 1) `x++` и `x--` в примере 1 нужно проверить всего два варианта входных данных.

Пример 2 >	<code>if (a or b) and c then</code>
Покрытие условий	Покрытие условий альтернатив
<code>a = true, b = true, c = true</code>	<code>a = false, b = false, c = true</code>
<code>a = false, b = false, c = false</code>	<code>a = true, b = false, c = true</code>
	<code>a = false, b = true, c = true</code>
	<code>a = false, b = true, c = false</code>

В примере 2 для покрытия условий (проход по ветке `if` и другой вариант) достаточно взять два набора тестовых данных, значение `b` не важно при условии `a = true`. Для покрытия условий альтернатив требуется больше вариантов входных данных, так как важно, чтобы каждое условие внутри альтернативы было выполнено.

Покрытие множественных условий считается 100%, если входные данные охватили все возможные комбинации условий. 100% покрытие множественных условий для примера 3 гарантирует проверка восьми вариантов (2^3) входных данных.

Пример 3 >	<code>if (a or b) and c then</code>
<code>a = false, b = false, c = false</code>	<code>a = true, b = false, c = false</code>
<code>a = false, b = false, c = true</code>	<code>a = true, b = false, c = true</code>
<code>a = false, b = true, c = false</code>	<code>a = true, b = true, c = false</code>
<code>a = false, b = true, c = true</code>	<code>a = true, b = true, c = true</code>

В целях автоматизации тестирования инструменты анализа покрытия кода встроены во многие *IDE* (Integrated Development Environment, интегрированная среда разработки), например *Visual Studio*. Существуют и частные решения – утилиты (*gcov*, *tcov*, *lcov*, *codecov*) и системы анализа покрытия программного кода (*Testwell CTC++*), написанного на разных языках.

Задание к лабораторной работе №4

Задания по определению тестового покрытия выполняются на примере логически завершенного фрагмента программного кода (фрагмент лабораторной или курсовой работы) размером 60-100 строк. Фрагмент текста программы обязательно должен содержать условные операторы.

1. Постройте наборы входных данных, которые в совокупности обеспечат покрытие кода не менее 30%. В отчете перечислите эти наборы,

а также строки кода, которые они покрывают. Вычислите точное значение покрытия кода T_{cov} .

2. Определите наборы входных данных, которые в совокупности обеспечат покрытие альтернатив не менее 30%. В отчете перечислите эти наборы, укажите, какие альтернативы они покрывают. Вычислите точное значение покрытия альтернатив.

3. Пункт 2 проделайте для третьего и четвертого уровней тестового покрытия.

4. Проведите обзор инструментов автоматизации тестирования. В отчет занесите примеры таких инструментов и их краткое описание.

Факультативно. Используйте инструменты для автоматического вычисления тестового покрытия на практике. Приведите краткое описание своих действий. Сравните с результатами п. 1 и 2.

Контрольные вопросы

1. Основное действие при анализе качества, валидации и верификации программного обеспечения – это ... ?
2. В каких случаях следует проводить тестирование ПО?
3. Перечислите стандарты, описывающие процесс тестирования ПО.
4. Известны две методики тестирования –
5. Тестирование методом "черного ящика" – это
6. Статический анализ исходного кода включает
7. Какие уровни тестирования выделяют?
8. Что такое тип тестирования?
9. Какие типы тестирования предусмотрены для характеристики качества "производительности"?
10. Что такое тестирование защищенности?
11. Что включает нефункциональное тестирование?
12. Что такое тестирование совместимости?
13. В чем различие между альфа- и бета-тестированием?
14. Что понимается под тестовым покрытием?
15. Какие виды тестового покрытия предусмотрены?
16. Перечислите виды документации тестирования.
17. Что включает организационная документация тестирования?
18. Как проводится сертификация специалистов по тестированию программных продуктов?

Лабораторная работа №5

РАЗРАБОТКА ПРОЕКТА ЛЕГАЛИЗАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Цель работы: познакомиться с программами лицензирования крупных производителей программного обеспечения; получить практические навыки подбора лицензионного программного обеспечения для нужд организации заданного профиля.

Краткие теоретические сведения

Для принятия решения о количестве приобретаемых лицензий на право использования программного обеспечения предусмотрен ряд метрик лицензирования (см. табл. 8).

Метрика лицензирования (англ. *license metric*) – это измеримый показатель, на основании которого определяется число лицензий, необходимых для полноценного и корректного лицензирования программного обеспечения.

