1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32

N 1: СТБ ИСО/МЭК 12207-2003. Процессы жизненного цикла программных средств и их классификация. Общие сведения.

Жизненный Цикл (ЖЦ) программного средства (ПС) – совокупность процессов, работ и задач, включающая в себя разработку, эксплуатацию и сопровождение программного средства, начиная с анализа его концепции или потребности в заказе до прекращения его использования.

СТБ ИСО/МЭК 12207–2003 – стандарт, принятый в 2004 году, который определяет жизненный цикл ПС и систем в целом. При этом под системой подразумевается комплекс, состоящий из процессов, технических и программных средств и персонала, обладающий возможностью удовлетворять установленным потребностям или целям.  
  
В соответствии с данным стандартом ПС и системы имеют трёхуровневую структуру – Процессы -> Работы -> задачи. Всего 17 процессов, 74 работы и 232 задачи.

Есть 3 наиболее важные группы процессов ЖЦ: Основные процессы ЖЦ, Вспомогательные процессы ЖЦ, Организационные процессы ЖЦ.

N2. СТБ ИСО/МЭК 12207-2003. Основные процессы жизненного цикла программных средств.

К основным процессам ЖЦ относятся:  
а) **Заказ**: определяет работы и задачи заказчика. Процесс заказа состоит из определения потребностей заказчика в системе, программном продукте или программной услуге, подготовки и выпуска заявки на подряд, выбора поставщика и управления процессом заказа до завершения приемки системы, программного продукта или программной услуги. Имеет 5 задач.

б) **Постановка**: определяет работы и задачи поставщика. Сначала идёт решение о подготовке предложения в ответ на заявку на подряд, присланную заказчиком, или с подписания договора с заказчиком на поставку системы, программного продукта (ПП) или программной услуги. Затем определяются процедуры и ресурсы, необходимые для управления и обеспечения проекта, включая разработку проектных планов и их выполнение. Имеет 7 работ.

в) **Разработка**: определяет работы и задачи разработчика. Данный процесс включает работы по анализу требований, проектированию, программированию, сборке, тестированию, вводу в действие и приемке программного продукта или системы. Имеет 13 работ.

г) **Эксплуатация**: определяет работы и задачи оператора. Данный процесс включает эксплуатацию программного продукта и поддержку пользователей в процессе эксплуатации. Процесс эксплуатации состоит из 4х работ.

д) **Сопровождение**: определяет работы и задачи персонала сопровождения и реализуется при модификациях программного продукта. Цель процесса – изменение существующего ПП при сохранении его целостности. Имеет 6 работ.

N 3. СТБ ИСО/МЭК 12207-2003. Процесс разработки и его структура (работы 1-6).

1) Подготовка процесса разработки.

2) Анализ требований к системе.

3) Проектирование системной архитектуры.

4) Анализ требований к ПС.

5) Проектирование программной архитектуры.

6) Технологическое проектирование П.С.

N4. СТБ ИСО/МЭК 12207-2003. Процесс разработки и его структура (работы 7-13).

7) Проектирование и тестирование ПС.

8) Сборка ПС.

9) Квалификационные испытания ПС.

10) Сборка системы.

11) Квалификационные испытания системы.

12) Вход в действия ПС.

13) Обеспечения приёмки ПС.

N5. СТБ ИСО/МЭК 12207-2003. Вспомогательные процессы жизненного цикла программных средств.

К вспомогательным процессам ЖЦ относятся:  
а) **Документирование**: процессом формализованного описания информации, созданной в процессе или работе жизненного цикла. Он включает планирование, проектирование, разработку, выпуск, редактирование, распространение и сопровождение документов по программному продукту. б) **Управление конфигурацией**: является процессом применения административных и технических процедур на всем протяжении ЖЦ ПС для определения состояния (базовой линии) программных объектов в системе, управления их изменениями и выпуском. Имеет 6 работ.

в) **Обеспечения качества**:является процессом обеспечения гарантий того, что программные продукты и процессы в жизненном цикле проекта соответствуют требованиям и планам. Данный процесс должен быть независимым от субъектов, участвующих в проекте. Это позволяет достичь объективности процесса.

г) **Верификации**: является процессом определения того, что программные продукты функционируют в полном соответствии с требованиями и условиями, реализованными в предшествующих работах

д) **Аттестации**: является процессом определения полноты соответствия установленных требований, созданной системы или программного продукта их функциональному назначению

е) **Совместного анализа**: является процессом оценки состояний и результатов работ по проекту. Совместные анализы проводятся в течение всего договора и применяются как на уровне управления проектом, так и на уровне его технической реализации. Данный процесс может выполняться двумя любыми сторонами, участвующими в договоре, когда одна сторона (анализирующая) проверяет другую (анализируемую).

ж) **Аудита**: является процессом определения соответствия требованиям, планам и условиям договора. Данный процесс может выполняться двумя сторонами, участвующими в договоре, когда одна сторона (ревизующая) проверяет другую сторону (ревизуемую). з) **Решения** **проблем**: является процессом анализа и решения проблем (включая обнаруженные несоответствия), которые обнаружены в ходе выполнения разработки, эксплуатации, сопровождения или других процессов.

N6. СТБ ИСО/МЭК 12207-2003. Организационные процессы жизненного цикла программных средств.

а) **Управление**: состоит из общих работ и задач, которые могут быть использованы любой стороной, управляющей соответствующим процессом. За управление продуктом, проектом, работами и задачами основных и вспомогательных процессов отвечает администратор. Имеет 5 работ.

б) **Создание инфраструктуры**: является процессом установления и сопровождения инфраструктуры, необходимой для любого другого процесса. Инфраструктура содержит технические и программные средства, инструментальные средства, методики, стандарты и условия для разработки, эксплуатации или сопровождения.

в) **Процесс усовершенствования**: является процессом установления, оценки, измерения, контроля и улучшения любого процесса жизненного цикла программных средств. Имеет 3 работы.

г) **Процесс обучения**: является процессом обеспечения первоначального и продолженного обучения персонала работам по заказу, поставке, разработке, эксплуатации или сопровождению программного проекта.

N7. ISO/IEC 12207: 2008. Процесс разработки программных средств.

Группы процессов жизненного цикла включают в себя:

**1. Процессы соглашения**

**2. Процессы организационного обеспечения проекта**

**3. Процессы проекта**

**4. Технические процессы**

**5. Процессы реализации программных средств**

**6. Процессы поддержки программных средств**

**7. Процессы повторного применения программных средств**

.

N8. Качество программных средств. Основные понятия и определения(атрибут,качество,мера,метрика,модель качества,характеристика качества программных средств, подхарактеристика качества программных средств, ошибка, отказ, ранжирование, уровень ранжирования, шкала).

