**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ**

**ЛУГАНСКОЙ НАРОДНОЙ РЕСПУБЛИКИ**

Государственное образовательное учреждение

среднего профессионального образования

Луганской Народной Республики

**АНТРАЦИТОВСКИЙ КОЛЛЕДЖ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И ЭКОНОМИКИ**

**ПРИЛОЖЕНИЕ 3.1 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ**

**ПРОВЕДЕНИЯ ЛАБОРАТОРНЫХ РАБОТ**

**по**

**МДК 03.03 Документирование и сертификация**

**Антрацит, 2020**

### **Лабораторная работа № 1**

### **Тема:** Выбор характеристик и мер качества программного средства

**Цель работы:** изучить основные характеристики и меры качества программных продуктов.

**Оборудование:** Компьютер,MS Office

**Теоретические данные**

Одной из важнейших проблем обеспечения качества программных средств является формализация характеристик качества и методология их оценки. Для определения адекватности качества функционирования, наличия технических возможностей программных средств к взаимодействию, совершенствованию и развитию необходимо использовать стандарты в области оценки характеристик их качества.

Показатели качества программного обеспечения устанавливают ГОСТ 28.195-89 «Оценка качества программных средств. Общие положения» и ГОСТ Р ИСО/МЭК 9126 «Информационная технология. Оценка программной продукции. Характеристика качества и руководства по их применению». Одновременное существование двух действующих стандартов, нормирующих одни и те же показатели, ставит вопрос об их гармонизации. Ниже рассмотрим каждый из перечисленных стандартов.

**ГОСТ 28.195-89 «Оценка качества программных средств. Общие положения»** устанавливает общие положения по оценке качества программных средств, номенклатуру и применяемость показателей качества.

Оценка качества ПС представляет собой совокупность операций, включающих выбор номенклатуры показателей качества оцениваемого ПС, определение значений этих показателей и сравнение их с базовыми значениями.

Методы определения показателей качества ПС различаются:

* по способам получения информации о ПС – измерительный, регистрационный, органолептический, расчетный;
* по источникам получения информации – экспертный, социологический.

*Измерительный метод* основан на получении информации о свойствах и характеристиках ПС с использованием инструментальных средств. Например, с использованием этого метода определяется объем ПС - число строк исходного текста программ и число строк - комментариев, число операторов и операндов, число исполненных операторов, число ветвей в программе, число точек входа (выхода), время выполнения ветви программы, время реакции и другие показатели.

*Регистрационный метод* основан на получении информации во время испытаний или функционирования ПС, когда регистрируются и подсчитываются определенные события, например, время и число сбоев и отказов, время передачи управления другим модулям, время начала и окончания работы.

*Органолептический метод* основан на использовании информации, получаемой в результате анализа восприятия органов чувств (зрения, слуха), и применяется для определения таких показателей как удобство применения, эффективность и т.п.

*Расчетный метод* основан на использовании теоретических и эмпирических зависимостей (на ранних этапах разработки), статистических данных, накапливаемых при испытаниях, эксплуатации и сопровождении ПС. При помощи расчетного метода определяются длительность и точность вычислений, время реакции, необходимые ресурсы.

Определение значений показателей качества ПС *экспертным методом* осуществляется группой экспертов-специалистов, компетентных в решении данной задачи, на базе их опыта и интуиции. Экспертный метод применяется в случаях, когда задача не может быть решена никаким другим из существующих способов или другие способы являются значительно более трудоемкими. Экспертный метод рекомендуется применять при определении показателей наглядности, полноты и доступности программной документации, легкости освоения, структурности.

*Социологические методы* основаны на обработке специальных анкет-вопросников.

**1. Факторы качества**

**2. Критерии качества**

 **3. Метрики**

**4. Оценочные**

**элементы**

Рис. 1 – Уровни системы показателей качества

Показатели качества объединены в систему из четырех уровней. Каждый вышестоящий уровень содержит в качестве составляющих показатели нижестоящих уровней (рисунок 1).

Стандарт **ИСО 9126 (ГОСТ Р ИСО/МЭК 9126) «Информационная технология. Оценка программной продукции. Характеристика качества и руководства по их применению».**

Определенные настоящим стандартом характеристики дополнены рядом требований по выбору метрик и их измерению для различных проектов ПС. Они применимы к любому типу ПС, включая компьютерные программы и данные, содержащиеся в программируемом оборудовании. Эти характеристики обеспечивают согласованную терминологию для анализа качества ПС. Кроме того, они определяют схему для выбора и специфицирования требований к качеству ПС, а также для сопоставления возможностей различных программных продуктов, таких как функциональные возможности, надежность, практичность и эффективность.

Все множество атрибутов качества ПС может быть классифицировано в структуру иерархического дерева характеристик и субхарактеристик. Самый высший уровень этой структуры состоит из характеристик качества, а самый нижний уровень – из их атрибутов. Эта иерархия не строгая, поскольку некоторые атрибуты могут быть связаны с более чем одной субхарактеристикой. Таким же образом, внешние свойства (такие, как пригодность, корректность, устойчивость к ошибкам или временная эффективность) влияют на наблюдаемое качество. Недостаток качества в использовании (например, пользователь не может закончить задачу) может быть прослежен к внешнему качеству (например, функциональная пригодность или простота использования) и связанным с ним внутренним атрибутам, которые необходимо изменить.

Внутренние метрики могут применяться в ходе проектирования и программирования к неисполняемым компонентам ПС (таким, как спецификация или исходный программный текст). При разработке ПС промежуточные продукты следует оценивать с использованием внутренних метрик, которые измеряют свойства программ, и могут быть выведены из моделируемого поведения. Основная цель внутренних метрик – обеспечивать, чтобы было достигнуто требуемое внешнее качество. Внутренние метрики дают возможность пользователям, испытателям и разработчикам оценивать качество ЖЦ программ и заниматься вопросами технологического обеспечения качества задолго до того, как ПС становится готовым исполняемым продуктом.

***Внутренние метрики*** позволяют измерять внутренние атрибуты или формировать признаки внешних атрибутов путем анализа статических свойств промежуточных или поставляемых программных компонентов. Измерения внутренних метрик используют категории, числа или характеристики элементов из состава ПС, которые, например, имеются в процедурах исходного программного текста, в графе потока управления, в потоке данных и в представлениях изменения состояний памяти. Документация также может оцениваться с использованием внутренних метрик.

***Внешние метрики*** используют меры ПС, выведенные из поведения системы, частью которых они являются, путем испытаний, эксплуатации или наблюдения исполняемого ПС или системы. Перед приобретением или использованием ПС его следует оценить с использованием метрик, основанных на деловых и профессиональных целях, связанных с использованием, эксплуатацией и управлением продуктом в определенной организационной и технической среде. Внешние метрики обеспечивают заказчикам, пользователям, испытателям и разработчикам возможность определять качество ПС в ходе испытаний или эксплуатации.

Метрики качества в использовании измеряют, в какой степени продукт удовлетворяет потребности конкретных пользователей в достижении заданных целей с результативностью, продуктивностью и удовлетворением в заданном контексте использования. При этом результативность подразумевает точность и полноту достижения определенных целей пользователями при применении ПС; продуктивность соответствует соотношению израсходованных ресурсов и результативности при эксплуатации ПС, а удовлетворенность – психологическое отношение к качеству использования продукта. Эта метрика не входит в число шести базовых характеристик ПС, регламентируемых стандартом ИСО 9126, однако рекомендуется для интегральной оценки результатов функционирования комплексов программ.