Таблица 8. Базовые метрики лицензирования

Метрика	Описание	Примеры
Device (устройство) или Seat (рабочее место)	Число лицензий определяется числом устройств (компьютеры, ноутбуки и т.д.), на которых устанавливается и/или используется лицензируемое приложение.	Локально работающие приложения, например, из пакета MS Office, а также клиентские лицензии серверных продуктов.
User (пользователь)	Число лицензий определяется числом пользователей, которые будут использовать лицензируемое приложение. Обычно при этом число устройств, на которых устанавливается лицензируемое приложение, не ограничено.	Локально работающие приложения, приложения для домашнего использования, а также клиентские лицензии серверных продуктов.
Access (доступ) или Subscriber (подписчик)	Число лицензий определяется числом реквизитов доступа (логин, пароль и т.д.), которые выдаются пользователям для доступа к приложению.	Сетевые приложения, web-сервисы.
Concurrent connection (конкурентное подключение)	Число лицензий определяется максимальным числом одновременно работающих пользователей. Обычно при этом число устройств, на которых устанавливается лицензируемое приложение, не ограничено. Требуется менеджер лицензий.	Приложения, работа с которыми возможна по сети, например, продукты 1С.

Метрика	Описание	Примеры
Processor (процессор)	Число лицензий определяется числом процессоров, установленных в серверах, на которых работает лицензируемое приложение. При этом число пользователей и число запускаемых экземпляров приложений не ограничено.	Серверные приложения, большинство СУБД, например, Oracle Database, Microsoft SQL Server.
Named User (NU)	Число лицензий определяется числом пользователей (как для User), при этом лицензии являются именными, т.е. оформляют на конкретное юридическое лицо.	Продукты Oracle, например, Oracle Database.
Processor Value Unit (PVU)	Число лицензий определяется суммой баллов для всех процессоров, установленных в серверах, на которых работает лицензируемое приложение. Баллы для каждого из типов процессоров устанавливаются на основе таблиц компании IBM.	Серверные приложения компании IBM, например, Lotus (документооборот).
Network address (сетевой адрес)	Число лицензий определяется числом IP-адресов (подсетей) компьютеров, с которыми взаимодействует лицензируемое приложение.	Сетевые продукты.

Исходя из способа получения конечным пользователем программного продукта, существует три канала распространения лицензий: OEM, коробочные и корпоративные лицензии.

OEM-лицензия (от англ. *original equipment manufacturer*) – лицензия, допускающая использование ПО только в том случае, если оно приобретено в составе нового компьютерного оборудования (например, компьютера или сервера).

В виде OEM-лицензий доступны операционные системы, офисные приложения, средства резервного копирования. При этом имеет место ряд особенностей: OEM-версию ПО допустимо использовать только с тем оборудованием, в составе которого оно приобретено; в OEM-версиях часто реализуется активация с привязкой к оборудованию; OEM-лицензию можно передать или продать третьему лицу, но только в составе оборудования, с которым она была первоначально приобретена.

Коробочная лицензия (от англ. *box license* или *BOX*) – лицензия, дающее право использования ПО тому физическому или, реже, юридическому лицу, которое правомерно владеет коробкой (упаковкой), вместе с которой поставляется коробочная лицензия.

Особенности: коробочные продукты не привязаны к какому-либо оборудованию и не являются именованными. В этом случае права использования передаются не какому-то конкретному юридическому или физическому лицу, а тому, кто является владельцем коробки и выполнил установку программного продукта на свой компьютер.

Корпоративная лицензия (англ. *corporate license*) – лицензия, предназначенная в первую очередь для распространения юридическим лицам.

Особенности: стоимость корпоративных лицензий не является фиксированной, а зависит от многих факторов (объема закупки, состава продуктов, прогноза закупки на год-два вперед, скидок учебным учреждениям). Корпоративные лицензии в большинстве случаев являются именованными, т.е. оформляются на конкретное юридическое или, реже, физическое лицо, которое указывается в тексте лицензионного соглашения. По умолчанию корпоративные лицензии нельзя передать третьему лицу, если иное не указано в лицензионном соглашении. На приобретение корпоративных лицензий часто действуют экспортные ограничения.

Любое программное обеспечение, находящееся на балансе предприятия, называется программными активами. Для комплексного управления программными активами разработана методология SAM (англ. *Software Asset Management, Управление активами программного обеспечения*).