**Атрибут**: измеримое физическое или абстрактное свойство продукта. Атрибуты могут быть внешними или внутренними.

**Качество:** совокупность характеристик ПП, относящаяся к его способности удовлетворять установленные и подразумеваемые потребности.

**Критерий оценки качества**: совокупность принятых в установленном порядке правил и условий, с помощью которых устанавливается приемлемость общего качества программного продукта.

**Мера**: число или категория, присваиваемая атрибуту продукта путем измерения.

**Мера косвенная**: мера атрибута, которая получена из мер одного или большего числа других атрибутов.

**Мера прямая**: мера атрибута, которая не зависит от меры любого другого атрибута.

**Метрика**: определенный метод и шкала измерения. Метрики могут быть внутренними, внешними или метриками качества в использовании; прямыми или косвенными. Метрики включают методы для категоризации качественных данных (данных, которые нельзя измерить количественно).

**Модель качества**: набор характеристик и связей между ними, обеспечивающий основу для определения требований к качеству и для оценки качества.

**Отказ:** прекращение способности продукта выполнять требуемую функцию или его неспособность работать в пределах заданных ограничений.

**Оценка качества**: Систематическое исследование степени, в которой продукт способен к выполнению указанных требований.

**Ошибка**: некорректный шаг, процесс или определение данных в программе.

**Подразумеваемые потребности**: потребности, которые не были установлены, но являются действительными потребностями при использовании продукта в конкретных условиях.

**Подхарактеристика качества ПС**: это характеристика качества программного средства, входящая в состав другой характеристики качества.

**Показатель качества ПС** – признак, определяющий свойство программного средства, которое может быть соотнесено с некоторой характеристикой качества.

**Ранжирование:** действие по отнесению измеренного значения к соответствующему уровню ранжирования.

**Уровень ранжирования**: точка на порядковой шкале, которая используется для категоризации шкалы измерения. Уровень оценки позволяет ранжировать программное обеспечение в соответствии с установленными или подразумеваемыми потребностями. Соответствующие уровни ранжирования могут быть связаны с различными точками зрения на качество, например, пользователей, администраторов или разработчиков.

**Характеристика качества ПС**: набор свойств программного средства, с помощью которых описывается и оценивается его качество.

**Шкала**: набор значений с определенными свойствами.

N9 Типы шкал. Примеры. шкал.

**Шкала**: набор значений с определенными свойствами.

При оценке качества используются следующие типы шкал:

**Номинальная** – соответствует набору категорий; классифицирует программы по признаку наличия или отсутствия некоторого свойства без учета градаций (например «да», «нет»);

**Порядковая** (упорядоченная) – соответствует упорядоченному набору делений шкалы; позволяет ранжировать свойства путем сравнения с опорными значениями; имеет небольшое количество делений (например, шкала с четырьмя градациями «отлично», «хорошо», «удовлетворительно», «неудовлетворительно», с двумя градациями «удовлетворительно», «неудовлетворительно»);

**Интервальная** – соответствует упорядоченной шкале с равноудаленными делениями; обычно содержит достаточно большое количество делений с количественными значениями (например, шкала с делениями 0, 1, 2, …, 10);

**Относительная** – соответствует упорядоченной шкале с равноудаленными делениями, оцененными в относительных единицах относительно некоторой абсолютной величины (обычно в диапазоне от 0 до 1).

N10. СТБ ИСО/МЭК 9126-2003. Метод оценки качества программных средств.

СТБ ИСО/МЭК 9126–2003 определяет метод оценки качества ПС, основанный на трехуровневой иерархической модели качества. На первом уровне модели находятся шесть характеристик качества.

Второй уровень составляют подхарактеристики и третий – метрики качества.

Процесс оценки состоит из трех стадий: определение требований к качеству ПС, подготовка к оцениванию и процедура оценивания. Данный процесс может применяться после любой подходящей работы жизненного цикла для каждого компонента программного продукта.

**Стадия 1. Определение требований к качеству.** Целью данной стадии является установка требований в терминах характеристик и подхарактеристик качества. Требования выражают потребности

внешнего окружения ПС и должны быть определены до начала разработки. Так как ПС разделяется на компоненты, то требования для ПС в целом могут отличаться от требований для отдельных компонентов.

**Стадия 2. Подготовка к оцениванию.** Целью второй стадии является подготовка основы для оценивания. Данная стадия состоит из трех этапов.

***Этап 2.1. Выбор метрик качества.*** С учетом регламентированной в СТБ ИСО/МЭК 9126–2003 иерархической модели качества уровень характеристик качества ПС определяется уровнем входящих в них подхарактеристик, а значения подхарактеристик в свою оче-

редь определяются значениями входящих в них метрик.

В стандарте СТБ ИСО/МЭК 9126–2003 набор рекомендуемых метрик от-

сутствует. Поэтому существует потребность в установлении метрик, которые соотносятся с подхарактеристиками, а следовательно, и с характеристиками ПС. Каждый количественный признак и каждое количественно оцениваемое взаимодействие ПС с его окружением, которые соотносятся с характеристикой, могут быть приняты в качестве метрики. Метрики, используемые в процессе разработки, должны быть соотнесены с соответствующими метриками пользователя, потому что метрики пользователя являются решающими.

***Этап 2.2. Определение уровней ранжирования.*** Для измерения количественных признаков ПС используются метрики качества. Измеренные значения отображаются на некоторой шкале. Данные значения не показывают уровень удовлетворения требований к качеству ПС. Для этой цели шкалы метрик должны быть разделены на диапазоны, соответствующие различным степеням удовлетворения требований.

В стандарте ISO/IEC 14598–1:1999 приведен пример следующих диапазонов ранжирования :

• разделение шкалы на две категории: неудовлетворительно и удовлетворительно;

• разделение шкалы на четыре категории (отлично, хорошо, удовлетворительно, неудовлетворительно), ограниченные соответственно запланированным уровнем, текущим уровнем для существующего или альтернативного продукта и уровнем худшего случая.

Текущий уровень определяется для управления тем, чтобы новая система не становилась хуже по сравнению с существующей. Запланированный уровень определяет уровень, который считается достижимым при доступных ресурсах.

Уровень худшего случая определяет границу принятия пользователем в случае, если изделие не удовлетворяет запланированному уровню. Так как качество ПС связано с конкретными потребностями, общие уровни ранжирования невозможны и должны определяться для каждого конкретного оценивания.

***Этап 2.3. Определение критерия оценки.*** Для определения общего качества ПС должна быть учтена вся совокупность результатов оценивания различных метрик. Оценщик должен подготовить для этого процедуры, используя, например, таблицы решений или средние

взвешенные значения. Обычно при этом учитываются и другие аспекты, такие как время и стоимость, которые являются косвенными факторами качества ПС.

