

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Сыров Игорь Анатольевич
Должность: Директор
Дата подписания: 22.08.2023 10:56:22
Уникальный программный ключ:
b683afe664d7e9f64175886cf9626a198149ad36

СТЕРЛИТАМАКСКИЙ ФИЛИАЛ
ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО ОБРАЗОВАТЕЛЬНОГО
УЧРЕЖДЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ
«УФИМСКИЙ УНИВЕРСИТЕТ НАУКИ И ТЕХНОЛОГИЙ»

Факультет Математики и информационных технологий
Кафедра Прикладной информатики и программирования

Оценочные материалы по дисциплине (модулю)

дисциплина **Оценка качества программного обеспечения**

Блок Б1, вариативная часть, Б1.В.ДВ.07.01

цикл дисциплины и его часть (базовая, вариативная, дисциплина по выбору)

Направление

10.03.01

Информационная безопасность

код

наименование направления

Программа

Безопасность компьютерных систем (по отрасли или в сфере профессиональной деятельности)

Форма обучения

Очная

Для поступивших на обучение в
2020 г.

Разработчик (составитель)
к.т.н., доцент
Антипин А. Ф.
ученая степень, должность, ФИО

1. Перечень компетенций с указанием этапов их формирования и описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания.....	3
2. Контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы	5
3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций	23

				программных средств.		
	3 этап: Владения (навыки / опыт деятельности)	Отсутствие умений.	Умеет выполнять измерение отдельных метрик программного обеспечения.	Умеет выполнять измерение метрик программного обеспечения по наиболее известным методикам, имеет небольшие трудности в оценке характеристик ПО.	Умеет выполнять оценку характеристик и измерение метрик программного обеспечения по наиболее известным методикам.	Лабораторные работы.

2. Контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Задания для контрольной работы

Описание контрольной работы:

Контрольные работы предназначены для оценки уровня сформированности компетенции **ПК-10** на этапе «Знания» и представляют собой письменное задание, выполняемое в течение 90 минут, и включает в себя ответы на 2 теоретических вопроса и решение практической задачи.

Перечень вопросов для контрольной работы №1:

1. Каковы основные понятия качества программного обеспечения (ПО)?
2. Что представляют собой характеристики программы?
3. Как определить понятие системы обеспечения качества?
4. Какие объекты программных систем подвержены уязвимости?
5. Какова классификация дестабилизирующих факторов ПО?
6. Какие существуют внутренние источники угроз ПО?
7. Какие внешние факторы могут дестабилизировать работу ПО?
8. Какие общие факторы влияют на качество ПО?
9. Как определить понятия фактора качества, критерия качества, метрики?
10. Что представляют собой понятия оценочного элемента, показателя качества, базового значения показателя качества?
11. Какой смысл вкладывается в понятия сертификации, верификации и аттестации?
12. Как взаимно связаны факторы, влияющие на качество ПО?
13. Каковы функциональные и конструктивные характеристики качества ПО?
14. Что такое внутреннее качество программных средств?
15. Как определить внешнее качество программных средств?
16. Что вкладывается в понятие качества при использовании ПО?
17. Как определить понятия системы измерений характеристик ПО, измерительной шкалы?
18. Каковы условия обеспечения объективности измерений?
19. Какие существуют категории групп показателей качества ПО?

20. Как классифицируются типы измерительных шкал?

Перечень вопросов для контрольной работы №2:

1. Каковы задачи и область применения программометрики?
2. Что такое алгоритмическая сложность? Как определить ее верхнюю оценку?
3. Какими свойствами обладает алгоритмическая сложность?
4. Что устанавливают законы Ципфа, Хика и Миллера?
5. Каковы измеряемые свойства программ?
6. Какие существуют классы несовершенств программных средств?
7. Что определяет соотношение Холстеда?
8. Какие элементы содержит словарь программы?
9. Как определить объем программы? Что такое потенциальный объем программы?
10. Что характеризует уровень реализации программы?
11. Как оптимизировать количество и длину модулей в программе?
12. Как количественно оценить работу программирования?
13. Как оценить уровень языка программирования?
14. Каков порядок расчета метрических характеристик программных средств?
15. Как рассчитать начальную надежность программы?
16. Какой смысл вкладывается в понятие структурной сложности программ?
17. Как определить сложность вычислительных маршрутов, маршрутов принятия логических решений и общую сложность программ?
18. Каково назначение метрики Маккейба?
19. Что характеризует связность модулей программных средств? Какие существуют типы связности программных модулей?
20. Каковы основные особенности модели надежности Джелински-Моранды, Миллса и Мусы?

