

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Сыров Игорь Анатольевич
Должность: Директор
Дата подписания: 2024.11.09 10:20
Уникальный программный ключ:
b683afe664d7e9f64175886cf9626a199149ad34

ПРИЛОЖЕНИЕ 2

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

Стерлитамакский филиал

Колледж

Фонд оценочных средств

по профессиональному **ПМ.03 Ревьюирование программных модулей**
модулю

цикл дисциплины и его часть (обязательная, вариативная)

специальность

09.02.07

Информационные системы и программирование

код

наименование специальности

Квалификация

Специалист по информационным системам

Разработчик (составитель)

Игдисамова Р.Р.

ученая степень, ученое звание, катего-
рия, Ф.И.О.

Стерлитамак 2024

I. Общие положения

1. Фонды оценочных средств предназначены для проверки результатов освоения вида профессиональной деятельности (ВПД) ПМ.03 Ревьюирование программных модулей и составляющих его профессиональных и общих компетенций, программы подготовки специалистов среднего звена по специальности 09.02.07 Информационные системы и программирование.

Формой аттестации по профессиональному модулю является экзамен по модулю.

Форма проведения экзамена –устный опрос и выполнение задания по билетам.

2. Формы контроля и оценивания элементов профессионального модуля

Таблица 1.1.

Элемент моду-	Форма контроля и оценивания	
	Промежуточная аттестация	Текущий контроль
МДК.03.01 «Моделирование и анализ программного обеспечения»	Дифференцированный зачет (4 семестр), экзамен по модулю.	Наблюдение за выполнением практических работ. Контроль результата выполнения практических работ. Устный опрос на практических занятиях. Самостоятельная работа. Тестирование.
МДК.03.02 «Управление проектами»	Дифференцированный зачет (4 семестр), экзамен по модулю.	Наблюдение за выполнением практических работ. Контроль результата выполнения практических работ. Устный опрос на практических занятиях. Самостоятельная работа. Тестирование.
УП	Дифференцированный зачет (4 семестр).	Наблюдение за выполнением работ на учебной практике.
ПП	Дифференцированный зачет (4 семестр).	Наблюдение за выполнением работ на производственной практике.

Далее размещаются ФОС по каждому МДК, УП и ПП.

3. Результаты освоения профессионального модуля, подлежащие проверке

В результате аттестации по профессиональному модулю комплексная проверка общих и профессиональных компетенций профессионального модуля осуществляется в форме оценки качества выполнения заданий на экзамене по модулю/квалификационном экзамене:

Планируемые результаты освоения образовательной программы	Этап	Показатели и критерии оценивания результатов обучения				Вид оценочного средства
1.	2.	3.				4.
		неуд.	удовл.	хорошо	отлично	
ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам.	1 этап: Знания основных методы и средства эффективного анализа функционирования программного обеспечения; основные виды работ на этапе сопровождения программного обеспечения; основные принципы контроля конфигурации и поддержки целостности конфигурации программного обеспечения; средства защиты программного обеспечения в	незнание и неумение применять методы решения профессиональных задач.	необоснованный выбор методов и способов решения профессиональных задач; несвоевременность выполнения профессиональных заданий.	незначительные затруднения в выборе и применении методов и способов решения профессиональных задач; несвоевременность выполнения профессиональных заданий.	обоснованность выбора и применения методов и способов решения профессиональных задач; своевременность выполнения профессиональных заданий.	Тестовые задания Решение ситуационных задач.

	компьютер- ных систе- мах.					
	2 этап: Умения подбирать и настраивать конфигура- цию про- граммного обеспечения компьютер- ных систем; использо- вать методы защиты про- граммного обеспечения компьютер- ных систем; проводить инсталля- цию про- граммного обеспечения компьютер- ных систем; производить настройку отдельных компонен- тов про- граммного обеспечения компьютер- ных систем; анализиро- вать риски и характери- стики каче- ства про- граммного обеспечения					
	3 этап: Иметь прак- тический опыт в на- стройке от- дельных компонен- тов ПО ком- пьютерных сетей, вы-					

