

ДЗ (пз5, пз6). ДОМАШНЕЕ ЗАДАНИЕ: контролируемая (контактная) самостоятельная работа студента (КСРС)

КАЧЕСТВО ПРОГРАММНОГО СРЕДСТВА

(Составили Никонова Г.В. ngvlad@mail.ru, Никонов А.В. nalva@mail.ru)

ЭЛЕКТРОННЫЙ ВАРИАНТ

<https://disk.yandex.ru/d/xgBQ5OhfmR1fdg>

1 ВВОДНЫЙ МАТЕРИАЛ

Рассмотрим методику оценки качества программного обеспечения в соответствии с требованиями стандарта ГОСТ 28195-89 [1].

Процесс оценки качества программного обеспечения осуществляется для каждой фазы его жизненного цикла и включает:

- **выбор совокупности (номенклатуры) показателей качества** оцениваемого программного средства;
- **определение значений** этих показателей;
- **сравнение полученных значений с базовыми значениями показателей.**

Под **жизненным циклом** программного обеспечения понимается период времени с момента начала предпроектного обследования до момента полного выхода программы из употребления пользователями.

В соответствии с ГОСТ 28195-89 весь период жизненного цикла программного обеспечения делится на следующие временные промежутки или фазы.

1. Анализ – этап определения требований к программному обеспечению, спецификация требований и формирования технического задания на проектирование программы.

2. Проектирование – этап разработки технического проекта.

3. Реализация – этап разработки программного обеспечения, средств тестирования и документации.

4. Тестирование – этап испытания программного обеспечения и устранение недостатков.

5. Изготовление – этап преобразования программного обеспечения в форму, готовую для поставки; завершение формирования документации.

6. Внедрение – этап подтверждения стабильной работы программного обеспечения; ввод в стадию активного применения.

7. Эксплуатация – этап применения программного обеспечения по назначению.

8. Сопровождение – этап устранения дефектов в процессе эксплуатации; усовершенствование, оптимизация и модификация используемого программного обеспечения при условии сохранности целостности программного продукта.

Оценка качества программного обеспечения на всех фазах жизненного цикла осуществляется на основе **четырёхуровневой системы показателей (рисунок 1)**.

Показатели первого уровня (факторы качества) характеризуют потребителски-ориентированные свойства программных средств, которые соответствуют потребностям пользователей. Факторы качества, собственно, определяют наиболее значимые (с точки зрения использования) свойства программ. Для оценки качества программного обеспечения используют следующие факторы:

- надёжность;
- сопровождаемость;
- удобство применения;
- эффективность;
- универсальность;
- корректность.



Рисунок 1 – Система показателей качества в соответствии с ГОСТ 28195-89

Каждый фактор представляет собой интегральную оценку, которой соответствует несколько критериев качества (комплексных показателей второго уровня).

В таблице 1 представлен состав факторов по критериям и метрикам для различных фаз жизненного цикла программных средств. Символом «+» отмечены фазы жизненного цикла, на которых определяются значения указанных метрик для расчёта соответствующих критериев и факторов. Пустые ячейки таблицы, соответствующие метрикам и фазам жизненного цикла, означают, что указанные

метрики на указанных в таблице фазах не определяются.

Таблица 1. Состав и соответствие показателей качества программных систем на различных фазах их жизненного цикла в соответствии с ГОСТ 28195-89

Факторы	Критерии	Метрики	Фаза жизненного цикла					
			Анализ	Проектирование	Реализация	Тестирование	Изготовление	Сопровождение
Надежность	Устойчивость функционирования	Средства восстановления при ошибках на входе	+	+	+	+	+	+
		Средства восстановления при сбоях оборудования	+	+	+	+	+	+
		Реализация управления средствами восстановления		+	+	+	+	+
Надежность	Работоспособность	Функционирование в заданных режимах			+	+	+	+
		Обеспечение обработки заданного объема информации			+	+	+	+
Сопровождаемость	Простота конструкции	Простота архитектуры проекта	+	+				+
		Сложность архитектуры проекта		+	+	+	+	+
		Межмодульные связи		+				
		Простота кодирования			+	+	+	
	Наглядность	Комментарии логики программного проекта			+	+	+	+
		Оформление текста программ			+	+	+	+
	Структурность	Соблюдение принципа нисходящего программирования			+	+	+	+
Удобство применения	Легкость освоения	Освоение работы программного обеспечения					+	+

		Документация для освоения					+	+
	Доступность эксплуатационных документов	Полнота документации			+	+	+	+
		Понятность документации			+	+	+	+
		Техническое исполнение документации			+	+	+	+
		Прослеживание вариантов документации			+	+	+	+
	Удобство эксплуатации и обслуживания	Эксплуатация	+	+	+	+	+	+
		Управление меню	+	+	+	+	+	+
		Функции поддержки справочной системы	+	+	+	+	+	+
		Управление данными	+	+	+	+	+	+
		Рабочие процедуры	+	+	+	+	+	+
Эффективность	Уровень автоматизации	Функции автоматизации	+	+	+	+	+	+
	Временная эффективность	Затраты времени	+		+	+	+	+
	Ресурсоемкость	Использование вычислительных ресурсов	+	+	+	+	+	+
Универсальность	Гибкость	Широта охвата функции	+	+	+	+	+	+
		Простота архитектуры проекта		+	+	+	+	+
		Сложность архитектуры проекта		+	+	+	+	+
		Сложность структуры кода программы			+	+	+	+
		Применение стандартных протоколов связи			+	+	+	+
		Применение стандартных интерфейсных программ			+	+	+	+
	Мобильность	Зависимость от используемого комплекса технических средств	+		+	+	+	+
		Зависимость от базового программного	+		+	+	+	+

		обеспечения						
		Изоляция немобильности	+		+	+	+	+
Универсальность	Модифицируемость	Простота кодирования			+	+	+	+
		Число комментариев			+	+	+	+
		Качество комментариев			+	+	+	+
		Использование описательных средств языка			+	+	+	+
		Независимость модулей			+	+	+	+
Корректность	Полнота реализации	Полнота документации разработчика	+	+	+	+	+	+
		Полнота программной документации	+		+	+	+	+
	Согласованность	Непротиворечивость документации			+	+	+	
		Непротиворечивость программы	+	+	+	+	+	
		Единообразие межмодульных и пользовательских интерфейсов	+	+	+	+	+	+
		Единообразие кодирования и определения переменных	+	+	+	+	+	+
		Соответствие документации стандартам		+	+	+	+	
		Соответствие программы стандартам программирования	+		+	+	+	
Корректность	Проверенность	Требования к полноте тестирования	+		+	+	+	+

Численное значение каждого фактора определяется на основе значений соответствующих критериев:

$$R^{\Phi} = \sum_{j=1}^N (K_j \cdot V_j^k), \quad (1)$$

где R^{Φ} – значение фактора качества,

K_j – относительное значение j -го критерия качества,

V_j^k – весовой коэффициенту j -го критерия качества,

N – количество критериев, входящих в состав фактора.