Перечень задач для контрольной работы №1:

1. Разработать программу для вычисления значений функции $F(x, y)$. Значения аргументов функции вводятся с клавиатуры. В результате выполнения программы на экран выводится значение функции $F(x, y)$.

$$F(x, y) = \begin{cases} \ln|-x + 4 \sin^3 y|, & \text{при } x < -2; \\ \frac{4y^x - 1}{\sqrt{x^2 + |y| + 1}}, & \text{при } -2 \leq x < 2; \\ \sqrt{\ln x^2 + 4}, & \text{при } x \geq 2. \end{cases}$$

Определить значения метрик Холстеда и Джилба, на базе которых дать оценку качества разработанного исходного текста программы.

2. Разработать программу для вычисления значений функции $F(x, y)$. Значения аргументов функции вводятся с клавиатуры. В результате выполнения программы на экран выводится значение функции $F(x, y)$.

$$F(x, y) = \begin{cases} \frac{|4 + x^y|}{\sqrt{|x| + 5}}, & \text{при } x < 0; \\ (y + \cos x)^2 + 8, & \text{при } 0 \leq x \leq 3; \\ \frac{|\ln|y| - \ln x^2|}{\sqrt{x}}, & \text{при } x > 3. \end{cases}$$

Определить значения метрик Холстеда и Джилба, на базе которых дать оценку качества разработанного исходного текста программы.

3. Разработать программу для вычисления значений функции $F(x, y)$. Значения аргументов функции вводятся с клавиатуры. В результате выполнения программы на экран выводится значение функции $F(x, y)$.

$$F(x, y) = \begin{cases} \frac{\sqrt{y^2 + \ln x^2 + 1}}{x}, & \text{при } x \leq -2; \\ \frac{2y^x}{\sqrt{x^2 + |y| + 1}}, & \text{при } -2 < x \leq 0; \\ \frac{1}{x^y + 2} + \ln x, & \text{при } x > 0. \end{cases}$$

Определить значения метрик Холстеда и Джилба, на базе которых дать оценку качества разработанного исходного текста программы.

4. Разработать программу для вычисления значений функции $F(x, y)$. Значения аргументов функции вводятся с клавиатуры. В результате выполнения программы на экран выводится значение функции $F(x, y)$.

$$F(x, y) = \begin{cases} \frac{x^2 + \ln|y|}{\sqrt{-x + 5}}, & \text{при } x \leq -2; \\ \cos^2 y - \sin x^2, & \text{при } -2 < x < 5; \\ \frac{\sqrt{\ln x^2 + 8}}{y}, & \text{при } x \geq 5. \end{cases}$$

Определить значения метрик Холстеда и Джилба, на базе которых дать оценку качества разработанного исходного текста программы.

5. Разработать программу для вычисления значений функции $F(x, y)$. Значения аргументов функции вводятся с клавиатуры. В результате выполнения программы на экран выводится значение функции $F(x, y)$.

$$F(x, y) = \begin{cases} \frac{\sqrt{(y-1)^2 + \ln x^2}}{|x|+5} - 15, & \text{при } x < -4; \\ (y + \cos x)^2 + 2, & \text{при } -4 \leq x \leq 4; \\ \sqrt{\frac{x}{x^y + 2}} - \ln x + 10, & \text{при } x > 4. \end{cases}$$

Определить значения метрик Холстеда и Джилба, на базе которых дать оценку качества разработанного исходного текста программы.

6. Разработать программу для вычисления значений функции $F(x, y)$. Значения аргументов функции вводятся с клавиатуры. В результате выполнения программы на экран выводится значение функции $F(x, y)$.

$$F(x, y) = \begin{cases} \ln |\cos y - 2 \ln^2 |x||, & \text{при } x \leq -1; \\ \frac{\ln |x| + x^{-y}}{y^2 + x^{-2} + 3}, & \text{при } -1 < x < 5; \\ \sqrt{\ln x^2 + 2}, & \text{при } x \geq 5. \end{cases}$$

Определить значения метрик Холстеда и Джилба, на базе которых дать оценку качества разработанного исходного текста программы.

7. Разработать программу для вычисления значений функции $F(x, y)$. Значения аргументов функции вводятся с клавиатуры. В результате выполнения программы на экран выводится значение функции $F(x, y)$.

$$F(x, y) = \begin{cases} y^2 - \frac{\cos^2 x}{4}, & \text{при } x \leq -3; \\ \frac{2x}{|\ln |y| + 2|}, & \text{при } -3 < x \leq 3; \\ \frac{\sqrt{\ln x + 4}}{4}, & \text{при } x > 3. \end{cases}$$

Определить значения метрик Холстеда и Джилба, на базе которых дать оценку качества разработанного исходного текста программы.

8. Вывести на экран все натуральные шестизначные числа из диапазона от A до B , у которых совпадают сумма трех младших и трех старших цифр. При отсутствии чисел с указанными свойствами сформировать сообщение «Требуемых чисел нет». Границы диапазона A и B вводятся с клавиатуры.