SAM – методология, направленная на оптимизацию процессов управления программным обеспечением в организации и его защиту. В рамках методологии процессы управления включают: учет программного обеспечения и его использования; учет лицензий (прав на использование), документов, подтверждающих наличие прав на использование (сертификаты, лицензионные свидетельства, лицензионные соглашения, договора, бухгалтерские документы); разработку и использование регламентов и политик закупки программного обеспечения, ввода его в эксплуатацию, эксплуатации, вывода из эксплуатации и др.

Использование SAM (универсальный стандарт ISO/IEC 19770) на предприятии – это стандартизированный подход к любым операциям с программным обеспечением.

Задания к лабораторной работе №5

1. Ознакомьтесь с программами лицензирования, предлагаемыми известными производителями программного обеспечения (см. табл. 9). В отчет включить следующие сведения: специализация компании-производителя, перечень программ лицензирования и их краткая характеристика.

2. Предприятие, профиль деятельности которого определен в таблице 10, направляет Вам заказ на разработку проекта легализации программного обеспечения. Проект легализации должен распространяться как на общеприменимое ПО (операционная система, антивирусная программа, пакет офисных программ и т.п.), так и на специализированное ПО, характерное для предметной области, с которой связана деятельность организации.

Исходные условия задания Вы можете доопределить самостоятельно: перечень ПО, на которое истекают лицензии, размер организации, количество компьютеров и другого оборудования, размер средств, выделяемых на легализацию ПО, и др.

Проект легализации должен включать две части. Во-первых, обследование состояния программных активов (какое программное обеспечение используется, лицензионное ли оно, срок действия имеющихся лицензий и их тип). Во-вторых, новая схема лицензирования со сметой расходов на приобретение истекающих или отсутствующих лицензий.

Шаблон для проекта легализации приведен в Приложении 1.

При разработке новой схемы лицензирования должны быть соблюдены следующие условия:

- допускается замена нелицензионного программного обеспечения на аналогичное свободное ПО;
- необходимо указать метрики лицензирования, на основе которых принимается решение о количестве приобретаемых лицензий;
- необходимо указать тип приобретаемой лицензии (свободная, коммерческая, для разработчиков, для учебных заведений и т.д.);
- при составлении сметы следует указывать источники (сайт фирмы-поставщика, цена);
- рекомендуется использовать программы лицензирования от компаний-производителей программного обеспечения;
- желательно представить новую схему лицензирования в нескольких вариантах на выбор заказчика.

Таблица 9. Вендоры программных продуктов

№ вар.	Организация-производитель программного обеспечения
1,17	Kaspersky Lab (Лаборатория Касперского)
2, 18	Autodesk
3, 19	Adobe Systems
4, 20	ABBYY
5, 21	Symantec
6, 22	Embarcadero
7, 23	Oracle
8, 24	Citrix Systems
9, 25	VMware
10, 26	SAP
11, 27	Corel Corporation
13, 28	Microsoft Corporation
14, 29	MathWorks
15, 30	ESET
16, 31	IC

Таблица 10. Перечень профилей для проекта легализации ПО

№ вар.	Профиль организации
1, 11	Высшее учебное заведение. Главная цель – автоматизировать процедуру управления учебным заведением (автоматизация бухгалтерского и финансового учета, управление студенческим контингентом и приемной кампанией).
2, 12	Издательский центр с сетью офисов по городу (категория* предприятия - малое).
3, 13	Отдел бухгалтерии (15 чел.) государственного бюджетного учреждения. Главная цель – автоматизация бухгалтерского учета с возможностью отправлять электронную отчетность в контролирующие органы (ФСС, ФНС, ПФР, Росстат).
4, 14	Малая компания, специализирующаяся на дизайне помещений.
5, 15	Отдел разработки и тестирования (20-30 чел.) программных продуктов компании, специализирующейся на создании программного обеспечения.
6, 16	Бюро переводов (на иностранные языки) с отделом взаимоотношений с клиентами компании. Необходима возможность обмена юридически значимыми электронными документами (договоры, накладные) с клиентами.
7, 17	Отдел организации удаленных совещаний (видеоконференций) международной компании.
8, 18	Малая веб-студия с внутренней системой электронного документооборота.
9, 19	Средняя компания по разработке программного обеспечения для финансового сектора.
10, 20	Отдел бухгалтерии (20-30 чел.) коммерческой компании по продаже недвижимости. Главная цель – создание виртуальной инфраструктуры и переход на облачные решения для бизнеса.