Весовые коэффициенты критериев для конкретного фактора подбираются таким образом, чтобы сумма их значений была равна единице. Величина каждого коэффициента определяется на основе субъективной оценки значимости каждого критерия качества, входящего в состав оцениваемого фактора.

Комплексные показатели второго уровня (критерии) представляют собой характеристики программных средств, которые обеспечивают достижение требуемых потребителю-ориентированных свойств.

Численное значение критерия, входящего в состав того или иного фактора, представляет собой относительную величину. Относительное значение каждого критерия соответствующего фактора определяется как отношение

$$K_j = P_j / P_j^{\text{баз}} \quad (2)$$

где P_j – абсолютное значение критерия качества для оцениваемого программного обеспечения в соответствии с рассчитываемым фактором качества;

$P_j^{\text{баз}}$ – базовое (эталонное) значение критерия, с которым сравнивается критерий оцениваемого программного обеспечения.

Эталонные значения критериев выбирают в соответствии с реально существующими программными средствами того же функционального назначения, с такими же основными параметрами, подобной структуры и применяемыми в условиях, аналогичных использованию оцениваемой программной системы.

Абсолютные значения критериев P_j рассчитываются на основе значений, соответствующих критерию метрик (см. таблицу 1) по следующему соотношению:

$$P_j = \sum_{k=1}^n (M_k \cdot V_k^M) \quad (3)$$

где M_k – итоговое значение каждой метрики соответствующего, критерия,

V_k^M – весовые коэффициенты уровня метрик качества,

n – количество метрик, входящих в состав критерия.

Весовые коэффициенты метрик для конкретного критерия подбираются таким образом, чтобы сумма их значений была равна единице. Величина каждого

коэффициента определяется на основе субъективной оценки значимости каждой метрики качества, входящей в состав оцениваемого критерия.

Метрики – это показатели качества, которые находятся на третьем уровне системы оценки качества программных систем и представляют собой абсолютную меру количественной оценки заданного критерия. В соответствии со стандартом ГОСТ 28195-89, каждому критерию соответствует свой набор метрик (см. таблицу 1), состав которого назначается конкретно для каждой фазы жизненного цикла.

Итоговые значения каждой метрики M_k рассчитываются на основе показателей качества более низкого уровня – оценочных элементов. Значения M_k вычисляются по следующей формуле:

$$M_k = \frac{\sum_{i=1}^Q m_i^k}{Q} \quad (4)$$

где m_i^k – среднее значение i -го оценочного элемента для k -й метрики, полученного на основе его Q конкретных значений.

Оценочные элементы для метрики выбираются в зависимости от их функционального назначения с учетом данных, полученных при проведении испытаний различных видов, а также по результатам эксплуатации. Процесс определения конкретных значений оценочных элементов, как правило, осуществляется группой экспертов. Каждый эксперт выставляет свою субъективную оценку, определяя значение оценочного элемента по конкретной метрике. Затем на основе выставленных значений оценочных элементов определяется m_i , по соотношению

$$m_i = \frac{\sum_{i=1}^N m_i}{N}. \quad (5)$$

Значения оценочных элементов определяются экспертным и расчетным методами. Для каждого фактора качества в соответствии с метриками (таблица 1) в ГОСТ 28195-89 определен перечень оценочных элементов с указанием метода нахождения их значений в заданных численных пределах. В таблицах 2–7 представлены составы метрик для каждого фактора качества.

Таблица 2 – Состав метрик фактора «Надёжность»

Метрики	Наименование оценочных элементов	Метод оценки	Возможные значения
---------	----------------------------------	--------------	--------------------

1. Средства восстановления при ошибках на входе	1. Наличие требований к программе по устойчивости функционирования при наличии ошибок во входных данных	Экспертный	0..1
	2. Возможность обработки ошибочных ситуаций	Экспертный	0..1
	3. Полнота обработки ошибочных ситуаций	Экспертный	0..1
	4. Наличие текстов для проверки допустимых значений входных данных	Экспертный	0..1
	5. Наличие системы контроля полноты входных данных	Экспертный	0..1
	6. Наличие средств контроля корректности входных данных	Экспертный	0..1
	7. Наличие средств контроля непротиворечивости входных данных	Экспертный	0..1
	8. Наличие проверки параметров и адресов по диапазону их значений	Экспертный	0..1
	9. Наличие обработки граничных результатов	Экспертный	0..1
	10. Наличие обработки неопределённости	Экспертный	0..1
2. Средства восстановления при сбоях оборудования	1. Наличие требований к программе по восстановлению процесса выполнения в случае сбоя ОС, процессора, внешних устройств	Экспертный	0..1
	2. Наличие требований к программе по восстановлению результатов при отказах процессора	Экспертный	0..1
	3. Наличие средств восстановления процесса в случае сбоев оборудования	Экспертный	0..1
	4. Наличие возможности разделения по времени выполнения отдельных функций программ	Экспертный	0..1
	5. Наличие возможности повторного старта с точки останова	Экспертный	0..1
3. Реализация управления средствами восстановления	1. Наличие централизованного управления процессами, конкурирующими из-за ресурсов	Экспертный	0..1
	2. Наличие возможности обходить ошибочные ситуации в процессе вычисления	Экспертный	0..1
	3. Наличие средств, обеспечивающих завершение процесса решения в случае помех	Экспертный	0..1
	4. Наличие средств, обеспечивающих выполнение	Экспертный	0..1