9. Вывести на экран все натуральные числа из диапазона от A до B , в записи которых цифра 7 встречается ровно N раз. При отсутствии чисел с указанными свойствами сформировать сообщение «Требуемых чисел нет». Границы диапазона A и B и значение N вводятся с клавиатуры.

10. Вычислить и вывести на экран суммы K старших (т.е. находящихся слева) цифр натурального числа A . Число A и значение K вводятся с клавиатуры. Если количество цифр в числе меньше K , сформировать сообщение «Значение числа K слишком велико» и вывести его на экран.

Перечень задач для контрольной работы №2:

1. В результате тестирования программы серией из 29 тестов было обнаружено 3 ошибки: 3, 14 и 23 тестами. Требуется определить количество ошибок N в программе до начала тестирования, используя модель Джелински-Моранды.

2. В результате тестирования программы серией из 25 тестов было обнаружено 3 ошибки: 2, 10 и 20 тестами. Требуется определить количество ошибок N в программе до начала тестирования, используя модель Джелински-Моранды.

3. В результате тестирования программы серией из 33 тестов было обнаружено 3 ошибки: 5, 12 и 28 тестами. Требуется определить количество ошибок N в программе до начала тестирования, используя модель Джелински-Моранды.

4. В результате тестирования программы серией из 30 тестов было обнаружено 3 ошибки: 4, 17 и 21 тестами. Требуется определить количество ошибок N в программе до начала тестирования, используя модель Джелински-Моранды.

5. В результате тестирования программы серией из 27 тестов было обнаружено 3 ошибки: 8, 11 и 26 тестами. Требуется определить количество ошибок N в программе до начала тестирования, используя модель Джелински-Моранды.

6. В результате тестирования программы двумя независимыми группами: первой группой обнаружено 35 ошибок, а второй группой – 23 ошибки. 15 ошибок, обнаруженных первой группой совпадает с ошибками, обнаруженными второй группой.

Требуется оценить количество неустраненных ошибок, которые остались в программе после тестирования.

7. В результате тестирования программы двумя независимыми группами: первой группой обнаружено 38 ошибок, а второй группой – 24 ошибки. 11 ошибок, обнаруженных первой группой совпадает с ошибками, обнаруженными второй группой.

Требуется оценить количество неустраненных ошибок, которые остались в программе после тестирования.

8. В результате тестирования программы двумя независимыми группами: первой группой обнаружено 25 ошибок, а второй группой – 22 ошибки. 17 ошибок, обнаруженных первой группой совпадает с ошибками, обнаруженными второй группой.

Требуется оценить количество неустраненных ошибок, которые остались в программе после тестирования.

9. В результате тестирования программы двумя независимыми группами: первой группой обнаружено 40 ошибок, а второй группой – 32 ошибки. 25 ошибок, обнаруженных первой группой совпадает с ошибками, обнаруженными второй группой.

Требуется оценить количество неустранимых ошибок, которые остались в программе после тестирования.

10. В результате тестирования программы двумя независимыми группами: первой группой обнаружено 25 ошибок, а второй группой – 33 ошибки. 16 ошибок, обнаруженных первой группой совпадает с ошибками, обнаруженными второй группой.

Требуется оценить количество неустранимых ошибок, которые остались в программе после тестирования.

Описание методики оценивания:

Контрольная работа проводится с целью обеспечения своевременной обратной связи, для коррекции обучения, активизации самостоятельной работы студентов. Критериями оптимального усвоения знаний при проведении контрольной работы являются объем, системность, осмысленность, прочность и действенность знаний обучающихся.

Результаты контрольной работы оцениваются в соответствии с рейтинг-планом дисциплины.

Критерии оценки (в баллах):

- ***14-15 баллов*** выставляется студенту, если студент дал полные, развернутые ответы на все теоретические вопросы, продемонстрировал знание функциональных возможностей, терминологии, основных элементов, умение применять теоретические знания при выполнении практических заданий. Практическое задание работы выполнено полностью, без неточностей и ошибок;

- ***10-13 баллов*** выставляется студенту, если студент раскрыл в основном теоретические вопросы, однако допущены неточности в определении основных понятий. При выполнении практической части работы допущены несущественные ошибки;

- ***6-9 баллов*** выставляется студенту, если при ответе на теоретические вопросы студентом допущено несколько существенных ошибок в толковании основных понятий. Логика и полнота ответа страдают заметными изъянами. Заметны пробелы в знании основных методов. Теоретические вопросы в целом изложены достаточно, но с пропусками материала. Имеются принципиальные ошибки в логике построения ответа на вопрос. Студентом при решении допущены грубые ошибки;

- ***0-5 баллов*** выставляется студенту, если он не смог ответить на вопросы, ответ на теоретические вопросы свидетельствует о непонимании и крайне неполном знании основных понятий и методов. Обнаруживается отсутствие навыков применения теоретических знаний при выполнении практических заданий.