	полнять отдельные виды работ на этапе поддержки ПО компьютерных систем.					
<i>ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационных технологий для выполнения задач профессиональной деятельности.</i>	1 этап: Знание основных методов и средства эффективного анализа функционирования программного обеспечения; основные виды работ на этапе сопровождения программного обеспечения; основные принципы контроля конфигурации и поддержки целостности конфигурации программного обеспечения; средства защиты программного обеспечения в компьютерных системах.	отсутствие навыков использования источниками информации для анализа и оценки вариантов решения профессиональных задач.	значительные затруднения в обосновании выбора источников информации для анализа и оценки вариантов решения профессиональных задач.	незначительные затруднения в обосновании выбора источников информации для анализа и оценки вариантов решения профессиональных задач.	обоснованность выбора источников информации для анализа и оценки вариантов решения профессиональных задач и задач профессионального и личностного развития.	<i>Устный опрос Тестовые задания Решение ситуационных задач.</i>
	2 этап: Умение подбирать и настраивать конфигурацию программного обеспечения компьютер-					

	ных систем; использовать методы защиты программного обеспечения компьютерных систем; проводить инсталляцию программного обеспечения компьютерных систем; производить настройку отдельных компонентов программного обеспечения компьютерных систем; анализировать риски и характеристики качества программного обеспечения. 3 этап: Иметь практический опыт в настройке отдельных компонентов ПО компьютерных сетей, выполнять отдельные виды работ на этапе поддержки ПО компьютерных систем.					
ОК 09. Пользоваться профессией	1 этап: Знания правил построения простых и	Незнание и неумение понимать об-	Значительные затруднения в по-	Незначительные затруднения в на-	Обоснованность выбора и применение	Устный опрос Тестовые за-

нальной документации на государственном и иностранном языках.	сложных предложений на профессиональные темы; основные общеупотребительные глаголы.	ший смысл четко произнесенных высказываний на известные темы.	нимании общего смысла четко произнесенных высказываний на известные темы, понимании текстов на базовые профессиональные темы.	писании простых связанных предложений на знакомые или интересующие темы; применении лексического минимума, относящегося к правилам чтения текстов профессиональной направленности.	ния лексического минимума, относящегося к правилам чтения текстов профессиональной направленности.	дания <i>Решение ситуационных задач.</i>
	2 этап: Умения понимать общий смысл четко произнесенных высказываний на известные темы, понимать тексты на базовые профессиональные темы; участвовать в диалогах на знакомые общие и профессиональные темы; писать простые связанные сообщения на знакомые или интересующие профессиональные темы.					
	3 этап: Иметь практический опыт писать простые связанные предложения на знакомые или интересующие темы; применения лексическо-					

	го миниму- ма, относя- щегося к правилам чтения тек- стов проф- фессиональ- ной направ- ленности.					
ПК 3.1. Осуществ- лять ревъ- юирование программ- ного кода в соответст- вии с тех- нической документа- цией.	1 этап: Зна- ние основ- ных методов и средств эффективно- го анализа функциони- рования программ- ного обеспе- чения; ос- новные ви- ды работ на этапе сопро- вождения программ- ного обеспе- чения; ос- новные принципы контроля конфигура- ции и под- держки це- лостности конфигура- ции про- граммного обеспече- ния; средст- ва защиты программ- ного обеспе- чения в компьютер- ных систе- мах.	Совсем не умеет под- готавли- вать обо- рудования к работе, плохо зна- ет обра- ботку ста- тистиче- ского и динамиче- ского кон- тента.	Не умеет подгото- вить обо- рудования к работе, нормально освоил знание об- работку статисти- ческого и динамиче- ского кон- тента.	Частично освоил на- выки под- готовки оборудо- вания к работе, хорошо знает об- работку статисти- ческого и динамиче- ского кон- тента.	Овладел навыками подготов- ки обору- дования к работе, от- лично ос- воил обра- ботку ста- тистиче- ского и динамиче- ского кон- тента.	<i>Устный опрос Тесто- вые за- дания Реше- ние си- туаци- онных задач</i>
	2 этап: Умения подбирать и настраивать конфигура- цию про- граммного					