	программы в сокращенном объеме в случае ошибок или помех		
	5. Показатель устойчивости к искажающим воздействиям	Экспертный	$P=1-D/K$, где D – число экспериментов, в которых искажающие воздействия приводили к отказу; K – число экспериментов, в которых имитировались искажающие воздействия.
4. Функционирование в заданных режимах	1. Вероятность безотказной работы	Расчетный	$P=1-Q/N$, где Q – число зарегистрированных отказов; N – число экспериментов.
5. Обеспечение обработки заданного объема информации	1. Оценка по среднему времени восстановления	Расчетный	$Q = \begin{cases} 1, & \text{если } T_B < T_B^{\text{доп}} \\ \frac{T_B^{\text{доп}}}{T_B}, & \text{если } T_B > T_B^{\text{доп}} \end{cases}$ Здесь T_B – среднее время восстановления; $T_B^{\text{доп}}$ – допустимое среднее время восстановления.
	2. Оценка по продолжительности преобразования входного набора данных в выходной	Расчетный	$Q_{Pi} = \begin{cases} 1, & \text{если } T_{Pi} < T_{Pi}^{\text{доп}} \\ \frac{T_{Pi}^{\text{доп}}}{T_{Pi}}, & \text{если } T_{Pi} > T_{Pi}^{\text{доп}} \end{cases}$ Здесь T_{Pi} – фактическое время преобразования i -го входного набора данных; $T_{Pi}^{\text{доп}}$ – допустимое время преобразования i -го входного набора данных.

Таблица 3 – Состав метрик фактора «Сопровожаемость»

Метрики	Наименование оценочных элементов	Метод оценки	Возможные значения
1. Простота архитектуры	1. Наличие модульной схемы программы	Экспертный	0..1
	2. Оценка Программы по числу уникальных модулей	Экспертный	0..1
2. Сложность архитектуры проекта	1. Наличие ограничений на размеры модулей	Экспертный	0..1
3. Межмодульные связи	1. Наличие требований к независимости модулей программы от типов и форматов выходных данных	Экспертный	0..1
	2. наличие проверки корректности передаваемых данных	Экспертный	0..1

	3. Оценка простоты программы по числу точек входа и выхода	Расчетный	$W = \frac{1}{(D+1) \cdot (F+1)},$ где D – общее число точек входа в программу, F – общее число точек выхода из программы
	4. Осуществляется ли передача результатов работы модуля через вызывающий его модуль	Экспертный	0..1
	5. Осуществляется ли контроль за правильностью данных, поступающих в вызываемый модуль от вызываемого	Экспертный	0..1
4. Соблюдение принципа нисходящего программирования	1. Использование при построении программ метода структурного программирования	Экспертный	0..1
	2. Соблюдение принципа разработки программы сверху вниз	Экспертный	0..1
	3. Оценка программы по числу циклов с одним входом и одним выходом	Экспертный	0..1
	4. Оценка программы по числу циклов	Экспертный	0..1
5. Комментарии логики программного проекта	1. Наличие комментариев ко всем машиннозависимым частям программы	Экспертный	0..1
	2. Наличие комментариев к машиннозависимым операторам программы	Экспертный	0..1
	3. Наличие комментариев в точках входа и выхода программы	Экспертный	0..1
6. Оформление текста программы	1. Соответствие комментариев принятым соглашениям	Экспертный	0..1
	2. Наличие комментариев заголовков программы с указанием ее структурных и функциональных характеристик	Экспертный	0..1
	3. оценка ясности и точности описания последовательности функционирования всех элементов программы	Экспертный	0..1
7. Простота кодирования	1. Используется ли язык высокого уровня	Экспертный	0..1
	2. Оценка простоты программы по числу переходов по условию	Расчетный	$U = A/B$, где A – общее число переходов по условию, B – общее число исполняемых операторов.

Таблица 4 – Состав метрик фактора «Удобство применения» [2]

Метрики	Наименование оценочных элементов	Метод оценки	Возможные значения
1. Освоение работы программного обеспечения	1. Возможность освоения программных средств по документации	Экспертный	0..1
	2. Возможность освоения программных средств на контрольном примере при помощи ЭВМ	Экспертный	0..1
	3. Возможность поэтапного освоения	Экспертный	0..1
2. Документация для	1. Полнота и понятность документации	Экспертный	0..1

освоения	для освоения		
	2. Точность документации для освоения	Экспертный	0..1
	3. Техническое исполнение документации	Экспертный	0..1
3. Полнота документации	1. Наличие краткой аннотации	Экспертный	0..1
	2. Наличие описания решаемых задач	Экспертный	0..1
	3. Наличие описания структуры функций программы	Экспертный	0..1
	4. Наличие описания основных функций	Экспертный	0..1
	5. Наличие описания частных функций	Экспертный	0..1
	6. Наличие описания алгоритмов	Экспертный	0..1
	7. Наличие описания межмодульных интерфейсов	Экспертный	0..1
	8. Наличие описания пользовательских интерфейсов	Экспертный	0..1
	9. Наличие описания входных и выходных данных	Экспертный	0..1
	10. Наличие описания диагностических сообщений	Экспертный	0..1
	11. Наличие описания основных характеристик программы	Экспертный	0..1
	12. Наличие описания программной среды функционирования программы	Экспертный	0..1
	13. Достаточность документации для ввода программы в эксплуатацию	Экспертный	0..1
	14. Наличие информации о технологии переноса для мобильных программ	Экспертный	0..1
4. Точность документации	1. Соответствие оглавления содержанию документации	Экспертный	0..1
	2. Оценка оформления документации	Экспертный	0..1
	3. Грамматическая правильность изложения документации	Экспертный	0..1
	4. Отсутствие противоречий в документации	Экспертный	0..1
	5. Отсутствие неправильных ссылок в документации	Экспертный	0..1
	6. Ясность формулировок и описаний в документации	Экспертный	0..1
	7. Отсутствие неоднозначных формулировок и описаний в документации	Экспертный	0..1
	8. Правильность использования терминов в документации	Экспертный	0..1
	9. Краткость, отсутствие лишней детализации в документации	Экспертный	0..1
	10. Единство формулировок в документации	Экспертный	0..1
	11. Единство обозначений в документации	Экспертный	0..1
	12. Отсутствие ненужных повторений в документации	Экспертный	0..1
	13. Наличие нужных объяснений в документации	Экспертный	0..1
5. Понятность документации	1. Оценка стиля изложения документации	Экспертный	0..1
	2. Дидактическая отдельность документации	Экспертный	0..1
	3. Формальная отдельность	Экспертный	0..1
	4. Ясность логической структуры	Экспертный	0..1