**Стадия 3. Процедура оценивания.**

Последняя стадия модели процесса оценивания реализуется тремя этапами: «Измерение», «Ранжирование» и «Оценка».

***Этап 3.1. Измерение.*** Для измерения выбранные метрики применяются к ПС. Результатом являются значения в масштабах метрик.

***Этап 3.2. Ранжирование.*** На этапе ранжирования устанавливается уровень ранжирования для измеренного значения.

***Этап 3.3. Оценка.*** Оценка является последним этапом процесса оценивания ПС, на котором обобщается множество установленных уровней. Результатом является заключение о качестве ПС (приемлемый или неприемлемый уровень качества).

К недостаткам данного метода оценки качества следует отнести отсутствие рекомендуемых вариантов метрик и представление метода лишь в общем виде (в виде модели). Это затрудняет его конкретное использование.

N10 UPD  
10. СТБ ИСО/МЭК 9126-2003. Метод оценки качества программных средств.

Метод основан на 3-х уровневой модели качества:

1)характеристики;

2)подхарактеристики;

3)метрики (не обязательный уровень).

Метод включает **3 стадии**:

**1**. **Определение требований качества.**

Требования выражают потребности во внешнем окружении ПС и должны быть оценены до начала разработки.

**2. Подготовка к оцениванию.**

Этапы:

**1)Выбор метрик качества.** Набор рекомендуемых метрик отсутствует, поэтому существует потребность в их определении. Каждый количественный признак и количественное взаимодействие ПС с его окружением соотносится с характеристикой качества.

**2)Определение уровней ранжирования.** Для измерения количественных признаков ПС используются метрики качества. Измеренные значения отображаются на некоторую шкалу, которая зависит от предметной области:

Запланированный уровень Превышения ожидания Удовлетворительно

Измеренный уровень Целевой уровень Неуодвлетворительно

Текущий уровень Минимальный уровень

Худший случай

**3.Оценивание.**

N11. ISO/IEC 9126-1:2001. Связь качества программных средств с их жизненным циклом.

В течение жизненного цикла программного средства его качество изменяется. Требуемое качество, определенное в начале ЖЦ, отличается от актического качества поставленного продукта. Существует несколько точек зрения на качество ПС в течение его ЖЦ. От их выбора зависит как оценка качества ПС, так и управление качеством на каждой стадии жизненного цикла. С учетом этого для различных стадий ЖЦ стандарт ISO/IEC 9126–1:2001 определяет следующие виды качества программных средств [49]:

• *потребности пользователя в качестве* определяются как требования к качеству, выраженные в терминах метрик качества в использовании, внешних и иногда внутренних метрик; эти требования должны применяться как критерии при аттестации продукта; получение ПП, удовлетворяющего потребностям пользователя, обычно требует итеративного подхода к разработке программного средства с постоянной обратной связью с потенциальным пользователем;

• *требования к внешнему качеству* определяют требуемый уровень качества с внешней точки зрения; они включают требования, вытекающие из потребностей пользователя в качестве, включая требования к качеству в использовании; требования к внешнему качеству применяются как цель при аттестации продукта на различных стадиях разработки; данные требования для всех характеристик качества, определенных в ISO/IEC 9126–1:2001, во-первых, должны быть выражены в спецификации требований к качеству, используя внешние метрики, во-вторых, должны быть преобразованы в требования к внутреннему качеству и, втретьих, должны использоваться как критерии при оценке продукта;

• *требования к внутреннему качеству* определяют требуемый уровень качества с внутренней точки зрения на программный продукт; они спользуются для определения свойств промежуточных продуктов разработки; промежуточные продукты могут включать статические и динамические модели, другие документы и исходный код ПП; требования к внутреннему качеству могут использоваться как цель при аттестации продукта на различных стадиях разработки; они могут использоваться для определения стратегий разработки и критериев оценки и верификации в течение разработки; требования к внутреннему качеству должны определяться количественно, используя внутренние метрики;

• *внутреннее качество* – совокупность характеристик программного продукта с внутренней точки зрения; внутреннее качество измеряется с помощью внутренних метрик и оценивается по отношению к требованиям к нутреннему качеству; отдельные элементы качества ПП могут улучшаться при реализации кода, проверке или тестировании, но фундаментальная основа качества программного продукта, представленная внутренним качеством, остается неизменной до повторного проектирования;

• *оценочное (или прогнозируемое) внешнее качество* – оцененное или предсказанное качество конечного программного продукта на каждой стадии процесса разработки для каждой характеристики качества, основанное на знании внутреннего качества;

• *внешнее качество* – совокупность характеристик программного продукта с внешней точки зрения; это качество, измеряемое и оцениваемое на основе внешних метрик при выполнении ПП во время тестирования (испытаний) в моделируемой среде с моделируемыми данными или во время эксплуатации;

• *оценочное (или прогнозируемое) качество в использовании* – оцененное или предсказанное качество конечного программного продукта на каждой стадии процесса разработки для каждой характеристики качества в использовании, основанное на знании внутреннего и внешнего качества;

• *качество в использовании* – качество программного продукта, применяемого в заданной среде и заданном контексте использования, с точки зрения пользователя; оно оценивается на основе метрик качества в использовании и в первую очередь измеряет степень достижения пользователем своих целей в конкретной среде, а не свойства самого ПП; пользователь оценивает только те атрибуты программного продукта, которые он применяет в своих задачах.

N12. ISO/IEC 9126-1:2001Модель внешнего и внутреннего качества программных средств. Характеристики качества

В стандарте ISO/IEC 9126–1:2001 регламентированы две части модели качества программных средств: модель внутреннего и внешнего качества и модель качества в использовании. Данные модели различаются в зависимости от представления качества в ЖЦ ПС. Эти модели могут быть использованы, например, в следующих случаях:

• проверка полноты определения требований;

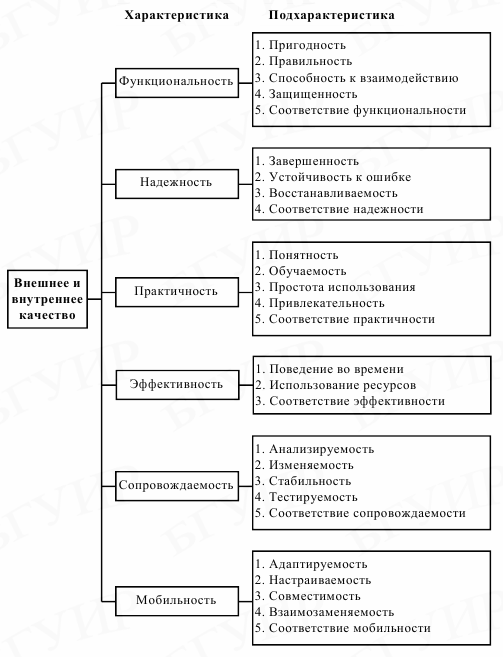
• определение требований к ПС;

• определение целей проектирования ПС;

• определение целей испытаний ПС;

• определение критериев обеспечения качества;

• определение критериев приемки завершенного ПС.