Оценивание качества в использовании должно подтверждать его для определенных сценариев и задач, оно составляет полный объединенный эффект характеристик качества ПС для пользователя. ***Качество в использовании*** – это восприятие пользователем качества системы, содержащей ПС, и оно измеряется скорее в терминах результатов использования комплекса программ, чем собственных внутренних свойств ПС. Связь качества в использовании с другими характеристиками качества ПС зависит от типа пользователя, так, например, для конечного пользователя качество в использовании обусловливают, в основном, характеристики функциональных возможностей, надежности, практичности и эффективности, а для персонала сопровождения ПС качество в использовании определяет сопровождаемость. На качество в использовании могут влиять любые характеристики качества, и это понятие шире, чем практичность, которая связана с простотой использования и привлекательностью. Качество в использовании, в той или иной степени, характеризуется сложностью применения комплекса программ, которую можно описать трудоемкостью использования с требуемой результативностью. Многие характеристики и субхарактеристики ПС обобщенно отражаются неявными технико-экономическими показателями, которые поддерживают функциональную пригодность конкретного ПС. Однако их измерение и оценка влияния на показатели качества, представляет сложную проблему.

**Ход работы**

**1. Изучить теоретический материал, законспектировать основные определения**

**2. Внести в отчет ответы на такие вопросы:**

### Как определяется понятие «качество» государственным и международным стандартами?

### Какие способы получения информации о ПС приведены в стандарте ГОСТ 28195-89?

### Какой стандарт необходимо применить для оценки качества ПС, если необходимо оценить уровень автоматизации ПС?

### Дайте определение фактора надёжность ПС и перечислите его критерии качества согласно ГОСТ 28195-89.

5) Чем обусловлена объективная необходимость повышения качества продукции и программных систем, в том числе, в современных условиях?

**3. Сделать вывод**

**Лабораторная работа № 2**

**Тема:**Требования к программной документации

**Цель**: Ознакомиться с документом ГОСТ 19.102-77 «Стадии разработки»; Ознакомиться с документом ГОСТ 19.105-78 «Общие требования к ПД».

**Оборудование:** Компьютер,MS Office.

ХОД РАБОТЫ

1. Изучите основные теоретические сведения (приложение А).
2. Определите стадии разработки программного обеспечения для своего проекта.
3. Опишите этапы, соответствующие выбранным стадиям.
4. Сделайте выводу по проделанной работе.

**ПРИЛОЖЕНИЕ А**

Разработку ПО осуществляют стадиями и этапами, содержание работ которых установлено в ГОСТ 19.102 (Стадии разработки). Стадии: техническое задание, эскизный проект, технический проект, рабочий проект, внедрение.

ГОСТ **19.105-78**

Настоящий стандарт устанавливает общие требования к оформлению программных документов для вычислительных машин, комплексов и систем, независимо от их назначения и области применения и предусмотренных стандартами Единой системы программной документации (ЕСПД) для любого способа выполнения документов на различных носителях данных.

Стандарт соответствует СТ СЭВ 2088-80 в части общих требований к оформлению информационной части (см. справочное приложение)

1. ОБЩИЕ ТРЕБОВАНИЯ
	1. Программный документ может быть представлен на различных типах носителей данных.

Программный документ состоит из следующих условных частей: 1.титульной; 2.информационной; 3.основной; 4.регистрации изменений.

* 1. Правила оформления документа и его частей на каждом носителе данных устанавливаются стандартами ЕСПД на правила оформления документов на соответствующих носителях данных.
1. ТИТУЛЬНАЯ ЧАСТЬ
	1. Титульная часть состоит из листа утверждения и титульного листа. Правила оформления листа утверждения и титульного листа устанавливаются по ГОСТ 19.104-78.
2. ИНФОРМАЦИОННАЯ ЧАСТЬ
	1. Информационная часть должна состоять из аннотации и содержания.
	2. Необходимость включения информационной части в различные виды программных документов установлена соответствующими стандартами ЕСПД на эти документы.
	3. В аннотации приводят сведения о назначении документа и краткое изложение его основной части.
	4. Содержание включает перечень записей о структурных элементах основной части документа, в каждую из которых входят:
* обозначение структурного элемента (номер раздела, подраздела и т.д.);
* наименование структурного элемента;
* адрес структурного элемента на носителе данных (например, номер страницы, номер файла и т.п.).

Правила обозначения структурных элементов основной части документа и их адресации устанавливаются стандартами ЕСПД на правила оформления документов на соответствующих носителях данных.

1. ОСНОВНАЯ ЧАСТЬ
	1. Состав и структура основной части программного документа устанавливаются стандартами ЕСПД на соответствующие документы.
2. ЧАСТЬ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ
	1. О каждом изменении программного документа в этой части делается запись в соответствии с требованиями ГОСТ 19.603-78.

**ГОСТ 19.102-77**

1. Настоящий стандарт устанавливает стадии разработки программ и программной документации для вычислительных машин, комплексов и систем независимо от их назначения и области применения.
2. Стадии разработки, этапы и содержание работ должны соответствовать указанным в таблице.

Таблица

|  |  |  |
| --- | --- | --- |
| **Стадии разработки** | **Этапы работ** | **Содержание работ** |
| 1. Техническое | Обоснование | Постановка задачи |
| задание | необходимости разработки | Сбор исходных материалов |
|  | программы | Выбор и обоснование критериев |
|  |  | эффективности и качества разрабатываемой |
|  |  | программы. |
|  |  | Обоснование необходимости проведения |
|  |  | научно-исследовательских работ. |
|  | Научно-исследовательские | Определение структуры входных и |
|  | работы | выходных данных. |
|  |  | Предварительный выбор методов решения |
|  |  | задач. |
|  |  | Обоснование целесообразности применения |
|  |  | ранее разработанных программ. |
|  |  | Определение требований к техническим |
|  |  | средствам. |
|  |  | Обоснование принципиальной возможности решения поставленной задачи |
| Разработка и утверждение технического задания | Определение требований к программе. Разработка технико-экономического обоснования разработки программы.Определение стадий, этапов и сроков разработки программы и документации на неё.Выбор языков программирования. Определение необходимости проведения научно-исследовательских работ на последующих стадиях.Согласование и утверждение технического задания. |
| 2. Эскизный проект | Разработка эскизного проекта | Предварительная разработка структуры входных и выходных данных.Уточнение методов решения задачи. Разработка общего описания алгоритма решения задачиРазработка технико-экономического обоснования. |
| Утверждение эскизного проекта | Разработка пояснительной записки. Согласование и утверждение эскизного проекта. |

|  |  |  |
| --- | --- | --- |
| 3.Технический проект | Разработка технического проекта | Уточнение структуры входных и выходных данных.Разработка алгоритма решения задачи. Определение формы представления входных и выходных данных.Определение семантики и синтаксиса языка. Разработка структуры программы.Окончательное определение конфигурации технических средств. |
| Утверждение технического проекта | Разработка плана мероприятий по разработке и внедрению программ. Разработка пояснительной записки. Согласование и утверждение технического проекта. |
| 4. Рабочий проект | Разработка программы | Программирование и отладка программы. |
| Разработка программной документации | Разработка программных документов в соответствии с требованиями ГОСТ 19.101- 77. |
| Испытания программы | Разработка, согласование и утверждение порядка и методики испытаний. Проведение предварительных государственных, межведомственных, приёмо-сдаточных и других видов испытаний Корректировка программы и программной документации по результатам испытаний. |
| 5. Внедрение | Подготовка и передача программы. | Подготовка и передача программы и программной документации для сопровождения и (или) изготовления. Оформление и утверждение акта о передаче программы на сопровождение и (или) изготовление.Передача программы в фонд алгоритмов и программ. |

Примечания**:**

* 1. Допускается исключать вторую стадию разработки, а в технически обоснованных случаях - вторую и третью стадии. Необходимость проведения этих стадий указывается в техническом задании.
* 2. Допускается объединять, исключать этапы работ и (или) их содержание, а также вводить другие этапы работ по согласованию с заказчиком.