*Пояснения к таблице 10 по категориям предприятий в соответствии с федеральным законом от 24.07.2007 N 209-ФЗ (ред. от 03.07.2016) "О

развитии малого и среднего предпринимательства в Российской Федерации":

- для микропредприятия численность работников – до 15 человек;
- для малых предприятий – до 100 человек;
- для средних предприятий – от 101 до 250 человек.

Контрольные вопросы

1. Перечислите опасности, связанные с использованием нелегального программного обеспечения.
2. Перечислите субъекты и объекты, участвующие в передаче прав на программное обеспечение.
3. Перечислите нормативные документы, являющиеся правовой основой для защиты авторских прав и прав правообладателей программного обеспечения.
4. Какие типы лицензий существуют?
5. Перечислите преимущества и недостатки свободных лицензий.
6. Какие организации занимаются продвижением идей свободного программного обеспечения?
7. В чем отличие лицензий для разработчиков и лицензий для независимых разработчиков?
8. В чем отличие между свободным ПО и ПО с открытым исходным кодом?
9. Что такое метрики лицензирования? Для чего они необходимы?
10. Перечислите базовые метрики лицензирования.
11. Укажите известные Вам каналы распространения лицензий.
12. От каких факторов зависит стоимость корпоративных лицензий?
13. Каким образом можно повысить эффективность управления программными активами организации?

Индивидуальное домашнее задание

СОЗДАНИЕ ПАКЕТА ДОКУМЕНТАЦИИ НА ПРОГРАММНУЮ ПРОДУКЦИЮ

Цель работы: изучить стандартизированный перечень документов, входящий в состав пакета документации на программную продукцию; получить навыки формирования основных документов из пакета программной документации.

Краткие теоретические сведения

Тематика индивидуального домашнего задания связана с созданием пакета документации на программное средство, разработанное в рамках любой выполненной курсовой работы (курсового проекта). При документировании программного средства следует руководствоваться стандартами документирования, например ГОСТ 34.201-89, ГОСТ 34.602-89.

Любые стандарты на организацию жизненного цикла систем и программной продукции вводят процессы (стадии, этапы) по документированию готового программного продукта и процедур его разработки. Кроме того, реализация в компании процессов по документированию хода разработки программной продукции свидетельствует о высоком уровне зрелости этой компании с точки зрения способности разрабатывать качественный продукт.

Трудовые функции, связанные со знаниями, навыками и умениями документировать процессы жизненного цикла программной продукции, определены в профессиональных стандартах по ряду сфер деятельности (технический писатель, программист и др.).

Документация на программный продукт (автоматизированную систему) – комплекс взаимосвязанных документов, в котором полностью описаны все решения по созданию и функционированию продукта, а также документов, подтверждающих соответствие продукта требованиям задания на выполнение работ (технического задания) и его готовность к функционированию в реальной среде.

Согласно стандартам серии ГОСТ 34 основными документами, выпускаемыми в рамках жизненного цикла программной продукции, являются:

- техническое задание на разработку;
- описание автоматизируемых функций;

- описание постановки задач или их комплекса;
- описание информационного обеспечения системы;
- описание организации информационной базы;
- описание систем классификации и кодирования (рекомендуется использовать Единую систему классификации и кодирования технико-экономической и социальной информации, *ЕСКК*);
- описание комплекса технических средств;
- описание программного обеспечения;
- описание организационной структуры;
- проектная оценка надежности системы;
- спецификация оборудования;
- каталог базы данных;
- методика автоматизированного проектирования;
- руководство пользователя;
- программа и методика испытаний системы.

Требования к содержанию перечисленных документов установлены в руководящем документе по стандартизации РД 50-34.698-90.

Документы, относящиеся к одному комплекту документации на программное средство, упорядочиваются и кодируются в соответствии со следующей структурой (рис. 6):



Рис. 6. Обозначение документов из пакета документации на программную продукцию

Адаптированное обозначение документа, включающее лист регистрации изменений, может иметь вид, представленный на рисунке 7.

Требования к документированию программной продукции согласно другому стандарту – ГОСТ Р ИСО/МЭК 12207-2010 – обозначены в

рамках одного из процессов ЖЦ, а именно процесса менеджмента программной документации.

Для автоматизации процесса документирования программной продукции на основе специальным образом оформленных комментариев, содержащихся в исходном коде, используются генераторы документации (для разных языков и сред программирования). К их числу относятся: *Doxygen* (C++, C, Objective C, C#, Java, PHP), *Epydoc* (Python), *Javadoc* (Java), *JSDoc* (Java Script), *PhpDocumentor* (PHP), *PHPDoc* (PHP), *Sandcastle* (C#, семейство языков платформы .NET), *Sphinx* (Python) и др.