	обеспечения компьютерных систем; использовать методы защиты программного обеспечения компьютерных систем; проводить инсталляцию программного обеспечения компьютерных систем; производить настройку отдельных компонентов программного обеспечения компьютерных систем; анализировать риски и характеристики качества программного обеспечения. 3 этап: Иметь практический опыт в настройке отдельных компонентов ПО компьютерных сетей, выполнять отдельные виды работ на этапе поддержки ПО компьютерных систем.					
--	--	--	--	--	--	--

<i>ПК</i> 3.2. Выполнять процесс измерения характеристик компонент программного продукта для определения соответствия заданным критериям.	1 этап: Знания основные методы и средства эффективно-го анализа функционирования программного обеспечения; основные виды работ на этапе сопровождения программного обеспечения; основные принципы контроля конфигурации и поддержки целостности конфигурации программного обеспечения; средства защиты программного обеспечения в компьютерных системах.	Не знает и не умеет разрабатывать технические документы для управления информационными ресурсами.	Частично знает как разрабатывать технические документы для управления информационными ресурсами, но не умеет использовать навыки на практике.	Знает как разрабатывать технические документы для управления информационными ресурсами, но не умеет использовать навыки на практике.	Знает и умеет разрабатывать технические документы для управления информационными ресурсами.	<i>Устный опрос</i> <i>Тестовые задания</i> <i>Решение ситуационных задач.</i>
	2 этап: Умения подбирать и настраивать конфигурацию программного обеспечения компьютерных систем; использовать методы защиты программного обеспечения компьютер-					

	ных систем; проводить инсталляцию программного обеспечения компьютерных систем; производить настройку отдельных компонентов программного обеспечения компьютерных систем; анализировать риски и характеристики качества программного обеспечения. 3 этап: Иметь практический опыт в настройке отдельных компонентов ПО компьютерных сетей, выполнять отдельные виды работ на этапе поддержки ПО компьютерных систем.					
ПК 3.3. Производить исследование созданного программного кода с использованием специализированных	1 этап: Знания основные методы и средства эффективного анализа функционирования программного обеспечения; ос-	Совсем не умеет подготавливать оборудование к работе, плохо знает обработку статистического и	Не умеет подготовить оборудование к работе, нормально освоил знание обработку статистического и	Частично освоил навыки подготовки оборудования к работе, хорошо знает обработку статистиче-	Овладел навыками подготовки оборудования к работе, отлично освоил обработку статистического и	Устный опрос Тестовые задания Решение ситуационных задач

программ- ных средств с целью выявления ошибок и отклонения от алгорит- ма.	новые ви- ды работ на этапе сопро- вождения программ- ного обеспе- чения; ос- новные принципы контроля конфигура- ции и под- держки це- лостности конфигура- ции про- граммного обеспече- ния; средст- ва защиты программ- ного обеспе- чения в компьютер- ных систе- мах.	динамиче- ского кон- тента.	динамиче- ского кон- тента.	ческого и динамиче- ского кон- тента.	динамиче- ского кон- тента.	
	2 этап: Умения подбирать и настраивать конфигура- цию про- граммного обеспечения компьютер- ных систем; использо- вать методы защиты про- граммного обеспечения компьютер- ных систем; проводить инсталля- цию про- граммного обеспечения компьютер- ных систем; производить настройку отдельных					