	5. Соблюдение стандартов и правил изложения в документации	Экспертный	0..1
	6. Оценка по числу ссылок вперед в тексте документов	Экспертный	0..1
6. Техническое исполнение документации	1. Наличие оглавления документа	Экспертный	0..1
	2. Наличие предметного указателя документа	Экспертный	0..1
	3. Наличие перекрестных ссылок в документе	Экспертный	0..1
	4. Наличие всех требуемых разделов	Экспертный	0..1
	5. Соблюдение непрерывности нумерации страниц документа	Экспертный	0..1
	6. Отсутствие незаконченных разделов, абзацев, предложений в документах	Экспертный	0..1
	7. Наличие всех рисунков, чертежей, формул, таблиц в документах	Экспертный	0..1
	8. Наличие всех строк и примечаний в документе	Экспертный	0..1
	9. Логический порядок частей внутри главы документа	Экспертный	0..1
7. Прослеживание вариантов документации	1. Наличие полного перечня документации	Экспертный	0..1
8. Эксплуатация	1. Уровень языка общения пользователя с программой	Экспертный	0..1
	2. Легкость и быстрота загрузки и запуска программы	Экспертный	0..1
	3. Легкость и быстрота завершения работы программы	Экспертный	0..1
	4. Возможность распечатки содержимого программы	Экспертный	0..1
	5. Возможность приостанова и повторного запуска работы без потерь информации	Экспертный	0..1
9. Управление меню	1. Соответствие меню требованиям пользователя	Экспертный	0..1
	2. Возможность прямого перехода вверх и вниз по многоуровневому меню (пропуск уровней)	Экспертный	0..1
10. Функция поддержки справочной системы	1. Возможно управления подробностью получаемых выходных данных	Экспертный	0..1
	2. Достаточность полученной информации для продолжения работы	Экспертный	0..1
11. Управление данными	1. Обеспечение удобства ввода данных	Экспертный	0..1
	2. Легкость восприятия	Экспертный	0..1
12. Рабочие процедуры	1. Обеспечение программой выполнения предусмотренных рабочих процедур	Экспертный	0..1
	2. Достаточность информации, выдаваемой программой для составления дополнительных процедур	Экспертный	0..1

Таблица 5 – Состав метрик фактора «Эффективность» [2]

Метрики	Наименование оценочных элементов	Метод оценки	Возможные значения
1. Широта охвата	1. Проблемно-ориентированные функции	Экспертный	0..1

функций	2. Машинно-ориентированные функции	Экспертный	0..1
	3. Функции ведения и управления	Экспертный	0..1
	4. Функции ввода-вывода	Экспертный	0..1
	5. Функции защиты и проверки данных	Экспертный	0..1
	6. Функции защиты от несанкционированного доступа	Экспертный	0..1
	7. Функции контроля доступа	Экспертный	0..1
	8. Функции защиты от внесения изменений	Экспертный	0..1
	9. Наличие соответствующих границ функциональных областей	Экспертный	0..1
	10. Точность, достигаемая в результатах вычислений	Экспертный	0..1
2. Затрата времени	1. Время выполнения программ	Экспертный	0..1
	2. Время реакции и ответов	Экспертный	0..1
	3. Время подготовки	Экспертный	0..1
	4. Затраты времени на защиту данных	Экспертный	0..1
	5. Время компиляции	Экспертный	0..1
3. Использование вычислительных ресурсов	1. Требуемый объем внутренней памяти	Экспертный	0..1
	2. Требуемый объем внешней памяти	Экспертный	0..1
	3. Требуемые периферийные устройства	Экспертный	0..1
	4. Требуемое базовое программное обеспечение	Экспертный	0..1

Таблица 6 – Состав метрик фактора «Универсальность»

Метрики	Наименование оценочных элементов	Метод оценки	Возможные значения
1. Широта охвата функций	1. Оценка числа потенциальных пользователей	Экспертный	0..1
	2. Оценка числа функций ПС	Экспертный	0..1
	3. Насколько набор функций удовлетворяет требованиям пользователя	Экспертный	0..1
	4. Насколько возможно программ охватывает область решаемых пользователем задач	Экспертный	0..1
	5. Возможность настройки формата входных данных для конкретных пользователей	Экспертный	0..1
2. Простота архитектуры проекта	1. Наличие схемы иерархии модулей программы	Экспертный	0..1
	2. Оценка независимости модулей	Экспертный	0..1
	3. Оценка числа уникальных элементов	Экспертный	0..1
	4. Используется ли в текущем вызове модуля информации, полученная в предыдущем вызове	Экспертный	0..1
	5. Оценка организации точек входа и выхода модуля	Экспертный	0..1
	6. Наличие описания атрибутов модуля	Экспертный	0..1
3. Сложность архитектуры проекта	1. Оценка программ по числу переходов и точек ветвления	Экспертный	0..1
4. Сложность структуры кода программы	1. Использование метода пошагового уточнения	Экспертный	0..1
	2. Наличие описания структуры программ	Экспертный	0..1
	3. Наличие описания связей между	Экспертный	0..1