Информация о документе

Название проекта:	Проектирование и развертывание сетевой и ИТ инфраструктуры на новом объекте Европейского медицинского центра		
Руководитель проекта TopS BI:	Е.В. Грицанова	Номер версии:	1.1
Руководитель проекта EMC:	Ю.М. Кашкаров	Дата версии:	10.04.2009
Подготовил:	Д.С. Потапов	Дата подготовки:	10.04.2009

Лист регистрации изменений

Версия	Дата	Изменение	Описание	Имя файла
1.0	21.03.2009	Д.С. Потапов	Проект документа	EMC_ИТ-инфраструктура_Отчет об обследовании_v1-0.doc
1.1	10.04.2009	Д.С. Потапов	Корректировка документа в соответствии с комментариями Заказчика	EMC_ИТ-инфраструктура_Отчет об обследовании_v1-1.doc

Рис. 7. Адаптированное описание документа из пакета документации

Стандарты документирования покрывают следующие типы документации на программное средство:

- проектная (включает текстовые описания, диаграммы, модели разрабатываемого программного средства на основе методологий проектирования IDEF0, DFD, IDEF3, IDEFX, UML и др.);
- техническая (используется для определения и описания структур данных и алгоритмов; предоставляется с исходным кодом программной продукции или включается в него);
- пользовательская (ориентирована на конечного пользователя).

Помимо этого выделяют четвертый тип документации - маркетинговую документацию, включающую рекламные материалы, например бизнес-план по продвижению программной продукции, графические материалы по оформлению коробки продукта и т.п.

Задания к ИДЗ

Индивидуальное домашнее задание включено в учебный план по направлению подготовки 09.03.02 – Информационные системы и технологии для студентов заочной формы обучения.

1. Ознакомьтесь с пакетом документов реального проекта (см. папку из комплекта лабораторных работ ...*пакет документации*).
2. По примеру документации реального проекта и стандартов серии ГОСТ 34 разработайте пакет документов на программную продукцию, полученную в рамках выполнения любой курсовой работы (курсового проекта). Пакет документов обязательно должен включать техническое задание на создание, программу и методику испытаний, инструкцию (руководство) по использованию. Другие виды документов включаются в пакет по решению разработчика (студента). Результаты работы сдаются на проверку в электронном виде (архив документов *Фамилия_группа.rar/zip/7z*).

Контрольные вопросы

1. Какие стандарты устанавливают требования к документированию программной продукции?
2. Приведите примеры видов и наименований документов, разрабатываемых на разных стадиях жизненного цикла программной продукции и автоматизированных систем.
3. Что такое техническое задание?
4. Какова структура технического задания в соответствии со стандартом ГОСТ 34.201-89?
5. Что такое эскизный проект?
6. Что такое технический проект?
7. Что такое рабочий проект?
8. Что входит в документацию тестирования?
9. Что должен содержать документ «Описание автоматизируемых функций»?
10. Что должен содержать документ «Описание информационного обеспечения системы»?
11. Что должен содержать документ «Описание программного обеспечения»?
12. Что должен содержать документ «Описание комплекса технических средств»?
13. Что должен содержать документ «Описание организационной структуры»?

14. Что должен содержать документ «Спецификация оборудования»?
15. Что должен содержать документ «Программа и методика испытаний (компонентов, комплексов средств автоматизации, подсистем, систем)»?
16. Какие документы относятся к организационно-распорядительным и разрабатываются на стадии ввода программной продукции в действие?
17. Какую структуру имеет обозначение каждого документа?

Приложение 1

Шаблон проекта легализации программных активов

Разработка проекта легализации для компании, специализирующейся на <профиль компании>.

1. Информация о компании и размерах средств, выделяемых на легализацию.

Размер организации – N человек, среди них <перечисляются должности и число сотрудников, их занимающих>. Средства, выделенные на легализацию, оцениваются в M тыс. руб.

2. Аудит программных активов.

Таблица 11.

Перечень программ, установленных на ПК и серверах компании

Рабочее место	Название ПО, версия	Срок действия лицензии, ее тип
<должность>	<перечень программ, установленных на рабочем месте>	

Итоги аудита программных активов: <указать, выявлено ли нелегальное ПО, для каких программных продуктов требуется продление лицензий и т.п.>.