	компонен- тов про- граммного обеспечения компьютер- ных систем; анализиро- вать риски и характери- стики каче- ства про- граммного обеспече- ния.					
	3 этап: Иметь прак- тический опыт в на- стройке от- дельных компонен- тов ПО ком- пьютерных сетей, вы- полнять от- дельные ви- ды работ на этапе под- держки ПО компьютер- ных систем.					
<i>ПК 3.4.</i> Проводить сравни- тельный анализ про- граммных продуктов и средств разработки, с целью выявления наилучшего решения согласно критериям, определен- ным техни- ческим за- данием..	1 этап: Зна- ния основ- ные методы и средства эффективно- го анализа функциони- рования про- грамм- ного обеспе- чения; ос- новные ви- ды работ на этапе сопро- вождения программ- ного обеспе- чения; ос- новные принципы контроля конфигура-	Не знает и не умеет разраба- тывать техниче- ские до- кументы для управ- ления ин- формаци- онными ресурсами.	Частично знает как разраба- тывать техниче- ские до- кументы для управ- ления ин- формаци- онными ресурсами, но не уме- ет исполь- зовать на- выки на практике.	Знает как разраба- тывать техниче- ские до- кументы для управ- ления ин- формаци- онными ресурсами, но не уме- ет исполь- зовать на- выки на практике.	Знает и умеет раз- рабатывать техниче- ские доку- менты для управле- ния ин- формаци- онными ресурсами.	<i>Устный опрос Тесто- вые за- дания Реше- ние си- туаци- онных задач</i>

	ции и поддержки целостности конфигурации программного обеспечения; средства защиты программного обеспечения в компьютерных системах.					
	2 этап: Умения подбирать и настраивать конфигурацию программного обеспечения компьютерных систем; использовать методы защиты программного обеспечения компьютерных систем; проводить инсталляцию программного обеспечения компьютерных систем; производить настройку отдельных компонентов программного обеспечения компьютерных систем; анализировать риски и характеристики качества про-					

	граммного обеспечения.					
	3 этап: Иметь практический опыт в настройке отдельных компонентов ПО компьютерных сетей, выполнять отдельные виды работ на этапе поддержки ПО компьютерных систем.					

4. Структура контрольно-оценочных средств, для экзамена по модулю

4.1. Перечень вопросов, выносимых на экзамен по модулю:

1. Дайте определение качества системы. Что понимают под характеристикой качества?
2. Что понимают под критерием качества? Перечислите требования, которым должен соответствовать критерий качества.
3. Объясните, для чего предназначены метрики программного кода.
4. По каким направлениям можно распределить метрики программ?
5. Перечислите четыре основные группы метрик оценки сложности программ. Дайте им характеристику.
6. Объясните предназначение метрик размера программ. Какую метрику оценки размера программ можете назвать? Опишите принцип вычисления данной метрики.
7. На чем основана оценка метрик сложности потока управления программ? Перечислите метрики оценки сложности потока управления программ. Опишите принцип вычисления самой популярной из них.
8. Объясните предназначение метрик стилистики и понятности программы.
9. Объясните предназначение измерительных методов оценки программ. Какие условия необходимы для применения измерительных методов?
10. Что такое программные измерительные мониторы? Для чего применяют измерительные мониторы?
11. Что понимают под корректностью программного средства? Что такое спецификация программного обеспечения?
12. Что включает в себя спецификация программного обеспечения? Для кого пишется данный документ и кем? Объясните основное назначение спецификации программного обеспечения.