	элементами структуры программы		
	4. Наличие в программе повторного выполнения функций (подпрограммы)	Экспертный	0..1
5. Применение стандартных протоколов связи	1. Использование стандартных протоколов связи	Экспертный	0..1
6. Применение стандартных интерфейсных программ	1. Использование стандартных интерфейсных подпрограмм	Экспертный	0..1
7. Зависимость от используемого комплекса технических средств	1. Оценка зависимости программ от емкости оперативной памяти	Экспертный	0..1
	2. Оценка зависимости временных характеристик программы от скорости вычислений ЭВМ	Экспертный	0..1
	3. Оценка зависимости функционирования программы от числа внешних запоминающих устройств в их общей емкости	Экспертный	0..1
	4. Оценка зависимости функционирования программы от специальных устройств ввода-вывода	Экспертный	0..1
8. Зависимость от базового ПО	1. Применение специальных языков программирования	Экспертный	0..1
	2. Оценка зависимости программы от программ операционной системы	Экспертный	0..1
	3. Зависимость от других программных средств	Экспертный	0..1
9. Изоляция немобильности	1. Оценка локализации переносимой части программы	Экспертный	0..1
10. Простота кодирования	1. Оценка использования отрицательных или булевых выражений	Экспертный	0..1
	2. Оценка программы по использованию условных переходов	Экспертный	0..1
	3. Оценка программы по использованию безусловных переходов	Экспертный	0..1
	4. Оформление процедур входа выхода из циклов	Экспертный	0..1
	5. Ограничения на модификацию переменной индексации в цикле	Экспертный	0..1
	6. Оценка модулей по направлению потока управления	Экспертный	0..1
	7. Оценка программы по использованию локальных переменных	Экспертный	0..1
11. Число комментариев	1. Оценка программы по числу комментариев	Экспертный	0..1
12. Качество комментариев	1. Наличие заголовка в программе	Экспертный	0..1
	2. Комментарии к точкам ветвления	Экспертный	0..1
	3. Комментарии к машинозависимым частям программы	Экспертный	0..1
	4. Комментарии к машинозависимым операторам программы	Экспертный	0..1
	5. Комментарии к операторам объявления переменных	Экспертный	0..1
	6. Оценка семантики операторов	Экспертный	0..1

	7. Наличие соглашений по форме представления переменных	Экспертный	0..1
	8. Наличие общих комментариев к программам	Экспертный	0..1
13. Использование описательных средств языка	1. Использование языков высокого уровня	Экспертный	0..1
	2. Семантика имен используемых переменных	Экспертный	0..1
	3. Использование отступов, сдвигов, пропусков при формировании текста	Экспертный	0..1
	4. Размещение операторов по строкам	Экспертный	0..1
14. Независимость модулей	1. Передача информации для управления по параметрам	Экспертный	0..1
	2. Параметрическая передача входных данных	Экспертный	0..1
	3. Наличие передачи результатов работы между модулями	Экспертный	0..1
	4. Наличие проверки правильности данных, получаемых модулями от вызываемого модуля	Экспертный	0..1
	5. Использование общих областей памяти	Экспертный	0..1

Таблица 7 – Состав метрик фактора «Корректность» [2]

Метрики	Наименование оценочных элементов	Метод оценки	Возможные значения
1. Полнота документации разработчика	1. Наличие всех необходимых документов для понимания и использования ПС	Экспертный	0..1
	2. Наличие описания и схемы иерархии модулей программы	Экспертный	0..1
	3. Наличие описания основных функций	Экспертный	0..1
	4. Наличие описания частных функций	Экспертный	0..1
	5. Наличие описания данных	Экспертный	0..1
	6. Наличие описания алгоритма	Экспертный	0..1
	7. Наличие описания интерфейса между модулями	Экспертный	0..1
	8. Наличие описания интерфейса с пользователем	Экспертный	0..1
	9. Наличие описания используемых числовых методов	Экспертный	0..1
	10. Указаны ли все числовые методы	Экспертный	0..1
	11. Наличие описания всех параметров	Экспертный	0..1
	12. Наличие описания методов настройки системы	Экспертный	0..1
	13. Наличие описания всех диагностических сообщений	Экспертный	0..1
	14. Наличие описания способов проверки работоспособности программы	Экспертный	0..1
	1. Реализация всех исходных модулей	Экспертный	0..1
	2. Реализация всех основных функций	Экспертный	0..1
	3. Реализация всех частных функций	Экспертный	0..1
	4. Реализация всех алгоритмов	Экспертный	0..1
	5. Реализация всех возможностей в системе	Экспертный	0..1
	6. Реализация всех интерфейсов между модулями	Экспертный	0..1

	7. Реализация возможности настройки системы	Экспертный	0..1
	8. Реализация диагностики всех граничных и аварийных ситуаций	Экспертный	0..1
	9. Наличие определения всех данных (переменные, индексы, массивы и проч.)	Экспертный	0..1
	10. Наличие интерфейсов с пользователем	Экспертный	0..1
3. Непротиворечивость документации	1. Отсутствие противоречий в описании частных функций	Экспертный	0..1
	2. Отсутствие противоречий в описании основных функций в разных документах	Экспертный	0..1
	3. Отсутствие противоречий в описании алгоритмов	Экспертный	0..1
	4. Отсутствие противоречий в описании взаимосвязей в системе	Экспертный	0..1
	5. Отсутствие противоречий в описании интерфейсов между модулями	Экспертный	0..1
	6. Отсутствие противоречий в описании интерфейсов с пользователями	Экспертный	0..1
	7. Отсутствие противоречий в описании настройки системы	Экспертный	0..1
	8. Отсутствие противоречий в описании диагностических сообщений	Экспертный	0..1
	9. Отсутствие противоречий в описании диагностических сообщений	Экспертный	0..1
	10. Отсутствие противоречий в описании данных	Экспертный	0..1
4. Непротиворечивость программы	1. Отсутствие противоречий в выполнении основных функций	Экспертный	0..1
	2. Отсутствие противоречий в выполнении частных функций	Экспертный	0..1
	3. Отсутствие противоречий в выполнении алгоритмов	Экспертный	0..1
	4. Правильность взаимосвязей	Экспертный	0..1
	5. Правильность реализации интерфейса между модулями	Экспертный	0..1
	6. Правильность реализации интерфейса с пользователем	Экспертный	0..1
	7. Отсутствие противоречий в настройке системы	Экспертный	0..1
	8. Отсутствие противоречий в диагностике системы	Экспертный	0..1
	9. Отсутствие противоречий в общих переменных	Экспертный	0..1
5. Единообразие межмодульных и пользовательских интерфейсов	1. Единообразие способов вызова модулей	Экспертный	0..1
	2. Единообразие процедур возврата управления из модулей	Экспертный	0..1
	3. Единообразие способов сохранения информации для возврата	Экспертный	0..1
	4. Единообразие способов восстановления информации для возврата	Экспертный	0..1
	5. Единообразие организации списков передаваемых параметров	Экспертный	0..1
6. Единообразие кодирования и определения	1. Единообразие наименования каждой переменной и константы	Экспертный	0..1
	2. Все ли одинаковые константы встречаются	Экспертный	0..1