UPD

12. ISO/IEC 9126-1:2001Модель внешнего и внутреннего качества

программных средств. Характеристики качества.

**3-х уровневая модель качества:**

1)характеристики;

2)подхарактеристики;

3)метрики (не обязательный уровень).

**Характеристики и подхарактеристики:**

1)Функциональность

(подхарактеристики: )

-пригодность

-правильность

-способность к взаимодействию

-согласованность

-защищенность

2)Надежность

-завершенность

-устойчивость к ошибке

-восстанавливаемость

3)Удобство использования

-понятность

-обучаемость

-простота использования

4)Эффективность

-время+емкость

-ресурсоемкость

5)Сопровождаемость

-анализируемость

-изменяемость

-стабильность

-тестируемость

6)Мобильность

-адаптируемость

-простота установки

-согласованность

-взаимозаменяемость

**Внутренние метрики** – измеряющие собственные свойства ПС. Измеряются в процессе разработки, на основе спецификаций, требований, результатов проектирования, исходного кода и др. документации.

N13. ISO/IEC 9126-1:2001Характеристики Функциональность и Надежность. Подхарактеристики Функциональности и Надежности.

**Функциональность** – способность программного продукта обеспечивать функции, удовлетворяющие установленные и подразумеваемые потребности при применении программного средства в заданных условиях. Эта характеристика определяет, что делает ПС в соответствии с потребностями.

**Пригодность** – способность программного продукта обеспечивать набор функций, соответствующий специфическим задачам и целям пользователей.

**Правильность** – способность программного продукта обеспечивать правильные или приемлемые результаты или эффекты с необходимой степенью точности.

**Способность к взаимодействию** – способность программного продукта взаимодействовать с одной или несколькими заданными системами.

**Защищенность** – способность программного продукта защищать информацию и данные так, чтобы несанкционированные пользователи или системы не могли прочитать или модифицировать их, а санкционированные пользователи или системы не могли получить отказа в доступе к ним.

**Соответствие функциональности** – способность программного продукта соответствовать стандартам, соглашениям или нормам законов, связанным с функциональностью.

**Надежность** – способность программного продукта поддерживать заданный уровень качества функционирования при его использовании в заданных условиях. Ограничения надежности в процессе эксплуатации вызваны ошибками в требованиях, проектировании и кодировании.

**Завершенность** – способность программного продукта избегать отказов вследствие ошибок в программах.

**Устойчивость к ошибке** – способность программного продукта поддерживать заданный уровень качества функционирования в случаях ошибок в программах или нарушения заданного интерфейса ПП.

**Восстанавливаемость** – способность программного продукта восстанавливать заданный уровень качества функционирования и данные, поврежденные в случае отказа. Одним из показателей восстанавливаемости является длительность восстановления.

**Готовность** – способность программного продукта быть в состоянии выполнять требуемую функцию в данный момент времени при заданных условиях использования.

N14. ISO/IEC 9126-1:2001 Характеристики Эффективность и Практичность. Подхарактеристики Эффективности и Практичности.

**Эффективность** – способность программного продукта обеспечить соответствующую производительность в зависимости от количества используемых вычислительных ресурсов в заданных условиях. Ресурсы могут включать другие программные продукты, конфигурацию программных и аппаратных средств системы и материалы.

**Поведение во времени** – способность программного продукта обеспечивать соответствующие времена отклика и обработки, а также пропускную способность при выполнении своих функций в заданных условиях.

**Использование ресурсов** – способность программного продукта использовать соответствующее количество всех типов ресурсов при выполнении своих функций в заданных условиях. В данной подхарактеристике человеческие ресурсы не учитываются.

**Соответствие эффективности** – способность программного продукта соответствовать стандартам и соглашениям, связанным с эффективностью.

**Практичность** – способность программного продукта быть понятным, изученным, использованным и привлекательным для пользователя при применении в заданных условиях.

**Понятность** – способность программного продукта, обеспечивающая понимание пользователем пригодности и способа использования программного средства для конкретных задач и условий применения.

**Обучаемость** – способность программного продукта, обеспечивающая изучение пользователем принципов его применения.

**Простота использования** – способность программного продукта, позволяющая пользователю эксплуатировать его и управлять им. На простоту использования влияют некоторые аспекты таких подхарактеристик, как пригодность, изменяемость, адаптируемость и простота внедрения. Простота использования зависит, например, от контролируемости и устойчивости к ошибке.

**Привлекательность** – способность программного продукта нравиться пользователю. Данная подхарактеристика связана со свойствами оформления ПП (например использование цветов, графики и т.п.).

**Соответствие практичности** – свойство программного продукта соответствовать стандартам, соглашениям и руководствам, связанным с практичностью.

N15. ISO/IEC 9126-1:2001 Характеристики Сопровождаемость и Мобильность. Подхарактеристики Сопровождаемости и Мобильности.

**Сопровождаемость** – способность программного продукта к модификации. Модификации могут включать исправления, усовершенствования или адаптацию ПС к изменениям в среде применения, в требованиях и функциональных спецификациях

**Анализируемость** – способность программного продукта к

диагностике его недостатков или причин отказов или к идентификации его час-

тей, которые должны быть модифицированы.

**Изменяемость** – способность программного продукта к

реализации заданной модификации. Реализация включает проектирование, ко-

дирование и изменение документации.

**Стабильность** – способность программного средства предот-

вращать непредвиденные эффекты от его модификации.

**Тестируемость** – способность программного продукта к про-

верке результата модификации.

**Соответствие сопровождаемости** – способность программного продукта соответствовать стандартам или соглашениям, связанным с сопровождением.

**Мобильность** – способность программного продукта к переносу из одной среды в другую. Среда может включать организационное, аппаратное и программное окружение.

**Адаптируемость** – способность программного продукта к адаптации к различным окружающим средам без применения дополнительных действий или средств.

**Настраиваемость** – способность программного продукта устанавливаться в заданной среде окружения.

**Совместимость** – способность программного продукта к сосуществованию с другими независимыми программными средствами в общей среде, разделяя общие ресурсы.