13. Что такое верификация программного средства? В каком случае и для чего проводится верификация программного средства?
14. Что называют валидацией программного средства? В каком случае программному средству необходима валидация и с какой целью ее проводят?
15. Объясните предназначение тестирования программного обеспечения.
16. В чем суть тестирования методом «черного ящика»? В чем суть тестирования методом «белого ящика»?
17. Что такое дизассемблирование и декомпиляция? Для чего предназначены дизассемблеры? Какие методы применяются для противодействия дизассемблированию?
18. Объясните суть статического анализа при обратном проектировании. В чем заключается смысл динамического анализа при обратном проектировании?
19. Расскажите о классификации вредоносного программного обеспечения. Назовите методы анализа вредоносного программного обеспечения.
20. Объясните понятия несанкционированный доступ к программным средствам, защита программных средств, профилактика, вакцинирование. Что объединяет эти понятия?
21. Что включает управление программным проектом? Что такое инициация программного проекта?
22. Что определяет концепция проекта? Как планируется программный проект?
23. Каков оптимальный состав коллектива разработчиков? Каков состав проектной команды? Какие есть рекомендации по совмещению и не совмещению ролей?
24. Как осуществляется управление в программистской команде? Кто такой лидер? Как осуществляются подбор коллектива и эффективное взаимодействие в команде?
25. Что такое риск в программном проекте? Как управляют рисками в программных проектах?
26. Какие роли возможны в команде проекта? Нужна ли команда при разработке проекта?
27. Команда программистов задерживает сроки разработки проекта. Чьей задачей будет общение с заказчиком? Назовите роль.
28. Какой набор инструментов, помогающих организовать командную работу при работе над проектом вы можете назвать? Приведите примеры.
29. Для чего нужны системы контроля версий? Назовите типы систем контроля версий.
30. Зачем нужна база данных ошибок?
31. Группа программистов использует распределенную систему контроля версий. Во время работы членов команды над проектом произошла потеря данных в центральном репозитории. Можно ли восстановить данные?
32. Группа программистов использует блокирующую систему контроля версий. Один из них вносит изменения в файл tt.db. Будет ли этот файл доступен для записи? Будет ли этот файл доступен для чтения?
33. Что такое ревьюирование кода? Какова цель ревьюирования? Назовите синонимы словосочетания «ревьюирование кода». Какие задачи может решить ревьюирование кода?
34. Кто должен присутствовать в группе по ревьюированию кода? Какие недостатки возможны при самостоятельном ревьюировании кода?

35. Какие методы ревьюирования кода возможны? Как вы считаете, нужно ли делать ревьюирование кода или это потеря времени? Обоснуйте ответ.
36. Как вы считаете, как часто необходимо делать ревьюирование кода? Обоснуйте ответ.
37. Дайте определение утилиты. Перечислите известные утилиты для ревьюирования. Какие функции выполняют утилиты для ревьюирования?
38. Расскажите о разновидностях предпроцессинга кода.
39. Дайте определение интегрированной среды разработки и перечислите известные IDE.
40. Приведите определение валидации. Перечислите преимущества валидного кода. Обязательна ли валидация?
41. Приведите пример валидного и невалидного кода при решении одной и той же задачи.
42. Охарактеризуйте различные инструменты для ревьюирования.
43. Кратко охарактеризуйте Git.
44. Что такое отладчики и дизассемблеры? Для чего они нужны?
45. Охарактеризуйте инструментарий различных сред разработки.

4.2. Практические задания:

1. Решите задачу, используя только элементарные конструкции (последовательность, ветвления, циклы). Программа должна быть рабочей!

Для созданной программы подсчитать все необходимые значения и оценить метрические характеристики по Холстеду.

Задача: Дано натуральное число. Установить, является ли последовательность его цифр при просмотре их слева направо упорядоченной по возрастанию. Например, для числа 1478 ответ положительный, для чисел 1782 и 1668 — отрицательный и т. п.

2. Решите задачу, используя только элементарные конструкции (последовательность, ветвления, циклы). Программа должна быть рабочей!

Для созданной программы подсчитать все необходимые значения и оценить метрические характеристики по Чепину.

Задача: Дана последовательность вещественных чисел, оканчивающаяся числом 10 000. Количество чисел в последовательности не меньше двух. Определить, является ли последовательность упорядоченной по возрастанию. В случае отрицательного ответа определить порядковый номер первого числа, нарушающего такую упорядоченность.

3. Решите задачу, используя только элементарные конструкции (последовательность, ветвления, циклы). Программа должна быть рабочей!

Для созданной программы подсчитать все необходимые значения и оценить метрические характеристики интеллектуального содержания программы.

Задача: Дано натуральное число. Выяснить, является ли оно простым (простым называется натуральное число, большее 1, не имеющее других делителей, кроме единицы и самого себя).

4. Решите задачу, используя только элементарные конструкции (последовательность, ветвления, циклы). Программа должна быть рабочей!

Для созданной программы подсчитать все необходимые значения и оценить метрические характеристики по Холстеду.

Задача: В компьютер по очереди поступают результаты спортсменов-участников соревнований по лыжным гонкам, уже пришедших к финишу (время, затраченное на прохождение дистанции гонки). Выводить на экран лучший результат после ввода результата очередного спортсмена.

