Министерство науки и высшего образования Российской Федерации

ФГАОУ ВО «УрФУ имени первого Президента России Б.Н.Ельцина»

ИРИТ-РТФ

Центр ускоренного обучения

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ N 5**

по дисциплине «Проектирование информационных систем»

**Тема: Оценка качества ПО по ГОСТ 28195-89**

Студенты гр. РИВ-420906у А.В. Васильев

А.Д. Бирючев

А.В. Литвиненко

Д.А. Махонин

С.Г. Миронов

Преподаватель: С.И. Тимошенко,

доц., к.т.н.

Екатеринбург 2025

СОДЕРЖАНИЕ

[1 Постановка задачи 3](#_Toc209883720)

[2 Анализ поставленной задачи 5](#_Toc209883721)

[3 Описание результатов 6](#_Toc209883722)

[3.1 Выбор критериев, метрик и оценочных элементов для заданного подкласса программных средств 6](#_Toc209883723)

[3.2 Описание программы для произведения расчётов по формулам. 7](#_Toc209883724)

[4 Анализ результатов 12](#_Toc209883725)

# 1 Постановка задачи

Рассчитать фактор надежности по ГОСТ 28195-89 для гипотетического программного обеспечения, относящегося к классу прикладных программ для научных исследований для фазы сопровождения. ГОСТ 28195-89 можно найти в папке «Стандарты» в форматах .pdf и многостраничного .tif (файл можно правильно открыть с помощью программы «Просмотр фотографий Windows»). Для расчета использовать параметры по вариантам из таблицы 1. Весовые коэффициенты для метрик и критериев выбрать одинаковыми внутри каждой группы.

Таблица 1 – Задания по вариантам

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Названия параметров | Значения параметров по вариантам | | | | |
| Вариант 1 | Вариант 2 | Вариант 3 | Вариант 4 | Вариант 5 |
| Число зарегистрированных отказов Q | 5 | 6 | 7 | 8 | 9 |
| Число экспериментов N | 1000 | 1200 | 1300 | 1400 | 1500 |
| Cреднее время восстановления Тв подчиняется равномерному распределению в интервале (оценить по выборке из 100 значений), с | [0,7; 1,2] | [0,8; 1,4] | [0,6; 1,0] | [0,5; 1,3] | [0,9; 1,6] |
| Допустимое время восстановления , с | 0,85 | 0,95 | 0,65 | 0,75 | 0,55 |
| Фактическая продолжительность преобразования i-го входного набора данных подчиняется равномерному закону распределения в интервале (оценить по выборке из 200 значений), с | [9; 14] | [8; 12] | [10; 16] | [6; 10] | [5; 10] |
| Допустимое время преобразования i-го входного набора данных , с | 12 | 14 | 10 | 16 | 15 |
| Базовые критерии надежности выбрать равными | 0,95 | 0,94 | 0,9 | 0,96 | 0,98 |

Все четыре уровня показателей качества, описанных в ГОСТ 28195-89, можно посмотреть в файле “Показатели качества по ГОСТ 28195-89.jpg”, находящемся в папке со стандартами.

Примерная последовательность расчетов следующая:

* по таблице 2 стандарта для выбранного фактора (с использованием номеров критериев из табл. 1) выбрать для заданного подкласса программных средств (ПС) те критерии, которые для него используются (обозначены знаком «+»);
* для указанной в задании фазы жизненного цикла выбрать древовидную структуру (см. для фактора надежность с. 11); по этой структуре определить входящие в критерии метрики;
* по таблице 5 «Оценочные элементы фактора надежность ПС» выбрать оценочные элементы для найденных на предыдущем шаге метрик;
* рассчитать оценочные элементы по правилам, указанным в поле «Оценка» таблицы 5, с учетом данных, приведенных в задании на лабораторную работу; если для оценочного элемента получается несколько значений, то их необходимо усреднить (см. п. 3.9 на с. 37 стандарта);
* рассчитать метрики по формуле 3 (см. с. 37 стандарта); чтобы понять кодировку оценочных элементов в таблицах 5-10, прочитать примечание на с. 36 стандарта;
* рассчитать абсолютные показатели критериев по формуле 4 (см. с. 37 стандарта) с учетом п. 3.6 стандарта (ограничение на весовые коэффициенты), а также ограничений, определенных в текущем задании;
* рассчитать относительные показатели критериев по формуле 5 (см. с. 37 стандарта) с учетом данных, определенных в текущем задании;
* рассчитать фактор надежности по формуле 6 (см. с. 37 стандарта).