3. Схема лицензирования со сметой.

Таблица 12.

Приобретаемые лицензии

Программный продукт	Количество, выбранная метрика лицензирования, тип лицензии	Стоимость, руб	Поставщик
<название программы, версия>	<перечень программ, установленных на рабочем месте>		<ссылка на прайс-лист дилера>
...
ИТОГО:			

<Альтернативная схема лицензирования со сметой>

4. Выводы.

Библиографический список

1. Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных средств [Электронный ресурс]: ГОСТ Р ИСО/МЭК 12207-2010. – Введ. 2012-03-01. – Режим доступа: Система Кодекс ТехЭксперт.
2. Информационные технологии (ИТ). Системная и программная инженерия. Требования и оценка качества систем и программного обеспечения (SQuaRE). Модели качества систем и программных продуктов [Электронный ресурс]: ГОСТ Р ИСО/МЭК 25010-2015. – Введ. 2016-06-01. – Режим доступа: Система Кодекс ТехЭксперт.
3. Информационные технологии (ИТ). Системная и программная инженерия. Требования и оценка качества систем и программного обеспечения (SQuaRE). Элементы показателя качества [Электронный ресурс]: ГОСТ Р ИСО/МЭК 25021-2014. – Введ. 2015-06-01. – Режим доступа: Система Кодекс ТехЭксперт.
4. Системная и программная инженерия. Тестирование программно-го обеспечения. Часть 1. Понятия и определения [Электронный ресурс]: ГОСТ Р 56920-2016/ISO/IEC/IEEE 29119-1:2013. – Введ. 2017-06-01. – Режим доступа: Система Кодекс ТехЭксперт.
5. Системная и программная инженерия. Тестирование программно-го обеспечения. Часть 2. Процессы тестирования [Электронный ресурс]: ГОСТ Р 56921-2016/ISO/IEC/IEEE 29119-2:2013. – Введ. 2017-06-01. – Режим доступа: Система Кодекс ТехЭксперт.
6. Информационная технология (ИТ). Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания [Электронный ресурс]: ГОСТ 34.601-90. – Введ. 1992-01-01. – Режим доступа: Система Кодекс ТехЭксперт.
7. Информационная технология (ИТ). Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы [Электронный ресурс]: ГОСТ 34.602-89. – Введ. 1990-01-01. – Режим доступа: Система Кодекс ТехЭксперт.
8. Информационная технология (ИТ). Комплекс стандартов на автоматизированные системы. Виды, комплектность и обозначение документов при создании автоматизированных систем [Электронный ресурс]: ГОСТ 34.201-89. – Введ. 1990-01-01. – Режим доступа: Система Кодекс ТехЭксперт.

Содержание

ВВЕДЕНИЕ	3
Лабораторная работа №1. ЗНАКОМСТВО С ИНФОРМАЦИОННО- ПОИСКОВЫМИ СИСТЕМАМИ НОРМАТИВОВ	5
Лабораторная работа №2. ОПИСАНИЕ ЖИЗНЕННОГО ЦИКЛА ПРОГРАММНОГО СРЕДСТВА	10
Лабораторная работа №3. ОЦЕНКА КАЧЕСТВА И СЛОЖНОСТИ ПРОГРАММНОГО СРЕДСТВА	15
Лабораторная работа №4. ОЦЕНКА ТЕСТОВОГО ПОКРЫТИЯ	21
Лабораторная работа №5. РАЗРАБОТКА ПРОЕКТА ЛЕГАЛИЗАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ	26
Индивидуальное домашнее задание. СОЗДАНИЕ ПАКЕТА ДОКУМЕНТАЦИИ НА ПРОГРАММНУЮ ПРОДУКЦИЮ.....	32
Приложение 1. Шаблон проекта легализации программных активов .	37
Библиографический список	38

Учебное издание

Стандартизация и лицензирование программного обеспечения

Методические указания к выполнению лабораторных работ
и индивидуальных домашних заданий для студентов всех форм
обучения направлений подготовки 09.03.02 – Информационные
системы и технологии, 09.03.03 – Прикладная информатика

Составитель **Косоногова** Марина Александровна

Подписано в печать Формат 60x84/16. Усл. печ. л. 2,3. Уч.-изд.л. 2,5.

Тираж

Заказ

Цена

Отпечатано в Белгородском государственном технологическом университете

им. В.Г. Шухова

308012, г.Белгород, ул.Костюкова, 46