**Взаимозаменяемость** – способность программного продукта к использованию вместо другого (заданного) ПП с той же целью и в той же среде. Например, для пользователя важна взаимозаменяемость новой версии ПП с его старой версией.

**Соответствие мобильности** – способность программного продукта соответствовать стандартам или соглашениям, связанным с мобильностью.

N16. ISO/IEC 9126-1:2001Модель качества в использовании. Характеристики качества в использовании.

**Качество в использовании** – это способность программного продукта позволять заданным пользователям достигать заданные цели с результативностью, продуктивностью, безопасностью и удовлетворением в заданном контексте использования.

**Результативность** – это способность программного продукта, позволяющая пользователям достигать заданные цели с точностью и полнотой в заданном контексте использования.

**Продуктивность** – это способность программного продукта, позволяющая пользователям расходовать количество ресурсов, соответствующее результативности, достигаемой в заданном контексте использования. Ресурсы могут включать время выполнения задачи, усилия пользователя, материалы, стоимость использования.

**Безопасность** – это способность программного продукта достигать приемлемых уровней риска причинения вреда людям, бизнесу, программному обеспечению, имуществу или окружающей среде в заданном контексте использования. Обычно риски – это результат дефектов в функциональности.

**Удовлетворенность** – это способность программного продукта удовлетворять пользователя в заданном контексте использования. Удовлетворенность определяется реакцией пользователя на взаимодействие с программным продуктом и включает отношение к применению продукта.

N17.ISO/IEC 14598-1. Метод оценки качества программных средств.

Основан на иерархической модели качества.

Процесс оценки состоит из четырех стадий: установка требований к оценке, определение оценки, проектирование оценки и выполнение оценки. Данный процесс может применяться после любой подходящей работы жизненного цикла для промежуточных или конечного продуктов разработки (сопровождения).

**Стадия 1. Установка требований к оценке**

**Этап 1.1. Установка цели оценки**

Общей целью оценки качества ПС является поддержка разработки и приобретения ПС, удовлетворяющего заявленные и подразумеваемые потребности пользователей. Конечная цель состоит в том, чтобы гарантировать, что продукт обеспечивает требуемое качество.

Целью оценки качества промежуточного продукта может быть:

* решение о принятии промежуточного продукта от субподрядчика;
* решение о завершении процесса и передаче продукта следующему процессу;
* прогноз или предварительная оценка качества конечного продукта;
* сбор информации о промежуточных продуктах для контроля и управления процессом.

Целью оценки качества конечного продукта может быть:

* решение о принятии продукта;
* решение о выпуске продукта;
* сравнение продукта с конкурентными продуктами;
* выбор продукта из числа альтернативных продуктов;
* оценка положительного и отрицательного результата использования продукта;
* решение о сроках улучшения или замены продукта.

Качество программного средства может оцениваться в процессах жизненного цикла:

В процессе заказа ПП, В процессе поставки, В процессе разработки, В процессе эксплуатации, В процессе сопровождения.

**Этап 1.2. Идентификация типов продуктов**

Тип оцениваемого промежуточного или конечного программного продукта зависит от цели оценки и стадии ЖЦ ПС. Например, в процессе разработки интерес представляет оценка промежуточного продукта. На ранних этапах процесса разработки это может быть спецификация требований, архитектура ПП, технический проект ПП, исходные коды модулей. Для них выполняется оценка внутреннего качества с целью прогноза внешнего качества.

На последующих этапах процесса разработки промежуточными продуктами являются исполнимые коды модулей, компонентов и других промежуточных продуктов сборки, а также конечный программный продукт. Для них выполняется оценка внешнего качества в моделируемой среде с моделируемыми данными с целью прогноза внешнего качества ПП в среде эксплуатации.

В процессе эксплуатации оценивается система, частью которой является программный продукт. В этом случае может выполняться, во-первых, оценка внешнего качества при использовании ПС в окружающей среде с целью подтверждения соответствия требованиям к внешнему качеству и прогноза качества ПП в использовании; и, во-вторых, оценка качества в использовании для подтверждения удовлетворения потребностей пользователя в выполнении заданных задач в заданных аппаратных и операционных средах.

**Этап 1.3. Определение модели качества**

На данном этапе, исходя из типов оцениваемых продуктов, выбирается соответствующая модель качества (внутреннего, внешнего или качества в использовании). Основу модели составляет общая модель качества из стандарта ISO/IEC 9126–1:2001. Выбранная модель должна быть адаптирована с учетом целей оценки и конкретных требований к качеству оцениваемого продукта. В процессе адаптации из модели качества стандарта ISO/IEC 9126–1:2001 должны быть выбраны соответствующие характеристики и подхарактеристики, которые будут оцениваться.

**Стадия 2. Определение оценки**

**Этап 2.1. Выбор метрик**

На данном этапе, исходя из разработанной модели качества программного средства, выбираются соответствующие метрики качества. Каждое измеримое внутреннее или внешнее свойство продукта, влияющее на значение характеристики или подхарактеристики качества, может быть установлено как метрическое. При выборе метрик следует учитывать простоту и экономность их использования.

На выбор метрик оказывает влияние также тип требуемых измерений, который зависит от цели оценки. Если целью оценки является исправление недостатков разработки, то для контроля достаточно выполнить несколько измерений продукта. Данные измерения могут быть дополнены мерами из имеющихся контрольных списков или мнениями экспертов.

Если же целью оценки является сравнение готовых программных продуктов или анализ соответствия продукта требованиям к качеству, то необходимо использовать строгие метрики с достаточной точностью измерений. При этом должны быть учтены ошибки в измерениях, вызванные человеческим фактором или инструментом измерения.

Кроме того, для осуществления прогноза качества программного средства важно, чтобы выбранные внутренние метрики коррелировали с некоторыми аспектами внешнего качества, а выбранные внешние метрики – с аспектами качества в использовании.

Этап 2.2. Установка уровней оценки (ранжирования) для метрик

Значение, измеренное с помощью метрики, имеет некоторую величину, которая сама по себе не отражает степень удовлетворения результатом измерения. Поэтому шкала измерений должна быть разделена на диапазоны, соответствующие различным степеням удовлетворения требований.

**Этап 2.3. Установка критериев для оценки**

Чтобы оценить качество продукта, необходимо некоторым образом объединить результаты оценки его различных характеристик. С этой целью должна быть разработана процедура, включающая отдельные критерии для различных характеристик качества. Каждая из характеристик может быть определена в терминах единственной подхарактеристики или средневзвешенной комбинации подхарактеристик. Процедура обычно включает и другие аспекты, такие как время и стоимость, которые вносят вклад в оценку качества программного продукта в конкретной среде окружения.