Результаты работы оформить по шаблону из файла ”Шаблон\_отчета\_по\_лабораторной\_работе.odt”.

# 2 Анализ поставленной задачи

В рамках выполнения работы необходимо на базе ГОСТ 28195-89 произвести расчёты по следующей последовательности:

1. выбрать заданный подкласс программных средств;
2. выбрать древовидную структуру для указанной фазы жизненно цикла;
3. выбрать оценочные элементы;
4. рассчитать оценочные элементы по правилам;
5. рассчитать метрики по заданным формулам;
6. рассчитать абсолютные показатели критериев;
7. рассчитать относительные показатели критериев;
8. рассчитать фактор надежности.

# 3 Описание результатов

## 3.1 Выбор критериев, метрик и оценочных элементов для заданного подкласса программных средств

В нашем случае происходит выбор критериев для гипотетического программного обеспечения, относящегося к классу прикладных программ для научных исследований для фазы сопровождения. А также было учтено, что весовые коэффициенты для метрик и критериев выбрать одинаковыми внутри каждой группы.

Исходя из данной информации было определено что у данных прикладных программ для научных исследований показатели характеризуются как работоспособность. В нашем случае фактор надежности имеет критерий работоспособности, который в свою очередь имеет следующие метрики:

* функционирование в заданных режимах;
* обеспечение обработки заданного объема информации.

Указанные выше метрики имеют следующие оценочные элементы:

* вероятность безотказной работы (Н0401);
* оценка по среднему времени восстановления (Н0501);
* оценка по продолжительности преобразования входного набора данных в выходной (Н0502).

Все полученные данные были учтены и далее производится расчёт по формулам.

## 3.2 Описание программы для произведения расчётов по формулам.

Основными формулами для произведения расчётов стали формулы, представленные на рисунке 1.

Изображение выглядит как текст, снимок экрана, Шрифт, документ

Автоматически созданное описание

Рисунок 1 – Формулы для расчётов

Ниже представлен код класса, который включает в себя все функции, связанные с оценкой надежности, включая вероятности, среднее время восстановления и продолжительность преобразования. Код можно увидеть на листинге 1.

Листинг 1 – Код класса для расчёта надежности

//Класс для оценки надежности

public class Reliability {

private static final Random RANDOM = new Random();

//Вероятность безотказной работы P

public static double probabilityWithoutFailure(int Q, int N) {

if (N <= 0) throw new IllegalArgumentException("N должно быть > 0");

return 1.0 - (double) Q / N;

}

//Оценка среднего времени восстановления Qв

public static double averageRecoveryTime(double leftSide, double rightSide, double allowedRecoveryTime) {

final int SAMPLE\_SIZE = 100;

double[] samples = new double[SAMPLE\_SIZE];

for (int i = 0; i < SAMPLE\_SIZE; i++) {

samples[i] = randomInRange(leftSide, rightSide);

}

double meanTv = Arrays.stream(samples).average().orElse(0.0);

return (meanTv <= allowedRecoveryTime) ? 1.0 : (allowedRecoveryTime / meanTv);

}

//Оценка продолжительности преобразования Qп

public static double transformationDuration(double leftSide, double rightSide, double allowedConversionTime) {

final int SAMPLE\_SIZE = 200;

double sum = 0.0;

for (int i = 0; i < SAMPLE\_SIZE; i++) {

double val = randomInRange(leftSide, rightSide);

double score = (val <= allowedConversionTime) ? 1.0 : (allowedConversionTime / val);

sum += score;

}

return sum / SAMPLE\_SIZE;

}

private static double randomInRange(double min, double max) {

return min + (max - min) \* RANDOM.nextDouble();

}

}

Класс Reliability содержит статические методы для вычисления показателей надежности. Метод probabilityWithoutFailure вычисляет вероятность безотказной работы на основе количества зарегистрированных отказов и количества экспериментов. Метод averageRecoveryTime оценивает среднее время восстановления, а метод transformationDuration оценивает продолжительность преобразования.