Задания для аудиторной работы, в т.ч. решения у доски

Задания для аудиторной работы, в т.ч. решения у доски, предназначены для оценки уровня сформированности компетенции ПК-10 на этапе «Владения (навыки / опыт деятельности)».

Перечень заданий для аудиторной работы:

1. Разработать программу для вычисления значений функции $F(x, y)$. Значения аргументов функции вводятся с клавиатуры. В результате выполнения программы на экран выводится значение функции $F(x, y)$.

$$F(x, y) = \begin{cases} \ln|-x + 4 \sin^3 y|, & \text{при } x < -2; \\ \frac{4y^x - 1}{\sqrt{x^2 + |y| + 1}}, & \text{при } -2 \leq x < 2; \\ \sqrt{\ln x^2 + 4}, & \text{при } x \geq 2. \end{cases}$$

Определить значения метрик Холстеда и Джилба, на базе которых дать оценку качества разработанного исходного текста программы.

2. Разработать программу для вычисления значений функции $F(x, y)$. Значения аргументов функции вводятся с клавиатуры. В результате выполнения программы на экран выводится значение функции $F(x, y)$.

$$F(x, y) = \begin{cases} \frac{|4 + x^y|}{\sqrt{|x| + 5}}, & \text{при } x < 0; \\ (y + \cos x)^2 + 8, & \text{при } 0 \leq x \leq 3; \\ \frac{|\ln|y| - \ln x^2|}{\sqrt{x}}, & \text{при } x > 3. \end{cases}$$

Определить значения метрик Холстеда и Джилба, на базе которых дать оценку качества разработанного исходного текста программы.

3. Разработать программу для вычисления значений функции $F(x, y)$. Значения аргументов функции вводятся с клавиатуры. В результате выполнения программы на экран выводится значение функции $F(x, y)$.

$$F(x, y) = \begin{cases} \frac{\sqrt{y^2 + \ln x^2 + 1}}{x}, & \text{при } x \leq -2; \\ \frac{2y^x}{\sqrt{x^2 + |y| + 1}}, & \text{при } -2 < x \leq 0; \\ \frac{1}{x^y + 2} + \ln x, & \text{при } x > 0. \end{cases}$$

Определить значения метрик Холстеда и Джилба, на базе которых дать оценку качества разработанного исходного текста программы.

4. Вывести на экран все натуральные шестизначные числа из диапазона от A до B , у которых совпадают сумма трех младших и трех старших цифр. При отсутствии чисел с указанными свойствами сформировать сообщение «Требуемых чисел нет». Границы диапазона A и B вводятся с клавиатуры.

5. Вывести на экран все натуральные числа из диапазона от A до B , в записи которых цифра 7 встречается ровно N раз. При отсутствии чисел с указанными свойствами сформировать сообщение «Требуемых чисел нет». Границы диапазона A и B и значение N вводятся с клавиатуры.

6. В результате тестирования программы серией из 29 тестов было обнаружено 3 ошибки: 3, 14 и 23 тестами. Требуется определить количество ошибок N в программе до начала тестирования, используя модель Джелински-Моранды.

7. В результате тестирования программы серией из 25 тестов было обнаружено 3 ошибки: 2, 10 и 20 тестами. Требуется определить количество ошибок N в программе до начала тестирования, используя модель Джелински-Моранды.

8. В результате тестирования программы двумя независимыми группами: первой группой обнаружено 25 ошибок, а второй группой – 22 ошибки. 17 ошибок, обнаруженных первой группой совпадает с ошибками, обнаруженными второй группой.

Требуется оценить количество неустраненных ошибок, которые остались в программе после тестирования.

9. В результате тестирования программы двумя независимыми группами: первой группой обнаружено 40 ошибок, а второй группой – 32 ошибки. 25 ошибок, обнаруженных первой группой совпадает с ошибками, обнаруженными второй группой.

Требуется оценить количество неустраненных ошибок, которые остались в программе после тестирования.

10. В результате тестирования программы двумя независимыми группами: первой группой обнаружено 25 ошибок, а второй группой – 33 ошибки. 16 ошибок, обнаруженных первой группой совпадает с ошибками, обнаруженными второй группой.

Требуется оценить количество неустраненных ошибок, которые остались в программе после тестирования.

Критерии оценки (в баллах):

- **3 балла** выставляются студенту, если задание выполнено полностью, без неточностей и ошибок;

- **2 балла** выставляются студенту, если при выполнении задания были допущены несущественные ошибки;

- **1 балл** выставляется студенту, если при выполнении задания допущены грубые ошибки, задание выполнено не полностью.

Задания для лабораторных работ

Лабораторные работы предназначены для оценки уровня сформированности компетенции **ПК-10** на этапе «Умения».

Лабораторная работа №1

Задания:

1. Разработать программу для вычисления значений функции $F(x, y)$. Значения аргументов функции вводятся с клавиатуры. В результате выполнения программы на экран выводится значение функции $F(x, y)$.