переменных	во всех программах под одинаковыми именами		
	3. Единообразие определения внешних данных во всех программах	Экспертный	0..1
	4. Используются ли разные идентификаторы для разных переменных	Экспертный	0..1
	5. Все ли общие переменные объявлены как общие переменные	Экспертный	0..1
	6. Наличие определений одинаковых атрибутов	Экспертный	0..1
7. Соответствие документации стандартам	1. Комплектность документации в соответствии со стандартами	Экспертный	0..1
	2. Правильное оформление частей документов	Экспертный	0..1
	3. Правильное оформление титульных и заглавных листов документов	Экспертный	0..1
	4. Наличие в документах всех разделов в соответствии со стандартами	Экспертный	0..1
	5. Полнота содержания разделов в соответствии со стандартами	Экспертный	0..1
	6. Деление документов на структурные элементы: разделы, подразделы, пункты, подпункты	Экспертный	0..1
8. Соответствие программы стандартам программирования	1. Соответствие организации вычислительного процесса эксплуатационной документации	Экспертный	0..1
	2. Правильность заданий на выполнение программы, правильность написания управляющих операторов (отсутствие ошибок)	Экспертный	0..1
	3. Отсутствие ошибок в описании действий пользователя	Экспертный	0..1
	4. Отсутствие ошибок в описании запуска	Экспертный	0..1
	5. Отсутствие ошибок в описании генерации	Экспертный	0..1
	6. Отсутствие ошибок в описании настройки	Экспертный	0..1
9. Требования в полноте тестирования	1. Наличие требований к тестированию программ	Экспертный	0..1
	2. Достаточность требований к тестированию программ	Экспертный	0..1
	3. Отношение числа модулей, отработавших в процессе тестирования и отладки, к общему числу модулей	Расчетный	Q_T^M/Q_0^M , где Q_T^M – число модулей, отработавших в процессе тестирования и отладки; Q_0^M – общее число модулей.
	4. Отношение числа логических боков, отработавших в процессе тестирования и отладки к общему числу логических блоков в программе	Расчетный	Q_T^B/Q_0^B , где Q_T^B – число блоков, отработавших в процессе тестирования и отладки; Q_0^B – общее число блоков в программе.

2 ПРИМЕР РЕШЕНИЯ ЗАДАЧИ ПО ОПРЕДЕЛЕНИЮ ПОКАЗАТЕЛЕЙ КАЧЕСТВА. ОЦЕНКА НАДЕЖНОСТИ И КОРРЕКТНОСТИ.

ВЫПОЛНЕНИЕ ЗАДАНИЯ НА КСРС ПРОВОДИТСЯ В СООТВЕТСТВИИ С ПРИВЕДЁННЫМ НИЖЕ ПРИМЕРОМ

В ходе выполнения данного задания необходимо:

- 1) изучить и разобрать материал раздела 1;
- 2) изучить и разобрать материал раздела 2. Оценить и осмыслить предстоящее решение задачи по примеру;
- 3) разобрать материал подраздела 2.1. Дать смысловую трактовку отдельным действиям в решении задачи;
- 4) интегрально осмыслить задачу примера раздела 2. Привести собственные суждения и выводы по методике решения задачи и полученным в ходе решения результатам.

ЗАДАНИЕ. 1. Оценить качество программного обеспечения для решения проектных (например, экономических) задач на ранней стадии разработки (в фазе анализа) по результатам проделанной работы, нашедших отражение в проектных документах и отчётах исполнителей.

2. Провести оценку качества программы на основе факторов надёжности и корректности. Для оценки возьмите любую программу, которая использовалась в заданиях ПЗ1–ПЗ4. Выступая в качестве эксперта, установите значения оценочных элементов в таблице 8 такими, какими именно вы их представляете, и затем продолжайте расчёты с применением значений оценочных элементов из таблицы 8, установленных вами. Используйте методику, приведённую в подразделе 2.1.

Значения всех базовых показателей надёжности принять на уровне 0,63. Значения всех базовых показателей корректности установить на уровне 0,9.

При оценке качества особое внимание уделить требованиям к полноте реализации программного средства. Значения оценочных элементов по результатам проделанной работы представлены в таблице 8.

Таблица 8 – Значения оценочных элементов надёжности и корректности

Метрики	Метрика	Оценочные элементы	Оценка
---------	---------	--------------------	--------

Фактор надёжности			
1. Устойчивость функционирования	1. Средства восстановления при ошибках на входе	1. Наличие требований к программе по устойчивости функционирования при наличии ошибок во входных данных	1
		2. Возможность обработки ошибочных ситуаций	0,5
		3. Полнота обработки ошибочных ситуаций	0,5
		4. Наличие текстов для проверки допустимых значений входных данных	1
		5. Наличие системы контроля полноты входных данных	0
		6. Наличие средств контроля корректности входных данных	1
		7. Наличие средств контроля непротиворечивости входных данных	1
		8. Наличие проверки параметров и адресов по диапазону их значений	1
		9. Наличие обработки граничных результатов	1
		10. Наличие обработки неопределённости	1
2. Средства восстановления при сбоях оборудования		1. Наличие требований к программе по восстановлению процесса выполнения в случае сбоя ОС, процессора, внешних устройств	1
		2. Наличие требований к программе по восстановлению результатов при отказах процессора	0
		3. Наличие средств восстановления процесса в случае сбоев оборудования	0
		4. Наличие возможности разделения по времени выполнения отдельных функций программ	1
		5. Наличие возможности повторного старта с точки останова	1
Фактор корректности			
1. Полнота реализации	1. Полнота документации разработчика	1. Наличие всех необходимых документов для понимания и использования ПС	1
		2. Наличие описания и схемы иерархии модулей программы	0,5
		3. Наличие описания основных функций	0,5
		4. Наличие описания частных функций	0
		5. Наличие описания данных	1
		6. Наличие описания алгоритма	1
		7. Наличие описания интерфейса между модулями	1
		8. Наличие описания интерфейса с пользователем	1
		9. Наличие описания используемых числовых методов	0
		10. Указаны ли все числовые методы	0,5
		11. Наличие описания всех параметров	1
		12. Наличие описания методов настройки системы	0,5
		13. Наличие описания всех диагностических сообщений	1
		14. Наличие описания способов проверки работоспособности программы	0,5
2. Согласованность	1. Единообразие межмодульных и пользовательских интерфейсов	1. Единообразие способов вызова модулей	0,6
		2. Единообразие процедур возврата управления из модулей	0,5
		3. Единообразие способов сохранения информации для возврата	1
		4. Единообразие способов восстановления информации для возврата	1
		5. Единообразие организации списков передаваемых параметров	0
	2. Единообразие кодирования и определения переменных	1. Единообразие наименования каждой переменной и константы	0
		2. Все ли одинаковые константы встречаются во всех программах под одинаковыми именами	1
		3. Единообразие определения внешних данных во всех	1