**Стадия 3. Проектирование оценки**

**Этап 3.1. Разработка плана оценки**

В плане оценки должны быть описаны методы оценки и график действий по оценке. Действия по оценке для разработчика, заказчика и оценщика описаны в стандартах ISO/IEC 14598–3:2000, ISO/IEC 14598–4:1999 и ISO/IEC14598–5:1998 [20 – 22]. План оценки должен быть согласован с планом количественной оценки качества, регламентированным в ISO/IEC 14598–2:2000 [19].

**Стадия 4. Выполнение оценки**

**Этап 4.1. Выполнение измерений**

Для измерения выбранные метрики применяются к программному продукту. Результатом являются значения на шкалах метрик.

**Этап 4.2. Сравнение с уровнями оценки (ранжирование)**

На данном этапе измеренные значения сравниваются с уровнями ранжирования (например, как показано на рис. 36).

**Этап 4.3. Оценка результатов**

Оценка результатов – заключительный этап процесса оценки ПС. На данном этапе с учетом решений, принятых при выполнении этапа 2.3, суммируются оцененные уровни метрик. Результатом является заключение о степени удовлетворения продуктом требований к качеству. Полученное в итоге качество сравнивается с другими аспектами, такими как время и стоимость.

На основании оценки и организационных критериев принимается административное решение относительно принятия или отклонения, выпуска или невыпуска программного продукта.

N18. ISO/IEC 9126-1:2001. Свойства и критерии обоснованности метрик качества программных средств.

1) **надежность**; надежность связана со случайной ошибкой; метрика свободна от случайной ошибки, если случайные изменения не влияют на результаты метрики;

2) **повторяемость**; повторное использование метрики для того же продукта теми же специалистами по оценке, используя ту же спецификацию оценки (включая ту же окружающую среду), тот же тип пользователей и окружения, должно привести к тем же результатам с соответствующими допусками; соответствующие допуски должны учитывать такие компоненты, как усталость и результат накопленных познаний;

3) **однотипность**; применение метрики для того же продукта различными специалистами по оценке, используя ту же спецификацию оценки (включая ту же окружающую среду), тот же тип пользователей и окружения, должно привести к тем же результатам с соответствующими допусками;

4) **применимость**; метрика должна четко указывать условия (например, наличие определенных атрибутов), которые ограничивают её употребление;

5) **показательность**; это способность метрики идентифицировать части или элементы программы, которые должны быть улучшены, на основании сравнения измеренных и ожидаемых результатов;

6) **корректность**; метрика должна обладать следующими свойствами:

• объективность; результаты метрики и её входные данные должны быть основаны на фактах и не подвластны чувствам или мнениям специалистов по оценке или тестированию (исключая метрики удовлетворенности или привлекательности, с помощью которых измеряются чувства и мнения пользователя);

• беспристрастность; измерение не должно быть направлено на получение какого-либо специфического результата;

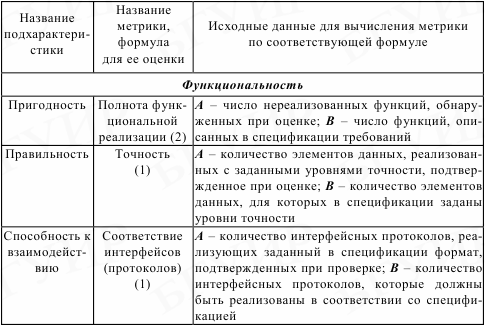
• адекватность точности; точность определяется при проектировании метрики и особенно при выборе описаний фактов, используемых как основа для метрики; разработчик метрики должен описать точность и чувствительность метрики;

7) **значимость**; измерение должно давать значащие результаты, касающиеся поведения программы или характеристик качества.

19. ISO/IEC TR 9126-3:2003. Внутренние метрики Функциональности и Надежности программных средств.

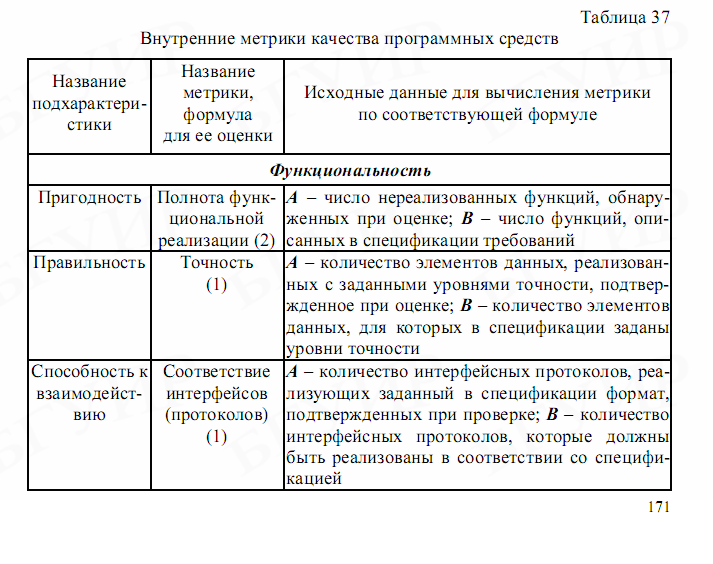
**Внутренние метрики функциональности** предназначены для предсказания того, удовлетворяет ли разрабатываемый программный продукт требованиям к функциональности и редполагаемым потребностям пользователя.

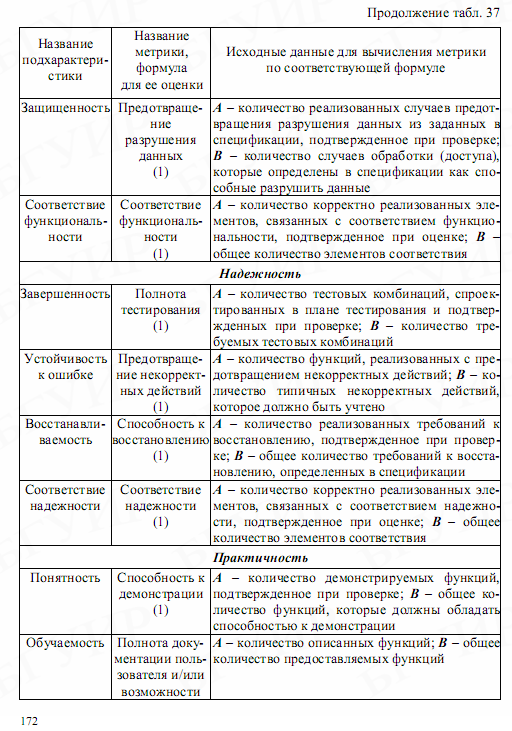
**Внутренние метрики надежности** используются во время разработки программного продукта для предсказания того, удовлетворяет ли ПП заявленным потребностям в надежности.