2. Определить значения метрик Холстеда и Джилба, на базе которых дать оценку качества разработанного исходного текста программы.

Вариант 1

$$F(x, y) = \begin{cases} \ln|-x + 2 \sin^2 y|, & \text{при } x < 0; \\ \frac{\cos x + 2^{-y}}{x + 5}, & \text{при } 0 \leq x \leq 5; \\ \sqrt{\ln x + 4}, & \text{при } x > 5. \end{cases}$$

Вариант 2

$$F(x, y) = \begin{cases} \frac{\sqrt{(y-1)^2}}{x} + \ln x^2, & \text{при } x < -2; \\ \frac{2x}{|\ln|y|+2|}, & \text{при } -2 \leq x < 0; \\ \frac{\sqrt{\cos x^2}}{x^y + 7} + y, & \text{при } x \geq 0. \end{cases}$$

Вариант 3

$$F(x, y) = \begin{cases} y^2 + \frac{\cos x}{3}, & \text{при } x \leq -10; \\ \cos^2 y + \sin x^2, & \text{при } -10 < x \leq 10; \\ \sqrt{\frac{y^4 + |7x-14|}{x}}, & \text{при } x > 10. \end{cases}$$

Вариант 4

$$F(x, y) = \begin{cases} \frac{x^2 + \ln|y|}{\sqrt{-x+10}}, & \text{при } x < -5; \\ y + \cos^3 x, & \text{при } -5 \leq x \leq 3; \\ \frac{|\ln|y| - \ln x^2|}{x}, & \text{при } x > 3. \end{cases}$$

Лабораторная работа №2

Задание: Выполнить оценку алгоритмической сложности программы по следующим шагам:

- разработать алгоритм решения задачи (по вариантам);
- построить граф потока управления;
- сформировать маршруты тестирования в соответствии с критериями;
- найти значение цикломатического числа, характеризующего структурную сложность программ;
- сформировать матрицы смежности и достижимости;
- провести анализ полученных результатов.

Вариант №1:

Вывести на экран все натуральные числа из диапазона от A до B , сумма цифр которых равна S . При отсутствии чисел с указанными свойствами сформировать сообщение «Требуемых чисел нет». Границы диапазона A и B и заданная сумма цифр S вводятся с клавиатуры.

Вариант №2:

Вывести на экран все натуральные трехзначные и пятизначные числа из диапазона от A до B , значение которых кратно 13. При отсутствии чисел с указанными свойствами сформировать сообщение «Требуемых чисел нет». Границы диапазона A и B вводятся с клавиатуры.

Вариант №3:

Вывести на экран все натуральные числа из диапазона от A до B , у которых совпадают старшая и младшая цифры. При отсутствии чисел с указанными свойствами сформировать сообщение «Требуемых чисел нет». Границы диапазона A и B вводятся с клавиатуры.

Вариант №4:

Вывести на экран для всех натуральных чисел из диапазона от A до B сами числа и суммы их цифр, которые находятся на нечетных позициях. Номера позиций отсчитываются с единицы, начиная с младшей цифры. Границы диапазона A и B вводятся с клавиатуры.

Лабораторная работа №3

Задание: Разработать программу, реализующую решение по приведенному условию, а затем оценить характеристики разработанной программы на основе применения объектно-ориентированных метрик Абреу.

Вариант №1:

Разработать приложение для вычисления результирующей информации об объектах, описанных с помощью наследования:

- 1) базовый объект – «Нефтехранилище» (поле: объем в тоннах – $vBase$);
- 2) производный объект 1 – «Перевозка танкером» с полями: вместимость в тоннах – $vTanker$;
- 3) производный объект 2 – «Перевозка железнодорожным составом» с полями: емкость цистерны в тоннах – $vCist$, количество цистерн в составе – $nCist$.

Для решения задачи необходимо:

- 1) определить базовый класс и производные классы, используя наследование;
- 2) используя виртуальный метод печати базового класса, разработать переопределенные методы производных классов для вывода результирующей информации: тип перевозки и количество танкеров, необходимых для вывоза всей нефти из хранилища, или количество железнодорожных составов;
- 3) создать массив для хранения ссылок на объекты, следующие в произвольном порядке. Всю необходимую информацию вводит пользователь;
- 4) создать объекты, присвоив начальные значения полям объектов с помощью конструктора;
- 5) используя массив ссылок, вывести в цикле информацию.

Вариант №2:

Разработать приложение для вычисления результирующей информации об объектах, описанных с помощью наследования:

- 1) базовый объект – «Бензобак» (поле: емкость бензобака – vB);
- 2) производный объект 1 – «Спортивный автомобиль» с полями: тип автотранспортного средства – $type$, средний расход бензина на 1 км пробега – rb , стоимость одного литра бензина – slb ;
- 3) производный объект 2 – «Автобус» с полями: тип автотранспортного средства – $type$, количество бензобаков – kb , средний расход бензина на 1 км пробега – rb , стоимость одного литра бензина – slb .