		программах	
		4. Используются ли разные идентификаторы для разных переменных	0,5
		5. Все ли общие переменные объявлены как общие переменные	1
		6. Наличие определений одинаковых атрибутов	0
	3. Требования к соблюдению стандартов	1. Соответствие организации вычислительного процесса эксплуатационной документации	0,8
		2. Правильность заданий на выполнение программы, правильность написания управляющих операторов (отсутствие ошибок)	1
		3. Отсутствие ошибок в описании действий пользователя	1
		4. Отсутствие ошибок в описании запуска	0,5
		5. Отсутствие ошибок в описании генерации	0,5
		6. Отсутствие ошибок в описании настройки	0
3. Проверен-ность	1. Требования в полноте тестирования	1. Наличие требований к тестированию программ	1
		2. Достаточность требований к тестированию программ	1

2.1 РЕШЕНИЕ ЗАДАЧИ

Из условия задачи известно, что необходимо провести оценку двух факторов: надёжность и корректность. Проведём расчет последовательно – сначала для надежности, затем для корректности.

На фазе анализа фактор надежности оценивается по одному критерию устойчивости функционирования, которая рассчитывается по двум метрикам:

- требования к средствам восстановления при ошибках на входе;
- требования к средствам восстановления при сбоях оборудования.

Состав и значения оценочных элементов указанных метрик представлены в таблице исходных данных.

Определим итоговое значение метрики M_1 «Средства восстановления при ошибках на входе». В соответствии с таблицей 8 исходных данных $m^1_1=1$; $m^1_2=0,5$; $m^1_3=0,5$; $m^1_4=1$; $m^1_5=0$; $m^1_6=1$; $m^1_7=1$; $m^1_8=1$; $m^1_9=1$; $m^1_{10}=1$; $Q=10$, тогда

$$M_1 = \frac{\sum_{q=1}^Q m_q^2}{Q} = \frac{m_1^2 + m_2^2 + m_3^2 + m_4^2 + m_5^2}{Q}$$

$$= \frac{1 + 0,5 + 0,5 + 1 + 0 + 1 + 1 + 1 + 1 + 1}{10} = 0,8.$$

(6)

Итоговое значение метрики M_2 «средства восстановления при сбоях

оборудования» определим по следующим исходным данным: $m^I_1=1$; $m^I_2=0$; $m^I_3=0$; $m^I_4=1$; $m^I_5=1$; $Q=5$. Тогда получим:

$$M_2 = \frac{\sum_{q=1}^Q m_q^2}{Q} = \frac{m_1^2 + m_2^2 + m_3^2 + m_4^2 + m_5^2}{Q} = \frac{1 + 0 + 0 + 1 + 1}{5} = 0,6. \quad (7)$$

На основе рассчитанных значений метрик вычислим абсолютное значение критерия P_1 «Устойчивость функционирования». При расчёте значения критерия отдадим предпочтение средствам восстановления при сбоях оборудования M_2 . Важность этой метрики определим весовым коэффициентом $V^M_2=0,8$. Тогда весовой коэффициент для M_1 будет равен $V^M_1=0,2$, так как в сумме все весовые коэффициенты должны давать 1. Следовательно,

$$P_1 = \sum_{k=1}^n (M_k \cdot V_k^M) = (V_1^M \cdot M_1 + V_2^M \cdot M_2) = (0,2 \cdot 0,8 + 0,8 \cdot 0,6) = 0,64. \quad (8)$$

Определим относительное значение критерия K_1 «Устойчивость функционирования». По условию задачи базовое значение критерия $P_1^{\text{баз}}=0,63$, отсюда можно записать:

$$K_1 = P_1 / P_1^{\text{баз}} = 0,64 / 0,63 = 1,02. \quad (9)$$

Поскольку на фазе жизненного цикла «Анализ» для оценки надежности используется только критерий «Устойчивость функционирования», то численное значение этого фактора равно значению самого критерия, следовательно, $R_{\phi 1} = K_1 = 1,02$.

Проведем расчёт фактора «Корректность», который на фазе анализа жизненного цикла программы определяется значениями трех критериев:

- полнота реализации;
- согласованность;
- проверенность.

Состав критериев и метрик для фазы анализа представлен в таблице исходных данных для задачи (см. таблицу 8).

Определим итоговые значения метрик для критерия «Полнота реализации».

В соответствии с ГОСТ 28195-89 указанный критерий определяется на основе одной метрики M_1 «требования, предъявляемые к полноте документации разработчика». В соответствии с таблицей 9 исходных данных значения оценочных элементов M_1 следующие:

$$m^I_1=1; m^I_2=0,5; m^I_3=0,5; m^I_4=1; m^I_5=0; m^I_6=1; m^I_7=1; m^I_8=1; m^I_9=1; m^I_{10}=1; m^I_{11}=1;$$

$$m^l_{12}=0,5; m^l_{13}=1; m^l_{14}=0,5; Q=14.$$

Тогда значение метрики

$$M_1 = \frac{\sum_{q=1}^Q m_q^2}{Q} = \frac{m_1^2 + m_2^2 + m_3^2 + m_4^2 + m_5^2}{Q} = \frac{1+0,5+0,5+0+1+1+1+1+0+0,5+1+0,5+1+0,5}{14} = 0,68. \quad (10)$$

На основе полученной метрики определим абсолютное значение критерия «Полнота реализации». Поскольку критерий определяется всего одной метрикой, то абсолютное значение критерия будет иметь значение $P_1 = M_1 = 0,68$.