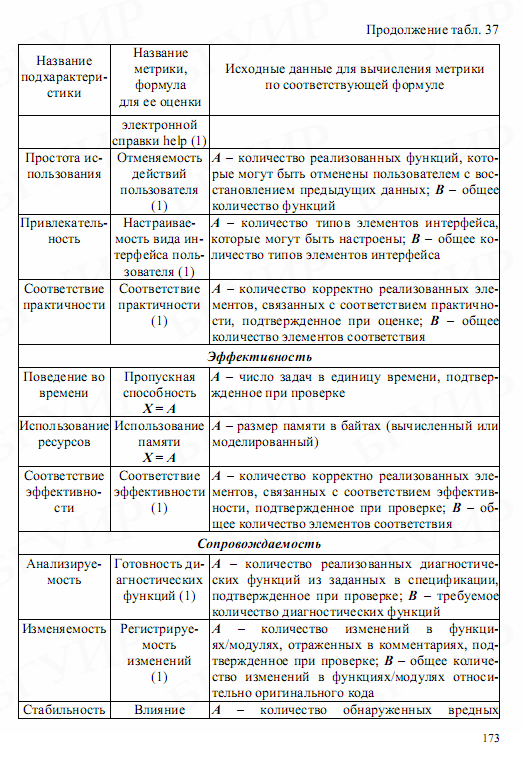


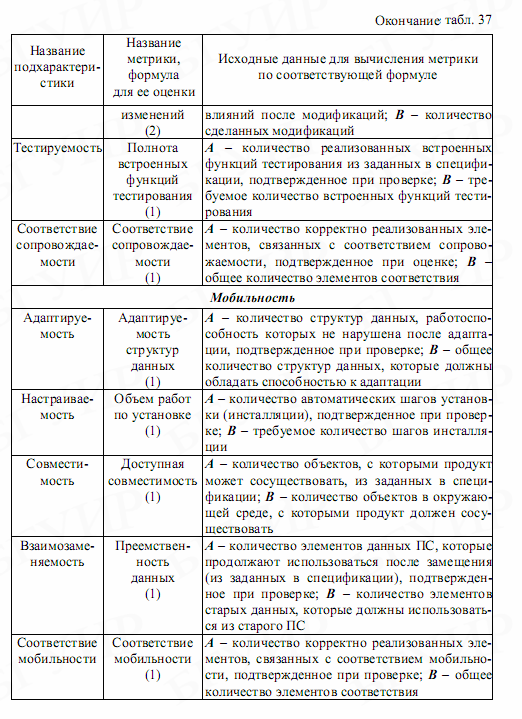


N20. ISO/IEC TR 9126-3:2003. Внутренние метрики Эффективности , Практичности, Сопровождаемости и Мобильности программных средств.









21. ISO/IEC TR 9126-2:2003. Внешние метрики Эффективности, Практичности, Сопровождаемости и Мобильности программных средств.

**Внешние метрики эффективности** должны измерять такие атрибуты, как характер изменения затрат времени и использования ресурсов компьютерной системы, включающей ПС, во время тестирования или эксплуатации.

**Внешние метрики практичности** показывают, в какой мере программное средство может быть понято, изучено, управляемо, привлекательно и соответствует договоренностям и руководствам по практичности.

**Внешние метрики сопровождаемости** измеряют такие атрибуты, как поведение персонала сопровождения, пользователя или системы, включающей ПС, при модификации ПС во время тестирования или сопровождения.

**Внешние метрики мобильности** измеряют такие атрибуты, как поведение оператора или системы при проведении работ по переносу.

N22. ISO/IEC TR 9126-2:2003. Внешние метрики Функциональности и Надежности программных средств.

**Внешние метрики функциональности** должны измерять свойства (атрибуты) функционального поведения системы, содержащей ПС.

**Внешние метрики надежности** должны измерять свойства, связанные с поведением системы, содержащей ПС, во время тестирования, чтобы показать степень надежности ПС в системе в процессе эксплуатации.

N23. ISO/IEC TR 9126-4:2004. Метрики качества программных средств в использовании.

**Метрики результативности** оценивают, достигают ли задачи, выполняемые пользователем, заданных целей с точностью и полнотой в заданном контексте использования.

**Метрики продуктивности** оценивают ресурсы, которые затрачивают пользователи в соответствии с достигнутой результативностью в заданном контексте использования.

**Метрики безопасности** оценивают уровень риска причинения вреда людям, бизнесу, программному обеспечению, имуществу или окружающей среде в заданном контексте использования.

**Метрики удовлетворенности** оценивают отношение пользователя к использованию продукта в заданном контексте использования.

24. ГОСТ 28195-99. Иерархическая модель качества программных средств (характеристики и подхарактеристики).

Основу описываемого метода оценки качества составляет четырёхуровневая иерархическая модель качества.

ГОСТ 28195-99 предлагает следующую терминологию для показателей качества каждого уровня:

**уровень 1** - факторы качества (в терминологии, принятой в международных стандартах соответствует характеристикам качества).

**уровень 2** - критерии качества (в международной терминологии - подхарактеристики качества )

**уровень 3** - метрики (соответствует международной терминологии)

**уровень 4** - оценочные элементы или единичные показатели (отсутствует в международных стандартах)

Показатели качества представляют собой иерархическую многоуровневую систему, в которой показатели вышестоящих уровней определяются через показатели нижестоящих уровней, и лишь на последнем уровне оценка значений показателей осуществляется на основе информации, относящейся непосредственно к ПС.

**Характеристики и подхарактеристики:**

1)Надеждность

(подхарактеристики: )

-устойчивость функционирования

-работоспособность

2) Сопровождаемость

-структурность

-простота конструкций

-наглядность

-повторяемость

-полнота документации

3)Удобство использования:

-легкость освоения

-доступность программных документов

-удобство эксплуатации и обслуживания

4)Эффективность

-уровень автоматизации

-временная эффективность

-ресурсоемкость

5)Универсальность(мобильность)

-гибкость

-мобильность

-модифицируемость

6)Функциональность

-полнота реализации

-согласованность

-логическая корректность

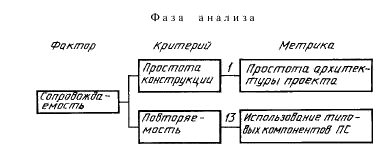
-проверенность

-защищенность

25. ГОСТ 28195-99. Процессы разработки и применения. Иерархическая

модель сопровождаемости для фазы анализа программных средств.