Для решения задачи необходимо:

- 1) определить базовый класс и производные классы, используя наследование;
- 2) используя виртуальный метод печати базового класса, разработать переопределенные методы производных классов для вывода результирующей информации: тип автотранспортного средства, средний пробег автотранспортного средства при полных бензобаках без дозаправки, стоимость заправки бензобаков;
- 3) создать массив для хранения ссылок на объекты, следующие в произвольном порядке. Всю необходимую информацию вводит пользователь;
- 4) создать объекты, присвоив начальные значения полям объектов с помощью конструктора;
- 5) используя массив ссылок и цикл, вывести на экран информацию.

Вариант №3:

Разработать приложение для вычисления результирующей информации об объектах, описанных с помощью наследования:

- 1) базовый объект – «Топливный бак» (поле: емкость бака – vB);
- 2) производный объект 1 – «Самолет» с полями: тип самолета – $typS$, средний расход керосина на 1 км пути – rka , предполагаемое расстояние перелета – km , тип топлива – tt ;
- 3) производный объект 1 – «Вертолет» с полями: тип вертолета – $typV$, средний расход бензина на 1 км пути – rkv , предполагаемое расстояние перелета – km , тип топлива – tt .

Для решения задачи необходимо:

- 1) определить базовый класс и производные классы, используя наследование;
- 2) используя виртуальный метод печати базового класса, разработать переопределенные методы производных классов для вывода результирующей информации: тип летательного аппарата, сведения о возможности (невозможности) перелета на одной заправке на заданное расстояние и количество дозаправок;
- 3) создать массив для хранения ссылок на объекты, следующие в произвольном порядке. Всю необходимую информацию вводит пользователь;
- 4) создать объекты, присвоив начальные значения полям объектов с помощью конструктора;
- 5) используя массив ссылок и цикл, вывести на экран информацию.

Вариант №4:

Разработать приложение для вычисления результирующей информации об объектах, описанных с помощью наследования:

- 1) базовый объект – «Тариф страхования на год» (поле: тариф – $tarif$);
- 2) производный объект 1 – «Сумма страховки начинающего водителя» с полями: фамилия – $name$, срок страхования – $time$, повышающий коэффициент – $koefup$;
- 3) производный объект 2 – «Сумма страховки опытного водителя» с полями: фамилия – $name$, срок страхования – $time$, понижающий коэффициент – $koefd$.

Для решения задачи необходимо:

- 1) определить базовый класс и производные классы, используя наследование;
- 2) используя виртуальный метод печати базового класса, разработать переопределенные методы производных классов для вывода результирующей информации: фамилия водителя, сумма страховой премии за заданный период;

3) создать массив для хранения ссылок на объекты, следующие в произвольном порядке. Всю необходимую информацию вводит пользователь;

4) создать объекты, присвоив начальные значения полям объектов с помощью конструктора;

5) используя массив ссылок, вывести в цикле информацию.

Лабораторная работа №4

Вариант №1:

1. В результате тестирования программы серией из 22 тестов было обнаружено 3 ошибки: 1, 10 и 21 тестами. Требуется определить количество ошибок N в программе до начала тестирования, используя модель Джелински-Моранды.

2. В программу было преднамеренно внесено (посеяно) 20 ошибок. Предположим, что в программе перед началом тестирования было 28 ошибок. В процессе семи тестовых прогонов было выявлено следующее количество ошибок:

Номер прогона	1	2	3	4	5	6	7
V	6	4	4	2	2	1	1
S	3	3	2	2	1	1	1

Необходимо оценить количество ошибок N перед каждым тестовым прогоном и степень отлаженности программы после каждого прогона, используя модель Миллса. Построить график зависимости числа ошибок N в программе до начала тестирования от номера тестового прогона. Проанализировать динамику отлаженности программы S , для чего построить диаграмму и оценить тенденцию изменения этого показателя.

3. В результате тестирования программы двумя независимыми группами: первой группой обнаружено 37 ошибок, а второй группой – 22 ошибки. 13 ошибок, обнаруженных первой группой совпадает с ошибками, обнаруженными второй группой.

Требуется оценить количество неустранимых ошибок, которые остались в программе после тестирования.

Вариант №2:

1. В результате тестирования программы серией из 35 тестов было обнаружено 3 ошибки: 3, 16 и 34 тестами. Требуется определить количество ошибок N в программе до начала тестирования, используя модель Джелински-Моранды.