Определим относительное значение критерия K_2 «Полнота реализации». По условию задачи базовое значение критерия $P^{баз}_1 = 0,9$, отсюда:

$$K_1 = P_1 / P^{баз}_1 = 0,68 / 0,9 = 0,75. \quad (11)$$

Определим итоговые значения метрик для критерия «Согласованность». В соответствии с ГОСТ 28195-89 этот критерий определяется на основе трёх метрик:

- M_1 – «Требования, предъявляемые к единообразию интерфейсов между модулями и пользователями»;
- M_2 – «Требования, предъявляемые к единообразию кодирования, символики и определения общих переменных»;
- M_3 – «Требования, предъявляемые к соответствию стандартам программирования».

Начнём с определения M_1 . В соответствии с таблицей 9 исходных данных значения оценочных элементов для определения значения M_1 следующие: $m^l_1=1$; $m^l_2=0,5$; $m^l_3=0,5$; $m^l_4=0$; $m^l_5=1$; $Q=5$, тогда:

$$M_1 = \frac{\sum_{q=1}^Q m_q^1}{Q} = \frac{m_1^1 + m_2^1 + m_3^1 + m_4^1 + m_5^1}{Q} = \frac{1+0,5+0,5+0+1}{5} = 0,6. \quad (12)$$

Определим M_2 . В соответствии с таблицей 9 исходных данных значения оценочных элементов для определения M_2 следующие:

$$\begin{aligned} m^l_1=0; m^l_2=1; m^l_3=1; m^l_4=0,5; m^l_5=1; m^l_6=1; Q=6, \text{ тогда:} \\ M_2 = \frac{\sum_{q=1}^Q m_q^2}{Q} = \frac{(m_1^2 + m_2^2 + m_3^2 + m_4^2 + m_5^2 + m_6^2)}{Q} = \frac{0 + 1 + 1 + 0,5 + 1 + 0}{6} \\ = 0,58. \end{aligned} \quad (13)$$

Определим M_3 . В соответствии с таблицей исходных данных значения оценочных элементов для определения значения M_3 следующие: $m^3_1=0,8$; $m^3_2=1$; $m^3_3=1$; $m^3_4=0,5$; $m^3_5=0,5$; $m^3_6=1$; $Q=6$, тогда:

$$M_3 = \frac{\sum_{q=1}^Q m_q^3}{Q} = \frac{m_1^3 + m_2^3 + m_3^3 + m_4^3 + m_5^3 + m_6^3}{Q} = \frac{0,8 + 1 + 1 + 0,5 + 0,5 + 1}{6} = 0,8. \quad (14)$$

На основе полученных значений метрик определим абсолютное значение критерия «Согласованность». Критерий определяется тремя метриками. Значимость метрик критерия равна, поэтому весовые коэффициенты должны иметь приблизительно равные значения и в сумме давать единицу. Установим следующие значения весовых коэффициентов для метрик: $V^M_1=0,33$; $V^M_2=0,33$; $V^M_3=0,34$, тогда:

$$P_2 = \sum_{k=1}^n (M_k \cdot V_k^M) = (V_1^M \cdot M_1 + V_2^M \cdot M_2 + V_3^M \cdot M_3) = (0,33 \cdot 0,6 + 0,33 \cdot 0,58 + 0,34 \cdot 0,8) = 0,66. \quad (15)$$

Определим относительное значение критерия K_2 «Согласованность». По условию задачи базовое значение критерия $P_2^{баз} = 0,9$, следовательно,

$$K_2 = P_2 / P_2^{баз} = 0,66 / 0,9 = 0,73. \quad (16)$$

Определим итоговые значения метрик для критерия «Проверенность». В соответствии с ГОСТ 28195-89 «Проверенность» оценивается на основе одной метрики M_1 «Требования, предъявляемые к полноте тестирования». Значение метрики определяется на основе значений двух оценочных элементов (см. таблицу 9). Исходя из исходных данных, оценочные элементы рассматриваемой метрики имеют следующие значения: $m^1_1=1$; $m^1_2=1$; $Q=2$, тогда получим:

$$M_1 = \frac{\sum_{q=1}^Q m_q^1}{Q} = \frac{m_1^1 + m_2^1}{Q} = \frac{1+1}{2} = 1. \quad (17)$$

Поскольку критерий P_3 «Проверенность» определяется только одной метрикой, можно утверждать, что $P_3 = M_1 = 1$.

Относительное значение критерия определим, используя соотношение $K_3 = P_3 / P_3^{баз} = 1 / 0,9 = 1,1$, т. к. по условию задачи известно, что $P_3^{баз} = 0,9$.

По полученным относительным значениям критериев фактора «Корректность» определим уровень этого фактора

$$R_2^\Phi = \sum_{j=1}^N (K_j \cdot V_j^k) = (V_1^k \cdot K_1 + V_2^k \cdot K_2 + V_3^k \cdot K_3). \quad (18)$$

Установим значения весовых коэффициентов V^k_1 , V^k_2 , V^k_3 . По условию задачи необходимо уделить особое внимание требованиям к полноте реализации программного средства. Следовательно, значимость критерия K_1 «Полнота реализации» выше прочих критериев. Учтем это обстоятельство при установке значений весовых коэффициентов и определим их следующим образом: $V^k_1 = 0,5$,

$V_2^k=0,25$, $V_3^k=0,25$, так как значимость K_1 выше, а сумма коэффициентов должна равняться единице. Отсюда получим:

$$R_2^\phi = (V_1^k \cdot K_1 + V_2^k \cdot K_2 + V_3^k \cdot K_3) = 0,5 \cdot 0,75 + 0,25 \cdot 0,73 + 0,25 \cdot 1,1 = 0,83. \quad (19)$$

Таким образом, в результате вычислений получены следующие результаты: $R^{\phi_1} = 1,02$, $R^{\phi_2} = 0,83$.

Можно сделать вывод, что разработчик программного средства обеспечил установленный уровень качества по надёжности и не обеспечил требуемый уровень корректности программы.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ГОСТ 28195-89. Оценка качества программных средств. Общие положения. М. : Государственный комитет СССР по стандартам, 1989.
2. Черников Б.В., Поклонов Б.Е. Оценка качества программного обеспечения. Практикум. – М., ИД «Форум»–ИНФРА–М. – 2012. – 400 с.
- 3.