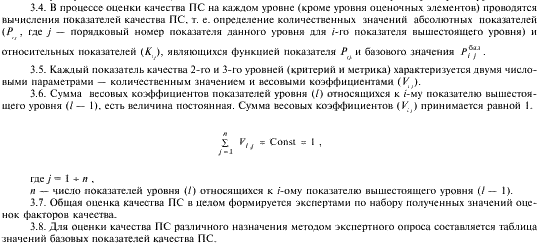


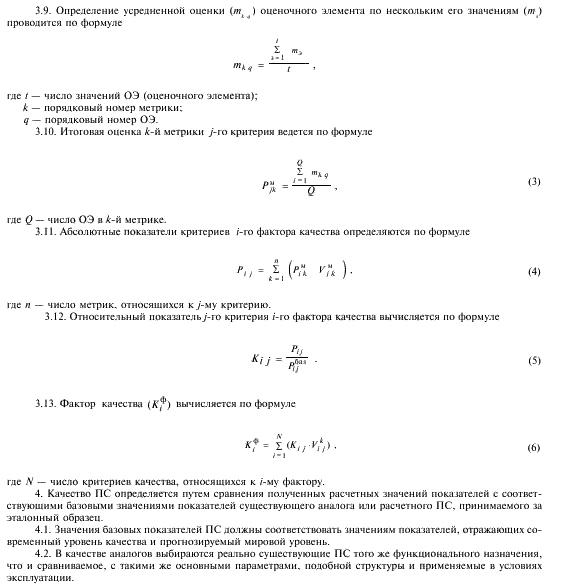
26. ГОСТ 28195-99. Метод количественной оценки качества программных

средств.

Оценку качества ПС проводят на фазах жизненного цикла. Она включает выбор номенклатуры показателей, их оценку и сопоставление значений показателей, полученных в результате сравнения с базовыми значениями.

Для показателей качества на всех уровнях (факторы, критерии, метрики, оценочные элементы) принимают единую шкалу оценки от нуля до единицы.

Показатели качества на каждом вышестоящем уровне (кроме уровня оценочных элементов) определяются показателями качества нижестоящего уровня. 



N27. Группы стандартов серии SQuaRE.

Группа стандартов ISO/IEC 14598, описывающих характеристики качества программного обеспечения и вместе составляющих модель качества, известную под названием SQuaRE (Software Quality Requirements and Evaluation).

Перечень стандартов, входящих в данную группу:

* − ISO/IEC 14598-1:1999.
* − ISO/IEC 14598-2:2000.
* − ISO/IEC 14598-3:2000.
* − ISO/IEC 14598-4:1999.
* − ISO/IEC 14598-5:1998.
* − ISO/IEC 14598-6:2001.

Стандарты ISO/IEC 9126 и ISO/IEC 14598 в настоящее время объединены в новую серию − ISO/IEC 25000 «Технология программного обеспечения. Требования и оценка качества программного продукта. Руководство» Quality Requirements and Evaluation или SQuaRE.

Стандарты серии SQuaRE состоят из 14 документов, сгруппированных в 5 секций в рамках модели SQuaRE.

Перечень секций:

− ISO/IEC 2500n – группа стандартов «**Управление качеством**»

− ISO/IEC 2501n – группа стандартов «**Модель качества**».

− ISO/IEC 2502n – группа стандартов «**Контроль качества**».

− ISO/IEC 2503n – группа стандартов «**Требования к качеству**».

− ISO/IEC 2504n – группа стандартов «**Оценка качества**».

Стандарты серии SQuaRE (ISO/IEC 25050 – ISO/IEC 25099) содержат в себе стандарты и (или) технические отчеты по качеству программного продукта, которые относятся к особым прикладным областям применения или дополняют один либо несколько стандартов.

N30. Основные понятия в области технического нормирования, стандартизации и оценки соответствия.

5 января 2004 г. в Республике Беларусь приняты и введены в действие Законы:

а) О техническом нормировании и стандартизации – регулирует отношения, возникающие при разработке и применении технических требований к продукции, процессам разработки, производства и эксплуатации продукта.

б) Об оценке соответствия требованиям технических нормативных правовых актов в области технического нормирования и стандартизации - определяет правовые и организационные основы оценки соответствия объектов оценки требованиям технических нормативных правовых

актов в области технического нормирования и стандартизации и направлен на

совершенствование механизма оценки в области подтверждения соответствия и

аккредитации с учетом международных принципов и требований Соглашения

по техническим барьерам в торговле ВТО.

N31. Оценка соответствия в Республике Беларусь.

Оценка соответствия в Беларуси выполняется на основе Закона Республики Беларусь № 269-З «Об оценке соответствия требованиям технических нормативных правовых актов в области технического нормирования и стандартизации».

**Оценка соответствия осуществляется в целях:**

• обеспечения защиты жизни, здоровья и наследственности человека, имущества и охраны окружающей среды;

• повышения конкурентоспособности продукции (услуг);

• создания благоприятных условий для обеспечения свободного перемещения продукции на внутреннем и внешнем рынках.

**Принципами оценки соответствия являются:**

• гармонизация с международными и межгосударственными (региональными) подходами в области оценки соответствия;

• обеспечение идентичности правил и процедур подтверждения соответствия продукции отечественного и иностранного производства;

• соблюдение требований конфиденциальности сведений, полученных при выполнении работ по оценке соответствия.

**Основными объектами оценки соответствия являются:**

• продукция;

• процессы жизненного цикла продукции (разработка, производство, эксплуатация, хранение, перевозка, реализация и утилизация);

• оказание услуг;

• система управления качеством;

• система управления окружающей средой;

• юридические лица;

• персонал.

**Субъектами оценки соответствия являются:**

• государство в лице уполномоченных государственных органов;

• аккредитованные органы по сертификации;

• аккредитованные испытательные лаборатории (центры);

• заявители на аккредитацию;

• заявители на подтверждение соответствия.

Законом предусмотрено два вида оценки соответствия:

**аккредитация**

**подтверждение соответствия**

N32. Основы сертификации программных средств.

Сертификация подтверждает, соответствует ли ПС нормам и требованиям. Есть 2 вида сертификации:  
а) **Обязательная**: подвергаются программные средства, выполняющие особо ответственные функции, в которых недостаточное качество и ошибки представляют потенциальную опасность для жизни, здоровья и наследственности человека, имущества и окружающей среды

б) **Добровольная**: применяется для удостоверения качества программного средства с целью повышения их конкурентоспособности и создания благоприятных условий для обеспечения свободного перемещения продукции на внутреннем и внешнем рынках.

Процесс сертификации:

1) Анализ работы.

2) Подача заявки на сертификацию.

3) Принятие органом сертификации решения.

4) Обзор версий ПС.

5) Сертификационные испытания ПС.

6) Анализ полученных результатов.

7) Выдача заявителю сертификата.