### **Лабораторная работа № 3**

### **Тема:** Жизненный цикл программного средства.

**Цель работы:** ознакомление со стандартами в области обеспечения жизненного цикла программных средств.

**Оборудование:** Компьютер,MS Office.

 В основе деятельности по созданию и использованию программных средств лежит понятие жизненного цикла. Жизненный цикл является моделью создания и использования программного обеспечения, отражающей его различные состояния, начиная с момента возникновения необходимости в программном средстве и заканчивая моментом его полного выхода из употребления у пользователей.

Основными целями применения стандартов и нормативных документов в жизненном цикле ПС являются:

* снижение трудоемкости, длительности, стоимости и улучшение других технико-экономических показателей проектов ПС;
* повышение качества разрабатываемых и/или применяемых компонентов и ПС в целом при их приобретении, разработке, эксплуатации и сопровождении;
* обеспечение возможности расширять ПС по набору прикладных функций и масштабировать в зависимости от размерности решаемых задач;
* обеспечение переносимости прикладных программ и данных между разными аппаратно-программными платформами.

Применение стандартов позволяет ориентироваться на построение систем из крупных функциональных узлов, отвечающих требованиям стандартов, применять отработанные и проверенные проектные решения. Они определяют унифицированные интерфейсы и протоколы взаимодействия компонентов таким образом, что разработчику системы, как правило, не требуется вдаваться в детали внутреннего устройства этих компонентов.

 Жизненный цикл разработки ПС установлен стандартом **ГОСТ 19.102-77 «Стадии разработки программ и программной документации»** и содержит следующие этапы работ:

* техническое задание (ТЗ);
* эскизный проект (ЭЗ);
* технический проект (ТП);
* рабочий проект (РП);
* внедрение.

В таблице 3 приведены стадии разработки и этапы, их составляющие.

Таблица 3. Стадии и этапы разработки ПС

|  |  |
| --- | --- |
| Стадии разработки | Этапы работ |
| Техническое задание | Обоснование необходимости разработки программы |
| Научно-исследовательские работы |
| Разработка и утверждение технического задания |
| Эскизный проект | Разработка эскизного проекта |
| Утверждение эскизного проекта |
| Технический проект | Разработка технического проекта |
| Утверждение технического проекта |
| Рабочий проект | Разработка программы |
| Разработка программной документации |
| Испытания программы |
| Внедрение | Подготовка и передача программы |

Кроме рассмотренного выше жизненного цикла программ, существует жизненный цикл автоматизированных систем (АС) **ГОСТ 34.601–90 «Информационная технология. Автоматизированные системы. Стадии создания»**. Настоящий стандарт распространяется на автоматизированные системы, используемые в различных видах деятельности (исследование, проектирование, управление и т. п.), включая их сочетания, создаваемые в организациях, объединениях и на предприятиях. Стандарт устанавливает стадии и этапы создания АС, а также содержание работ на каждом этапе.

Процесс создания АС представляет собой совокупность упорядоченных во времени, взаимосвязанных, объединенных в стадии и этапы работ, выполнение которых необходимо и достаточно для создания АС, соответствующей заданным требованиям (табл. 4).

Допускается исключение стадии «Эскизный проект» и отдельных этапов работ на всех стадиях, объединение стадий «Технический проект» и «Рабочая документация» в одну стадию «Техно-рабочий проект».

Таблица 4. Стадии и этапы разработки АС

| Наименование этапа | Содержание этапа |
| --- | --- |
| 1. Формирование требований к АС | Обследование объекта и обоснование необходимости создания АС.Формирование требований пользователя АС.Оформление отчета о выполненной работе и заявки на разработку АС (тактико-технического задания) |
| 2. Разработка концепции АС | Изучение объекта.Проведение необходимых научно-исследовательских работ.Разработка вариантов концепции АС и выбор варианта концепции АС, удовлетворяющего требованиям пользователя.Оформление отчета о выполненной работе |
| 3. Техническое задание | Разработка и утверждение технического задания на создание АС |
| 4. Эскизный проект | Разработка предварительных проектных решений по системе в целом и ее частям.Разработка документации на АС и ее части |
| 5. Технический проект | Разработка проектных решений по системе и ее частям.Разработка документации на АС и ее части.Разработка и оформление документации на поставку изделий для комплектования АС и/или технических требований (технических заданий) на их разработку.Разработка заданий на проектирование в смежных частях проекта объекта автоматизации |
| 6. Рабочая документация | Разработка рабочей документации на систему и ее части.Разработка или адаптация программ |

В зависимости от специфики создаваемых АС и условий их создания допускается выполнение отдельных этапов работ до завершения предшествующих стадий, параллельное выполнение этапов работ, включение новых этапов работ.

Стандарт **ISO 12207 (ГОСТ Р ИСО/МЭК 12207) «Информационная технология. Процессы жизненного цикла программных средств»** наиболее полно на уровне международных стандартов отражает жизненный цикл, технологию разработки и обеспечения качества сложных программных средств. Жизненный цикл ПС представлен набором этапов, частных работ и операций в последовательности их выполнения и взаимосвязи, регламентирующих ведение разработки на всех стадиях от подготовки технического задания до завершения испытаний ряда версий и окончания эксплуатации ПС. В жизненный цикл включаются описания исходной информации, способов выполнения операций и работ, устанавливаются требования к результатам и правилам их контроля, а также к содержанию технологических и эксплуатационных документов. Определяется организационная структура коллективов, распределение и планирование работ, а также контроль за реализацией жизненного цикла ПС.

Стандарт может использоваться как непосредственный директивный, руководящий или рекомендательный документ, а также как организационная база при создании средств автоматизации соответствующих технологических этапов или процессов. Для реализации положений стандарта должны быть выбраны инструментальные средства, совместно образующие взаимосвязанный комплекс технологической поддержки и автоматизации ЖЦ и не противоречащие предварительно скомпонованному набору нормативных документов. Имеющиеся в стандарте пробелы следует заполнять спецификациями или нормативными документами, регламентирующими применение выбранных или созданных инструментальных средств автоматизации разработки и документирования ПС.