2. В программу было преднамеренно внесено (посеяно) 28 ошибок. Предположим, что в программе перед началом тестирования было 38 ошибок. В процессе семи тестовых прогонов было выявлено следующее количество ошибок:

Номер прогона	1	2	3	4	5	6	7
---------------	---	---	---	---	---	---	---

V	8	6	5	4	2	2	1
S	5	4	3	2	2	1	1

Необходимо оценить количество ошибок N перед каждым тестовым прогоном и степень отлаженности программы после каждого прогона, используя модель Миллса. Построить график зависимости числа ошибок N в программе до начала тестирования от номера тестового прогона. Проанализировать динамику отлаженности программы S , для чего построить диаграмму и оценить тенденцию изменения этого показателя.

3. В результате тестирования программы двумя независимыми группами: первой группой обнаружено 39 ошибок, а второй группой – 28 ошибок. 16 ошибок, обнаруженных первой группой совпадает с ошибками, обнаруженными второй группой.

Требуется оценить количество неустранимых ошибок, которые остались в программе после тестирования.

Вариант №3:

1. В результате тестирования программы серией из 30 тестов было обнаружено 3 ошибки: 5, 14 и 30 тестами. Требуется определить количество ошибок N в программе до начала тестирования, используя модель Джелински-Моранды.

2. В программу было преднамеренно внесено (посеяно) 32 ошибки. Предположим, что в программе перед началом тестирования было 35 ошибок. В процессе семи тестовых прогонов было выявлено следующее количество ошибок:

Номер прогона	1	2	3	4	5	6	7
V	8	6	7	4	3	2	2
S	5	3	3	4	3	1	1

Необходимо оценить количество ошибок N перед каждым тестовым прогоном и степень отлаженности программы после каждого прогона, используя модель Миллса. Построить график зависимости числа ошибок N в программе до начала тестирования от номера тестового прогона. Проанализировать динамику отлаженности программы S , для чего построить диаграмму и оценить тенденцию изменения этого показателя.

3. В результате тестирования программы двумя независимыми группами: первой группой обнаружено 42 ошибки, а второй группой – 37 ошибок. 25 ошибок, обнаруженных первой группой совпадает с ошибками, обнаруженными второй группой.

Требуется оценить количество неустранимых ошибок, которые остались в программе после тестирования.

Вариант №4:

1. В результате тестирования программы серией из 30 тестов было обнаружено 3 ошибки: 2, 11 и 29 тестами. Требуется определить количество ошибок N в программе до начала тестирования, используя модель Джелински-Моранды.

2. В программу было преднамеренно внесено (посеяно) 32 ошибки. Предположим, что в программе перед началом тестирования было 38 ошибок. В процессе семи тестовых прогонов было выявлено следующее количество ошибок:

Номер прогона	1	2	3	4	5	6	7
V	7	7	5	5	4	2	2
S	5	5	3	5	3	3	1

Необходимо оценить количество ошибок N перед каждым тестовым прогоном и степень отлаженности программы после каждого прогона, используя модель Миллса. Построить график зависимости числа ошибок N в программе до начала тестирования от номера тестового прогона. Проанализировать динамику отлаженности программы S , для чего построить диаграмму и оценить тенденцию изменения этого показателя.

3. В результате тестирования программы двумя независимыми группами: первой группой обнаружено 16 ошибок, а второй группой – 19 ошибок. 10 ошибок, обнаруженных первой группой совпадает с ошибками, обнаруженными второй группой.

Требуется оценить количество неустранимых ошибок, которые остались в программе после тестирования.

Описание методики оценивания:

Результаты лабораторных работ оцениваются в соответствии с рейтинг-планом дисциплины.

Критерии оценки (в баллах):

- **7-8 баллов** выставляется студенту, если все задания работы выполнены полностью, без неточностей и ошибок;

- **5-6 баллов** выставляется студенту, если при выполнении работы были допущены незначительные ошибки;

- **3-4 баллов** выставляется студенту, если при выполнении работы допущены грубые ошибки, задания выполнены не полностью;

- **1-2 баллов** выставляется студенту, если он не смог выполнить задания работы, но продемонстрировал попытки ее решения.

Вопросы к зачету

Перечень вопросов для зачета:

1. Каковы основные понятия качества программного обеспечения (ПО)?
2. Что представляют собой характеристики программы?
3. Как определить понятие системы обеспечения качества?
4. Какие объекты программных систем подвержены уязвимости?
5. Какова классификация дестабилизирующих факторов ПО?
6. Какие существуют внутренние источники угроз ПО?
7. Какие внешние факторы могут дестабилизировать работу ПО?
8. Какие общие факторы влияют на качество ПО?
9. Как определить понятия фактора качества, критерия качества, метрики?
10. Что представляют собой понятия оценочного элемента, показателя качества, базового значения показателя качества?
11. Какой смысл вкладывается в понятия сертификации, верификации и аттестации?
12. Как взаимосвязаны факторы, влияющие на качество ПО?
13. Каковы функциональные и конструктивные характеристики качества ПО?
14. Что такое внутреннее качество программных средств?
15. Как определить внешнее качество программных средств?
16. Что вкладывается в понятие качества при использовании ПО?
17. Как определить понятия системы измерений характеристик ПО, измерительной шкалы?
18. Каковы условия обеспечения объективности измерений?
19. Какие существуют категории групп показателей качества ПО?
20. Как классифицируются типы измерительных шкал?
21. Каковы задачи и область применения программометрики?
22. Что такое алгоритмическая сложность? Как определить ее верхнюю оценку?
23. Какими свойствами обладает алгоритмическая сложность?
24. Что устанавливают законы Ципфа, Хика и Миллера?
25. Каковы измеряемые свойства программ?
26. Какие существуют классы несовершенств программных средств?
27. Что определяет соотношение Холстеда?
28. Какие элементы содержит словарь программы?