Далее был создан класс, в котором находятся методы для расчёта итоговой оценки, абсолютных и относительных показателей. Код класса можно увидеть на листинге 2.

Листинг 2 – Код класса Metrics

//Класс для расчётов метрик

public class Metrics {

//Итоговая оценка

public static double finalGrade(double qv, double qp) {

return (qv + qp) / 2.0;

}

//Абсолютные показатели

public static double absoluteIndicator(double firstMetric, double secondMetric) {

return 0.5 \* firstMetric + 0.5 \* secondMetric;

}

//Относительные показатели

public static double relativeIndicator(double P\_ij, double rho\_ij) {

if (rho\_ij == 0.0) throw new IllegalArgumentException("rho\_ij должно быть != 0");

return P\_ij / rho\_ij;

}

}

Класс Metrics содержит статические методы для вычисления метрик надежности. Первый метод finalGrade принимает оценки времени восстановления и преобразования, возвращая их среднее значение. Второй метод absoluteIndicator принимает два значения метрик и возвращает их среднее. Третий метод relativeIndicator вычисляет относительный показатель, принимая абсолютный показатель и базовый критерий надежности. Четвёртый метод reliabilityFactor принимает массив коэффициентов и двумерный массив значений, возвращая показатель надёжности.

Далее был написан главный класс Main, который инициализирует параметры, вызывает методы других классов и выводит результаты. Класс Main представлен на листинге 3.

Листинг 3 – Код класса Main

//Главный класс для запуска

public class Main {

public static void main(String[] args) {

//Параметры

int Q = 5; //количество зарегистрированных отказов

int N = 1000; //количество экспериментов

//Для Qв (T\_v)

double tvLeft = 0.7;

double tvRight = 1.2;

double T\_dop\_v = 0.85; //допустимое среднее время восстановления

//Для Qп (T\_i)

double tiLeft = 9.0;

double tiRight = 14.0;

double T\_dop\_i = 12.0; //допустимое время преобразования

double rho = 0.95; //базовый критерий надежности

//Вычисления согласно выбранным формулам

double P = Reliability.probabilityWithoutFailure(Q, N);

double Qv = Reliability.averageRecoveryTime(tvLeft, tvRight, T\_dop\_v);

double Qp = Reliability.transformationDuration(tiLeft, tiRight, T\_dop\_i);

double overall = Metrics.finalGrade(Qv, Qp);

double P\_ij = Metrics.absoluteIndicator(Qv, Qp);

double K\_ij = Metrics.relativeIndicator(P\_ij, rho);

//Вывод результатов

System.out.printf("Вероятность безотказной работы: %.6f%n", P);

System.out.printf("Оценка по среднему времени восстановления: %.6f%n", Qv);

System.out.printf("Оценка по продолжительности преобразования: %.6f%n", Qp);

System.out.printf("Итоговая оценка: %.6f%n", overall);

System.out.printf("Абсолютный показатель: %.6f%n", P\_ij);

System.out.printf("Относительный показатель: %.6f%n", K\_ij);

System.out.printf("Фактор надежности: %.6f%n", K\_ij);

}

}

Класс Main содержит метод main, который инициализирует параметры, выполняет расчёты различных метрик надёжности и выводит результаты. Параметры включают количество зарегистрированных отказов, количество экспериментов, допустимые временные показатели и базовый критерий надёжности.

Результат работы программы представлен на рисунке 2.

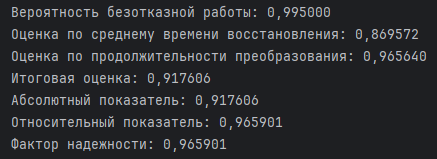


Рисунок 2 – Результат работы программы

# 4 Анализ результатов

В данной лабораторной работе была написана программа на языке Java, позволяющая производить расчеты оценки качества программных средств. По результатам расчетов, согласно данным по выбранному варианту, новое программное обеспечение хуже по качеству, чем базовое. Это является следствием того, что относительный показатель качества меньше единицы.