Вопросы:

### Опишите особенности каскадной модели жизненного цикла ПС.

### Опишите особенности спиральной модели жизненного цикла ПС.

### Какая модель жизненного цикла ПС характерна для периода 1970-1985 гг.?

### В чем состоит отличие спиральной модели ЖЦ ПС от каскадной?

### Перечислите этапы работ согласно ГОСТ 19.102-77 «Стадии разработки программ и программной документации».

**Лабораторная работа №4**

**Тема:** Изучение основных факторов, определяющих качество программных средств

**Цель:** изучить основные факторы определяющие качество ПО.

**Оборудование:** Компьютер, MS Office

Ход работы

Изучить следующие основные показатели качества:

### 1. Стандарты качества

### 2. Показатели качества

### 3. Цена качества

### 4. регрессионный анализ качества

### 5. Факторы и их оценка

### Каждую их характеристик качества внести в таблицу, пример которой приведен ниже.

|  |  |  |
| --- | --- | --- |
| №п/п | Показатели качества ПО | Описание показателей качества ПО |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |
| 5 |  |  |

### Сделать вывод.

**Лабораторная работа №5**

**Тема:** Метрики характеристик качества программных средств

**Цель:** изучить основные характеристик качества программных средств.

**Оборудование:** Компьютер, MS Office.

Общее представление о качестве ПС международным стандартом **ISO 9126:1-4:2002**рекомендуется описывать тремя взаимодействующими и взаимозависимыми ***метриками характеристик качества,***отражающими:

* внутреннее качество, проявляющееся в процессе разработки и других промежуточных этапов жизненного цикла ПС;
* внешнее качество, заданное требованиями заказчика в спецификациях и отражающееся характеристиками конечного продукта;
* качество при использовании в процессе нормальной эксплуатации и результативностью достижения потребностей пользователей с учетом затрат ресурсов.

***Внутренние метрики***в соответствии со стандартами могут применяться в ходе проектирования и программирования к компонентам ПС, таким, как спецификация или исходный программный текст. При разработке ПС промежуточные компоненты следует оценивать с использованием внутренних метрик, которые отражают функциональные и конструктивные свойства программ. Основная цель применения внутренних метрик — обеспечивать, чтобы разработчиками было получено требуемое внешнее качество. Рекомендуется использовать внутренние метрики, которые имеют наиболее сильные связи с приоритетными внешними метриками, чтобы они могли помогать при прогнозировании их достижимых значений.

### Внешние метрики используют меры ПС, отражающие поведение системы, частью которой они являются, путем испытаний, эксплуатации и наблюдения исполняемых программ или функционирования системы. Перед приобретением или использованием ПС его следует оценить с использованием метрик, основанных на реализации деловых и профессиональных целей, связанных с применением программного продукта в определенной организационной и технической среде. Внешние метрики обеспечивают заказчикам, пользователям и разработчикам возможность прослеживать и анализировать качество ПС в ходе испытаний или опытной эксплуатации. Подходящие внешние метрики специфицируются для получения числовых значений или категорий и свойств внутренних характеристик качества, чтобы их можно было использовать для проверки того, что промежуточные продукты в процессе разработки удовлетворяют внутренним спецификациям качества.

### Метрики качества в использовании отражают, в какой степени продукт удовлетворяет потребности конкретных пользователей в достижении заданных целей. Эта метрика не отражена в числе шести базовых характеристик ПС, регламентируемых стандартом ISO 9126-1 вследствие ее общности, однако рекомендуется для интегральной оценки результатов функционирования и применения комплексов программ в стандарте ISO 9126-4.

Вопросы:

### 1. Внутреннее качество.

### 2. Внешнее качество.

3. Качество при использовании в процессе нормальной эксплуатации.

**Лабораторная работа № 6**

**Тема:** Ресурсы, ограничивающие достижимые характеристики качества.

**Цель:** изучить, ограничивающие достижимые характеристики качества.

**Оборудование:** Компьютер, MS Office.

Конкретное ПС может быть частью большей системы, состоящей из других программных продуктов, с которыми оно имеет интерфейсы аппаратных средств, персонала операторов и рабочих потоков данных. Поставляемое ПС оценивается уровнями отобранных для спецификаций внешних метрик. Эти метрики описывают его взаимодействие со средой и оцениваются путем наблюдения и измерения ПС в действии.

Внутренние метрики в соответствии со стандартами могут применяться в ходе проектирования и программирования к неисполняемым компонентам ПС таким, как спецификация или исходный программный текст. При разработке ПС промежуточные продукты следует оценивать с использованием внутренних метрик, которые отражают некоторые функциональные и конструктивные свойства программ. Внутренние метрики дают возможность разработчикам, испытателям и заказчикам оценивать качество ЖЦ программ и заниматься вопросами технологического обеспечения качества до того, как ПС становится готовым к использованию продуктом. Измерения внутренних метрик используют свойства, категории, числа или характеристики элементов из состава ПС, которые, например, имеются в процедурах исходного программного текста, в графе потока управления, в потоке данных и в представлениях изменения состояний памяти. Качество документации также может оцениваться с использованием внутренних метрик.

Внешние метрики используют меры ПС, выведенные из поведения системы, частью которых они являются, путем испытаний, эксплуатации и наблюдения исполняемых программ или функционирования информационной системы. Перед приобретением или использованием ПС его следует оценить с использованием метрик, основанных на деловых и профессиональных целях, связанных с применением и управлением программным продуктом. Внешние метрики обеспечивают заказчикам, пользователям и разработчикам возможность прослеживать качество ПС в ходе испытаний или опытной эксплуатации.

Когда общие требования к функциям ПС определены, в них должны быть выделены и перечислены характеристики и субхарактеристики, которые составляют полный набор показателей качества конкретного комплекса программ. Затем определяются подходящие внешние метрики, их меры и приемлемые диапазоны значений, устанавливающие количественные и качественные критерии, которые подтверждают, что ПС удовлетворяет потребностям заказчика и пользователей. Далее определяются и специфицируются внутренние атрибуты качества, чтобы спланировать удовлетворение внешних характеристик качества в конечном продукте и обеспечить их в промежуточных продуктах в ходе разработки. Подходящие внутренние метрики специфицируются для получения числовых значений или категорий и свойств внутренних характеристик качества, чтобы их можно было использовать для проверки того, что промежуточные продукты в процессе разработки удовлетворяют внутренним спецификациям качества.

Вопросы:

1. Назовите и охарактеризуйте основные факторы, влияющие на качество программных средств.

2. Опишите модель характеристик качества программных средств согласно стандарту ISO 9126.

3. Опишите модель процесса конкретизации метрик качества в жизненном цикле программных средств согласно стандарту ISO 9126.

4. Какие группы показателей характеристик качества программных средств вы знаете? Какие они имеют особенности?

5. Какие негативные факторы влияют на качество программных средств?

6. Какие ресурсы ограничивают достижение заданных характеристик качества программных средств?

**Лабораторная работа № 7**

**Тема:** Процесс выбора характеристик в проектах программного обеспечения.

**Цель:** изучить, выбора характеристик в проектах ПО.

**Оборудование:** Компьютер, MS Office.

Конструктивным характеристикам качества, которые не отражаются на адекватном улучшении функций ПС. Поэтому для каждого проекта необходимо ранжировать характеристики и их атрибуты и выделять, прежде всего, те, которые могут в наибольшей степени улучшить функциональную пригодность для конкретных целей. Таким образом, при системном анализе, формировании технического задания и спецификаций требований возникает **два класса оптимизационных задач:**

— распределение затрат на улучшение отдельных, конструктивных характеристик ПС с целью достижения его максимальной или достаточно высокой функциональной пригодности;

— определение оптимальных или допустимых затрат на улучшение каждой конструктивной характеристики ПС, обеспечивающих адекватное или достаточно существенное увеличение качества функционирования.