29. Как определить объем программы? Что такое потенциальный объем программы?
30. Что характеризует уровень реализации программы?
31. Как оптимизировать количество и длину модулей в программе?
32. Как количественно оценить работу программирования?
33. Как оценить уровень языка программирования?
34. Каков порядок расчета метрических характеристик программных средств?
35. Как рассчитать начальную надежность программы?
36. Какой смысл вкладывается в понятие структурной сложности программ?
37. Как определить сложность вычислительных маршрутов, маршрутов принятия логических решений и общую сложность программ?
38. Каково назначение метрики Маккейба?
39. Что характеризует связность модулей программных средств? Какие существуют типы связности программных модулей?
40. Каковы основные особенности модели надежности Джелински-Моранды, Миллса и Мусы?

Описание методики оценивания:

Вопросы к зачету используются при защите лабораторных работ студентов, как уточняющие вопросы по теме задания.

Критерии оценки (в баллах):

- **5 баллов** выставляются студенту, если он дал полные и развернутые ответы на все вопросы преподавателя;
- **3-4 баллов** выставляются студенту, если при ответе на вопросы допущены небольшие неточности;
- **1-2 баллов** выставляются студенту, если при ответе на вопросы допущены существенные ошибки в толковании основных понятий.

3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Рейтинг-план дисциплины

Виды учебной деятельности студентов	Балл за конкретное задание	Число заданий за семестр	Баллы	
			Минимальный	Максимальный
Модуль 1				
Текущий контроль			0	25
1. Аудиторная работа, в т.ч. решение задач у доски	3	3	0	9
2. Лабораторные работы №1, 2	8	2	0	16
Рубежный контроль			0	25
1. Защита лабораторных работ	5	2	0	10
2. Контрольная работа №1	15	1	0	15
Модуль 2				
Текущий контроль			0	25
1. Аудиторная работа, в т.ч. решение задач у доски	3	3	0	9
2. Лабораторные работы №3, 4	8	2	0	16
Рубежный контроль			0	25
1. Защита лабораторных работ	5	2	0	10
2. Контрольная работа №2	15	1	0	15
Поощрительные баллы				
1. Участие в студенческих олимпиадах, публикация статей и пр.			0	10
Посещаемость (баллы вычитаются из общей суммы набранных баллов)				
1. Посещение лекционных занятий			0	-6
2. Посещение практических (семинарских, лабораторных занятий)			0	-10

Результаты обучения по дисциплине (модулю) у обучающихся оцениваются по итогам текущего контроля количественной оценкой, выраженной в рейтинговых баллах. Оценке подлежит каждое контрольное мероприятие.

При оценивании сформированности компетенций применяется четырехуровневая

шкала «неудовлетворительно», «удовлетворительно», «хорошо», «отлично».

Максимальный балл по каждому виду оценочного средства определяется в рейтинг-плане и выражает полное (100%) освоение компетенции.

Уровень сформированности компетенции «хорошо» устанавливается в случае, когда объем выполненных заданий соответствующего оценочного средства составляет 80-100%; «удовлетворительно» – выполнено 40-80%; «неудовлетворительно» – выполнено 0-40%

Рейтинговый балл за выполнение части или полного объема заданий соответствующего оценочного средства выставляется по формуле:

Рейтинговый балл = $k \times$ Максимальный балл,

где $k = 0,2$ при уровне освоения «неудовлетворительно», $k = 0,4$ при уровне освоения «удовлетворительно», $k = 0,8$ при уровне освоения «хорошо» и $k = 1$ при уровне освоения «отлично».

Оценка на этапе промежуточной аттестации выставляется согласно Положению о модульно-рейтинговой системе обучения и оценки успеваемости студентов УУНиТ:

На зачете выставляется оценка:

- зачтено - при накоплении от 60 до 110 рейтинговых баллов (включая 10 поощрительных баллов),
- не зачтено - при накоплении от 0 до 59 рейтинговых баллов.

При получении на экзамене оценок «отлично», «хорошо», «удовлетворительно», на зачёте оценки «зачтено» считается, что результаты обучения по дисциплине (модулю) достигнуты и компетенции на этапе изучения дисциплины (модуля) сформированы.