Решение этих задач должно быть направлено на обеспечение достаточно высокой функциональной пригодности ПС **путем сбалансированного улучшения остальных характеристик качества в условиях ограниченных ресурсов на**ЖЦ. Для этого в процессе системного анализа при подготовке технического задания и требований спецификаций, значения, требуемых атрибутов и субхарактеристик качества должны проверяться по степени их влияния на функциональную пригодность. Излишне высокие требования к отдельным атрибутам качества, требующие для реализации больших дополнительных трудовых и вычислительных ресурсов, целесообразно снижать, если они слабо влияют на основные, функциональные характеристики ПС. Таким образом, ограниченные ресурсы трудоемкости и длительности этапов ЖЦ ПС должны распределяться по процессам улучшения отдельных характеристик и атрибутов качества с учетом их воздействия на повышение функциональной пригодности.

Строгое формализованное решение этих задач в большинстве случаев невозможно, однако качественный системный анализ может помогать выявлению основных тенденций изменения и взаимосвязей значений характеристик. Наиболее просто могут быть установлены рациональные значения стандартизированных характеристик или их номинальные категории свойств для определенных классов ПС. При определении этих границ следует учитывать корреляцию как между атрибутами определенных характеристик, так и между различными характеристиками. Так, например, надежность функционирования ПС при больших нагрузках и перегрузках.

**Ход работы**

Сделать вывод

**Лабораторная работа № 8**

**Тема:** Выбор и установление характеристик в проектах программного обеспечения.

**Цель:** изучить, характеристики ПО.

**Оборудование:** Компьютер, MS Office.

Описание в стандарте **ISO 9126:1-4** характеристик качества программных средств не содержит рекомендаций и методик выбора их значений в требованиях к конкретным проектам. Необходимо, прежде всего, установить диапазоны рациональные мер и шкал для каждой характеристики и ее атрибутов, которые можно будет использовать в качестве первичных ограничений при выборе их значений для реальных проектов. Далее должны быть разработаны процессы выбора, установления и представления в спецификациях, требований к атрибутам каждой характеристики качества. Эти требования должны учитывать реальные ограничения ресурсов, доступных для обеспечения ЖЦ ПС. Ресурсы этих процессов и атрибуты характеристик качества ниже, по возможности, сводятся к трудоемкости и длительности их реализации, а также к соответствующему влиянию этих параметров в функциональную пригодность.

Улучшение каждой характеристики качества требует некоторых затрат (трудоемкости, финансов, времени), которые в той или иной степени отражаются на основной характеристике качества – на ***функциональной пригодности***. При выборе конкретных мер и шкал конструктивных характеристик качества следует учитывать возможные затраты на их достижение и результирующее повышение функциональной пригодности, желательно, в сопоставимых экономических единицах, в тех же мерах и масштабах. Такое, даже качественное ***сравнение эффекта и затрат***позволяет избежать многих не рентабельных повышений требований к отдельным конструктивным характеристикам качества, которые не отражаются на адекватном улучшении функций ПС. Поэтому для каждого проекта необходимо ранжировать характеристики и их атрибуты и выделять, прежде всего, те, которые могут в наибольшей степени улучшить функциональную пригодность для конкретных целей. Таким образом, при системном анализе, формировании технического задания и спецификаций требований возникает ***два класса оптимизационных задач***:

-  распределение затрат на улучшение отдельных, конструктивных характеристик ПС с целью достижения его максимальной или достаточно высокой, функциональной пригодности;

-  определение оптимальных или допустимых затрат на улучшение каждой конструктивной характеристики ПС, обеспечивающих адекватное или достаточно существенное увеличение качества функционирования.

Решение этих задач должно быть направлено на обеспечение достаточно высокой функциональной пригодности ПС ***путем сбалансированного улучшения остальных характеристик качества в условиях ограниченных ресурсов на ЖЦ***. Для этого в процессе системного анализа при подготовке технического задания и требований спецификаций, значения, требуемых атрибутов и субхарактеристик качества, должны проверяться по степени их влияния на функциональную пригодность. Излишне высокие требования к отдельным атрибутам качества, требующие для реализации больших дополнительных трудовых и вычислительных ресурсов, целесообразно снижать, если они слабо влияют на основные, функциональные характеристики ПС. Таким образом, ограниченные ресурсы трудоемкости и длительности этапов ЖЦ ПС должны распределяться по процессам улучшения отдельных характеристик и атрибутов качества с учетом их воздействия на повышение функциональной пригодности.

Строгое формализованное решение этих задач в большинстве случаев невозможно, однако качественный системный анализ может помогать выявлению основных тенденций изменения и взаимосвязей значений характеристик. Наиболее просто могут быть установлены рациональные значения стандартизированных характеристик или их номинальные категории свойств для определенных классов ПС. При определении этих границ следует учитывать корреляцию, как между атрибутами определенных характеристик, так и между различными характеристиками. Так, например, надежность функционирования ПС при больших нагрузках и перегрузках может сильно зависеть от временной эффективности использования производительности ЭВМ. Используемость ресурсов ЭВМ может ограничивать сопровождаемость и изменяемость программ, и то и другое необходимо учитывать при определении требований к характеристикам конкретных проектов ПС.

**Ход работы**

Сделать вывод

**Лабораторная работа № 9**

**Тема:** Выбор мер качества в проектах программного обеспечения

**Цель:** изучить, качество ПО.

**Оборудование:** Компьютер, MS Office.

Согласие, достигнутое по требованиями к качеству (в оригинале — quality requirements), наравне с четким доведением до инженеров того, что составляет качество <получаемого продукта>, требуют обсуждения и формального определения многих аспектов качества.

Инженеры должны понимать смысл, вкладываемый в концепцию качества, характеристики и значение качества в отношении разрабатываемого или сопровождаемого программного обеспечения.

Важной идеей является то, что программные требования определяют требуемые характеристики качества программного обеспечения, а также влияют на методы количественной оценки и сформулированные для оценки этих характеристик <соответствующие> критерии приемки.

Движущей силой программных проектов является желание создать программное обеспечение, обладающее определенной ценностью. Ценность программного обеспечения в может выражаться в форме стоимости, а может и нет. Заказчик, обычно, имеет свое представление о максимальных стоимостных вложениях, возврат которых ожидается в случае достижения основных целей создания программного обеспечения. Заказчик может, также, иметь определенные ожидания в отношении качества ПО. Иногда, заказчики не задумываются о вопросах качества и связанной с ними стоимостью. Является ли характеристики качества чисто декоративными или, все же, это неотъемлемая часть программного обеспечения? Ответ, вероятно, находится где-то посередине, как почти всегда бывает в таких случаях, и является предметом обсуждения степени вовлечения заказчика в процесс принятия решений и полного понимания заказчиком стоимости и выгоды, связанной с достижением того или иного уровня качества. В идеальном случае, большинство такого рода решений должно приниматься процессе работы с требованиями, однако эти вопросы могут подниматься на протяжении всего жизненного цикла программного обеспечения. Не существует каких-то <“стандартных”> правил того, как именно необходимо принимать такие решения. Однако, инженеры должны быть способны представить различные альтернативы и их стоимость.

Вопросы

1. Модели и характеристики качества

2.Качество процессов программного обеспечения

3. Качество программного продукта

**Лабораторная работа № 10**

**Тема:** Оценивание уровня зрелости процессов жизненного цикла и обеспечение качества программных средств.

**Цель:** изучить, уровни зрелости процессов жизненного цикла.

**Оборудование:** Компьютер, MS Office.

Серия стандартов ISO 9000. В серии ISO 9000 сформулированы необходимые условия для достижения некоторого минимального уровня организации процесса, но не дается никаких рекомендаций по дальнейшему совершенствованию процессов.

СММ. СММ представляет собой совокупность критериев оценки зрелости организации-разработчика и рецептов улучшения существующих процессов.

Примечание. Изначально СММ разрабатывалась и развивалась как методика, позволяющая крупным правительственным организациям США выбирать наилучших поставщиков программного обеспечения. Для этого предполагалось создать исчерпывающее описание способов оценки процессов разработки программного обеспечения и методики их дальнейшего усовершенствования. В итоге авторы смогли добиться такой степени подробности и детализации, что стандарт оказался пригодным и для обычных компаний-разработчиков, желающих качественно улучшить существующие процессы разработки, привести их к определенным стандартам.

СММ определяет пять уровней зрелости организаций-разработчиков, причем каждый следующий уровень включает в себя все ключевые характеристики предыдущих.

Начальный уровень (initial level) - описан в стандарте в качестве осно вы для сравнения со следующими уровнями. На предприятии такого уровня организации не существует стабильных условий для создания качественного программного обеспечения. Результат любого проекта целиком и полностью зависит от личных качеств менеджера и опыта программистов, причем успех в одном проекте может быть повторен только в случае назначения тех же ме неджеров и программистов на следующий проект. Более того, если эти мене джеры или программисты уходят с предприятия, то резко снижается качест во производимых программных продуктов. В стрессовых ситуациях процесс разработки сводится к написанию кода и его минимальному тестированию.

Повторяемый уровень (repeatable level) - на предприятии внедрены технологии управления проектами. При этом планирование и управление проектами основывается на накопленном опыте, существуют стандарты на разрабатываемое программное обеспечение (причем обеспечивается следо вание этим стандартам) и специальная группа обеспечения качества. В слу чае необходимости организация может взаимодействовать с субподрядчика ми. В критических условиях процесс имеет тенденцию скатываться на на чальный уровень.

Определенный уровень (defined level) - характеризуется тем, что стан дартный процесс создания и сопровождения программного обеспечения пол ностью документирован (включая и разработку ПО, и управление проекта ми). Подразумевается, что в процессе стандартизации происходит переход на наиболее эффективные практики и технологии. Для создания и поддержания подобного стандарта в организации должна быть создана специальная груп па. Наконец, обязательным условием для достижения данного уровня явля ется наличие на предприятии программы постоянного повышения квалифи кации и обучения сотрудников. Начиная с этого уровня, организация переста ет зависеть от качеств конкретных разработчиков, и процесс не имеет тен денции скатываться на уровень ниже в стрессовых ситуациях.

Управляемый уровень (managed level) - в организации устанавливают ся количественные показатели качества как на программные продукты, так и на процесс в целом. Таким образом, более совершенное управление проек тами достигается за счет уменьшения отклонений различных показателей проекта. При этом осмысленные вариации в производительности процесса можно отличить от случайных вариаций (шума), особенно в хорошо освоен ных областях.

5. Оптимизирующий уровень (optimizing level) - характеризуется тем, что мероприятия по улучшению применяются не только к существующим процессам, но и для оценки эффективности ввода новых технологий. Основной задачей всей организации на этом уровне является постоянное улучшение существующих процессов. При этом улучшение процессов в идеале должно помогать предупреждать возможные ошибки или дефекты. Кроме того, должны вестись работы по уменьшению стоимости разработки программного обеспечения, например с помощью создания и повторного использования компонентов.

Сертификационная оценка соответствия всех ключевых областей проводится по 10-балльной шкале. Для успешной квалификации данной ключевой области необходимо набрать не менее 6 баллов. Оценка ключевой области осуществляется по следующим показателям:

заинтересованность руководства в данной области, например, планируется ли практическое внедрение данной ключевой области, существует ли понимание у руководства необходимости данной области и т. д.;

насколько широко данная область применяется в организации, напри мер, оценке в 4 балла соответствует фрагментарное применение;

успешность использования данной области на практике, например, оценке в 0 баллов соответствует полное отсутствие какого-либо эффекта, а оценка в 8 баллов выставляется при наличии систематического и измеримо го положительного результата практически во всей организации.

В принципе, можно сертифицировать только один процесс или подразделение организации, например, подразделение разработки программного обеспечения компании IBM сертифицировано на пятый уровень. Кстати, в мире существует совсем немного компаний, которые могут похвастаться наличием у них пятого уровня СММ хотя бы в одном из подразделений - таких всего около 50-ти. С другой стороны, насчитывается несколько тысяч компаний, сертифицированных по третьему или четвертому уровням, т. е. существует колоссальный разрыв между оптимизированным уровнем зрелости и предыдущими уровнями. Однако еще больший разрыв наблюдается между количеством организаций начального уровня и числом их более продвинутых собратьев - по некоторым оценкам, свыше 70 % всех компаний-разработчиков находится на первом уровне СММ.

**Вопросы:**

 1. Что понимают под термином «технология программирования»? 2. Что называют подходом и чем подход отличается от метода? 3. Назовите основные периоды истории развития технологии программирования. Чем характеризуются эти периоды? Как изменялись основные подходы и используемые средства?

4. Дайте определение понятию «сложная иерархическая система». Какой под используют при разработке таких систем?

5. Что понимают под термином «жизненный цикл программного обеспечения»? Какие основные процессы включают в это понятие?

6. Назовите основные этапы разработки программного обеспечения. Какие основные задачи решаются на этих этапах?

7. Назовите основные модели жизненного цикла программного обеспечения. С чем связано появление новых моделей?

8. Какие технологии называют CASE-технологиями? Почему?

**Лабораторная работа № 11**

**Тема:** Оценивание готового программного обеспечения по стандарту ИСО 14598.

**Цель:** изучить, оценивание готового программного обеспечения.

**Оборудование:** Компьютер, MS Office.

Характеристика как внутреннего, так и внешнего качества «защищённость» используется в стандартах в значении: способность ПП защищать информацию и данные так, чтобы неавторизованные субъекты или процессы не смогли читать или модифицировать их, а авторизованным пользователям и процессам не было отказано в доступе к ним. В стандартах подчеркивается, что данное требование также относится и к данным, которые находятся в процессе пересылки.

Характеристика качества «безопасность» вводится как характеристика качества в использовании, данная характеристика определяет способность ПП достигать приемлемого уровня риска для здоровья людей, их бизнеса, ПО, имущества или окружающей среды при данном способе (контексте) применения.

Если учесть, что информационная безопасность включает в себя вопросы обеспечения целостности, конфиденциальности и доступности, то можно прийти к выводу, что при комплексной оценке безопасности ПП и его использования в соответствии с вышеназванными стандартами нельзя ограничиться только этими характеристиками. Необходимо также учесть (полностью или частично) такие характеристики как «надёжность», «сопровождаемость», а также ряд смежных субхарактеристик (в составе других характеристик), которые косвенно затрагивают атрибуты безопасности ПП и системы, в которой он используется.

Допустим, мы определились с теми характеристиками и субхарактеристиками, которые мы собираемся оценивать для разрабатываемого ПП. Следующий этап, это собственно «язык чисел». Стандарт определяет набор метрик, при помощи которых можно численно оценить каждую субхарактеристику.

Для иллюстрации подхода ниже будут приведены метрики субхарактеристик внешнего и внутреннего качества «защищённость» и качества в использовании «безопасность». В стандарте рекомендуется разрабатывать собственные метрики, в которых будет более точно учтены проблемы безопасности, затрагивающие ПС.

Таблица – Метрики субхарактеристик внешнего и внутреннего качества «защищённость» и качества в использовании «безопасность»:

|  |  |  |
| --- | --- | --- |
| **Метрика** | **Формула** | **Примечания из стандарта** |
| 1. Внешние метрики безопасности:1.1 протоколирование доступа; | Х = А / В;А = число «фактов доступа пользователя к системе и данным», зафиксированных в протоколе системы;В = число «фактов доступа пользователя к системе и данным», которые были произведены во время оценки; | 1. рекомендуется использовать «тесты на проникновения» для эмуляции атак на систему;2. под записью протокола «факт доступа пользователя к системе и данным» может подразумеваться запись «факт обнаружения вируса» для обеспечения антивирусной безопасности системы;3. метрика носит экспериментальный характер; |
| 1.2 контролируемость доступа; | Х = А / В;А = число обнаруженных видов несанкционированного доступа;В = число видов несанкционированного доступа в спецификации; | 1. необходимо проверить способность системы определять факты несанкционированного доступа при неправильном применении функций системы;2. рекомендуется использовать «тесты на проникновения» для эмуляции атак на систему;3. метрика носит экспериментальный характер; |
| 1.3 предотвращение повреждения данных; | а) Х = 1 – А / N;A = число фактов существенного повреждения данных;N = число видов тестов, при помощи которых пытались спровоцировать факт повреждения данных;b) Y = 1 – B / N;B = число фактов незначительного повреждения данных;c) Х = A / T или B / T;T = время выполнения операции; | 1. необходимо проверить корректность работы системы при неправильном применении её функций;2. необходимо построить классификацию эффекта от событий повреждения данных;3. для вычисления внешних метрик следует использовать информацию, доступную извне системы. Порядок подсчёта событий здесь отличается от порядка подсчёта событий при оценке аналогичных внутренних данных;4. рекомендуется использовать «тесты на проникновения» для эмуляции атак на систему;5. метрика носит экспериментальный характер;6. Резервирование данных – один из наиболее эффективных способов предотвращения фактов повреждения данных, однако, резервирование данных относится к метрике «надёжность». |
| 2. внутренние метрики безопасности:2.1 протоколирование доступа; | Х = А / В;А = число типов доступа, которые были зарегистрированы корректно, как определено в спецификации;В = число типов доступа, которые должны регистрироваться по спецификации; |   |
| 2.2 контроль доступа; | Х = А / В;А = число требований контроля доступа, реализованных корректно, в соответствии со спецификацией;В = число требований контроля доступа в спецификации; |   |
| 2.4 предотвращение повреждения данных; | Х = А / В;А = число реализованных механизмов защиты от повреждения данных;В = число механизмов, требуемых по спецификации; | необходимо учитывать уровни безопасности при использовании этой метрики; |
| 2.5 криптографическая защита данных; | Х = А / В;А = число реализованных механизмов;В = число требуемых механизмов по спецификации; | криптографическая защита данных может касаться, например, данных в открытых базах данных или общедоступных данных. |
| 3. метрики безопасности качества в использовании:3.1 безопасность пользователей и их здоровья; | Х = 1 – А / В;А = число пользователей, сообщивших о наличии проблем;В = число пользователей; | Проблемы со здоровьем могут включать: травмы от многократно повторяющихся мышечных напряжений, утомление, головная боль и т.д.; |
| 3.2 безопасность людей, задействованных в использовании системы; | Х = 1 – А / В;А = число людей, подверженных риску;В = число людей, задействованных в использовании продукта; |   |
| 3.3 экономический ущерб; | Х = 1 – А / В;А = число событий экономического ущерба;В = общее число использования системы; | также можно учитывать ситуации, где был риск экономического ущерба; |
| 3.4 повреждение прочего ПО; | Х = 1 – А / В;А = число событий повреждения прочего ПО;В = общее число использования системы; | также можно учитывать ситуации, где был риск повреждения прочего ПО;метрика также может быть вычислена как Х = суммарная стоимость повреждённого ПО / время использования. |

Вопросы:

Составить оценивание готового программного обеспечения по стандарту.

**Лабораторная работа № 12**

**Тема:** Правила выполнения и оформления блок-схем. Правила и стандарты оформления дипломной работы.

**Цель:** изучить, правила выполнения и оформления блок-схем.

**Оборудование:** Компьютер, MS Office.

Блок-схема представляет собой совокупность символов, соответствующих этапам работы алгоритма и соединяющих их линий. Пунктирная линия используется для соединения символа с комментарием. Сплошная линия отражает зависимости по управлению между символами и может снабжаться стрелкой. Стрелку можно не указывать при направлении дуги слева направо и сверху вниз. Согласно п. 4.2.4, линии должны подходить к символу слева, либо сверху, а исходить снизу, либо справа.

Есть и другие типы линий, используемые, например, для изображения блок-схем параллельных алгоритмов, но в текущей статье они, как и ряд специфических символов, не рассматриваются. Рассмотрены лишь основные символы, которых всегда достаточно студентам.

|  |  |
| --- | --- |
| flowcharts_terminatorТерминатор начала и конца работы функции | Терминатором начинается и заканчивается любая функция. Тип возвращаемого значения и аргументов функции обычно указывается в комментариях к блоку терминатора. |
| flowcharts_dataОперации ввода и вывода данных | В ГОСТ определено множество символов ввода/вывода, например вывод на магнитные ленты, дисплеи и т.п. Если источник данных не принципиален, обычно используется символ параллелограмма. Подробности ввода/вывода могут быть указаны в комментариях. |
| flowcharts_processВыполнение операций над данными | В блоке операций обычно размещают одно или несколько (ГОСТ не запрещает) операций присваивания, не требующих вызова внешних функций. |
| flowcharts_solutionБлок, иллюстрирующий ветвление алгоритма | Блок в виде ромба имеет один вход и несколько подписанных выходов. В случае, если блок имеет 2 выхода (соответствует оператору ветвления), на них подписывается результат сравнения — «да/нет». Если из блока выходит большее число линий (оператор выбора), внутри него записывается имя переменной, а на выходящих дугах — значения этой переменной. |
| flowcharts_procedureВызов внешней процедуры | Вызов внешних процедур и функций помещается в прямоугольник с дополнительными вертикальными линиями. |
| flowcharts_loopНачало и конец цикла | Символы начала и конца цикла содержат имя и условие. Условие может отсутствовать в одном из символов пары. Расположение условия, определяет тип оператора, соответствующего символам на языке высокого уровня — оператор с предусловием (while) или постусловием (do … while). |
| flowcharts_preprocessПодготовка данных | Символ «подготовка данных» в произвольной форме (в ГОСТ нет ни пояснений, ни примеров), задает входные значения. Используется обычно для задания циклов со счетчиком. |
| flowcharts_connectorСоединитель | В случае, если блок-схема не умещается на лист, используется символ соединителя, отражающий переход потока управления между листами. Символ может использоваться и на одном листе, если по каким-либо причинам тянуть линию не удобно. |
| flowcharts_commentКомментарий | Комментарий может быть соединен как с одним блоком, так и группой. Группа блоков выделяется на схеме пунктирной линией |

Вопросы:

Составить блок-схему согласно правилам оформления.

**Лабораторная работа № 13**

**Тема:** Требования к программной документации.

**Цель:** изучить, требования к программной документации.

**Оборудование:** Компьютер, MS Office.

Предварительный состав программной документации

Предварительный состав программной документации должен включать в себя:

1. техническое задание;
2. текст программы;
3. описание программы;
4. программу и методики испытаний;
5. пояснительную записку;
6. описание применения;

7) руководство пользователя;

5.Условия эксплуатации

Требования к организации и оборудованию рабочих мест с ПЭВМ регламентируются СанПиН 2.2.2/2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы» и Р\_2.2.2006-05 «Гигиенические критерии оценки и классификация условий труда по показателям вредности и опасности факторов производственной среды».

6. Стадии и этапы разработки

6.1. Стадии разработки

Разработка должна быть произведена в три стадии:

1. Разработка технического задания;
2. Рабочее проектирование;
3. Внедрение;

6.2. Этапы разработки

На стадии рабочего проектирования должны быть выполнены перечисленные ниже этапы работ:

1. разработка АИСТ;
2. разработка программной документации;
3. испытания АИСТ.

На стадии внедрения должен быть выполнен этап разработки - подготовка АИСТ.

6.3. Содержание работ по этапам

На этапе разработки АИСТ должна быть выполнена работа по программированию (кодированию) и отладке программного обеспечения (АИСТ).

На этапе разработки программной документации должна быть выполнена разработка программных документов в соответствии с требованием п. 4.1. настоящего технического задания.

На этапе испытаний АИСТ должны быть выполнены перечисленные ниже виды работ:

1. проверка выполнения заданных функций АИСТ;
2. выявления и устранения недостатков в АИСТ и программной документации;
3. корректировка АИСТ и программной документации по результатам тестирований.

На этапе подготовки АИСТ должна быть выполнена работа по подготовке программного средства и программной документации для эксплуатации.

Вопросы:

1. Составить техническую документацию согласно требованиям.

 **Лабораторная работа № 14**

**Тема:** Применение нормативных документов к основным видам продукции (услуг) и процессов.

**Цель:** изучить, требования к программной документации.

**Оборудование:** Компьютер, MS Office.

**Стандарт**— это нормативный документ, разработанный на основе консенсуса, утвержденный признанным органом, направленный на достижение оптимальной степени упорядочения в определенной области. В стандарте устанавливаются для всеобщего и многократного использования общие принципы, правила, характеристики, касающиеся различных видов деятельности или их результатов. Стандарт должен быть основан на обобщенных результатах научных исследований, технических достижений и практического опыта, тогда его использование принесет оптимальную выгоду для общества.

*Предварительный стандарт*— это временный документ, который принимается органом по стандартизации и доводится до широкого круга потенциальных потребителей, а также тех, кто может его применить. Информация, полученная в процессе использования предварительного стандарта, и отзывы об этом документе служат базой для решения вопроса о целесообразности принятия стандарта.

Стандарты бывают международными, региональными, национальными, административно-территориальными. Они принимаются соответственно международными, региональными, национальными, территориальными органами по стандартизации. Все эти категории стандартов предназначены для широкого круга потребителей. По существующим нормам стандартизации стандарты периодически пересматриваются для внесения изменений, чтобы их требования соответствовали уровню научно-технического прогресса, или, согласно терминологии [ИСО](https://www.xumuk.ru/bse/1110.html)/МЭК, стандарты должны представлять собой "признанные технические правила". Нормативный документ, в том числе и стандарт, считается признанным техническим правилом, если он разработан в сотрудничестве с заинтересованными сторонами путем консультаций и на основе консенсуса.

Указанные выше категории стандартов называют общедоступными. Другие же категории стандартов, такие, как фирменные или отраслевые, не являясь таковыми, могут, однако, использоваться и в нескольких странах согласно существующим там правовым нормам.

В учебнике стандарт рассматривается как одна из разновидностей нормативных документов. Однако в практике термин "стандарт" может употребляться и по отношению к эталону, образцу или описанию продукта, процесса (услуги). По существу это не является принципиальной ошибкой, хотя эталон правильнее относить к области метрологии, а термин "стандарт" использовать применительно к нормативному документу.

**Документ технических условий**(technical specification)2 устанавливает технические требования к продукции, услуге, процессу. Обычно в документе технических условий должны быть указаны методы или процедуры, которые следует использовать для проверки соблюдения требований данного нормативного документа в таких ситуациях, когда это необходимо.

**Свод правил,**как и предыдущий нормативный документ, может быть самостоятельным стандартом либо самостоятельным документом, а также частью стандарта. Свод правил обычно разрабатывается для процессов [проектирования](https://www.xumuk.ru/encyklopedia/2/3686.html), монтажа оборудования и конструкций, технического обслуживания или эксплуатации объектов, конструкций, изделий. Технические правила, содержащиеся в документе, носят рекомендательный характер.

Все вышеуказанные нормативные документы являются рекомендательными. В отличие от них обязательный характер носит регламент. **Регламент**— это документ, в котором содержатся обязательные правовые нормы. Принимает регламент орган власти, а не орган по стандартизации, как в случае других нормативных документов. Разновидность регламентов — *технический регламент —*содержит технические требования к объекту стандартизации. Они могут быть представлены непосредственно в самом этом документе либо путем ссылки на другой нормативный документ (стандарт, документ технических условий, свод правил). В отдельных случаях в технический регламент полностью включается нормативный документ. Технические регламенты обычно дополняются методическими документами, как правило, указаниями по методам контроля или проверок соответствия продукта (услуги, процесса) требованиям регламента.

**Вопросы:**

1. Что такое Стандарт.
2. Что такое Предварительный стандарт
3. Виды стандартов.

**Лабораторная работа № 15**

**Тема:** Разработка технического задания на создание программного обеспечения.

**Цель:** изучить, требования к программной документации.

**Оборудование:** Компьютер, MS Office.

Предварительный состав программной документации

Предварительный состав программной документации должен включать в себя:

1. техническое задание;
2. текст программы;
3. описание программы;
4. программу и методики испытаний;
5. пояснительную записку;
6. описание применения;

7) руководство пользователя;

5.Условия эксплуатации

Требования к организации и оборудованию рабочих мест с ПЭВМ регламентируются СанПиН 2.2.2/2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы» и Р\_2.2.2006-05 «Гигиенические критерии оценки и классификация условий труда по показателям вредности и опасности факторов производственной среды».

6. Стадии и этапы разработки

6.1. Стадии разработки

Разработка должна быть произведена в три стадии:

1. Разработка технического задания;
2. Рабочее проектирование;
3. Внедрение;

6.2. Этапы разработки

На стадии рабочего проектирования должны быть выполнены перечисленные ниже этапы работ:

1. разработка АИСТ;
2. разработка программной документации;
3. испытания АИСТ.

На стадии внедрения должен быть выполнен этап разработки - подготовка АИСТ.

6.3. Содержание работ по этапам

На этапе разработки АИСТ должна быть выполнена работа по программированию (кодированию) и отладке программного обеспечения (АИСТ).

На этапе разработки программной документации должна быть выполнена разработка программных документов в соответствии с требованием п. 4.1. настоящего технического задания.

На этапе испытаний АИСТ должны быть выполнены перечисленные ниже виды работ:

1. проверка выполнения заданных функций АИСТ;
2. выявления и устранения недостатков в АИСТ и программной документации;
3. корректировка АИСТ и программной документации по результатам тестирований.

На этапе подготовки АИСТ должна быть выполнена работа по подготовке программного средства и программной документации для эксплуатации.

Вопросы:

1. Составить техническое задание согласно требованиям.