5. Решите задачу, используя только элементарные конструкции (последовательность, ветвления, циклы). Программа должна быть рабочей!

Для созданной программы подсчитать все необходимые значения и оценить метрические характеристики по Чепину.

Задача: Даны 20 чисел, образующие неубывающую последовательность. Несколько чисел, идущие подряд, равны между собой. Найти количество таких чисел. Сколько различных чисел имеется в последовательности?

6. Решите задачу, используя только элементарные конструкции (последовательность, ветвления, циклы). Программа должна быть рабочей!

Для созданной программы подсчитать все необходимые значения и оценить метрические характеристики интеллектуального содержания программы.

Задача: Натуральное число называется совершенным, если оно равно сумме своих делителей, включая 1 и, естественно, исключая это самое число. Например, число 6 — совершенное ($6=1+2+3$). Дано натуральное число. Выяснить, является ли оно совершенным.

7. Решите задачу, используя только элементарные конструкции (последовательность, ветвления, циклы). Программа должна быть рабочей!

Для созданной программы подсчитать все необходимые значения и оценить метрические характеристики по Холстеду.

Задача: Дан массив целых чисел. Рассмотреть отрезки массива (группы идущих подряд чисел), состоящие из нечетных чисел. Получить наибольшую из длин рассматриваемых отрезков.

8. Решите задачу, используя только элементарные конструкции (последовательность, ветвления, циклы). Программа должна быть рабочей!

Для созданной программы подсчитать все необходимые значения и оценить метрические характеристики по Чепину.

Задача: массиве из 20 элементов числа образуют неубывающую последовательность. Несколько элементов, идущих подряд, равны между собой. Найти количество таких элементов. Сколько различных чисел имеется в массиве?

9. Решите задачу, используя только элементарные конструкции (последовательность, ветвления, циклы). Программа должна быть рабочей!

Для созданной программы подсчитать все необходимые значения и оценить метрические характеристики интеллектуального содержания программы.

Задача: Дан массив ненулевых целых чисел. Определить, сколько раз элементы массива при просмотре от его начала меняют знак. Например, в массиве 10, -4, 12, 56, -4, -89 знак меняется 3 раза.

10. Зарисуйте предположительную схему информационных потоков соответственно заданию: Отдел по обслуживанию покупателей получает заказ от покупателя, записывает его и посылает в отдел продаж и производства, а копию - в отдел технической поддержки.

11. Зарисуйте предположительную схему информационных потоков соответственно заданию: На основе информации о заказе покупателя отдел технической

поддержки разрабатывает техническую спецификацию на тип пищевой смеси, которая требуется покупателю, и посылает ее в отдел продаж и производства.

12. Зарисуйте предположительную схему информационных потоков соответственно заданию: Используя информацию о заказе покупателя и техническую спецификацию, отдел продаж и производства оформляет заказ на поставку, а также информацию о текущем уровне запасов. Этот заказ и информация передаются в планово-производственный отдел.

13. Зарисуйте предположительную диаграмму пользовательских сценариев соответственно заданию: Менеджер и клиент интернет-магазина.

14. Зарисуйте предположительную диаграмму пользовательских сценариев соответственно заданию: Бюро по трудоустройству: работодатели, менеджер бюро и посетитель.

15. Зарисуйте предположительную диаграмму пользовательских сценариев соответственно заданию: Поликлиника: посетитель, регистратура, кабинет врача.

Перечень практических работ МДК.03.01 «Моделирование и анализ программного обеспечения»:

Практическая работа № 1. Сравнительная характеристика СКВ.

1. Ознакомиться с информацией по наиболее популярным СКВ:

Наиболее **популярные СКВ** по версиям интернет-ресурсов следующие (в скобках указан сайт разработчика):

- GIT — распределенная СКВ, разработана в 2005 г. (<https://git-scm.com/>);
- SVN (Subversion) — централизованная СКВ, разработана в 2002 г. (<http://subversion.apache.org/>);
- Mercurial — распределенная СКВ, разработана в 2005 г. (<http://mercurial-scm.org/>);
- CVS (Concurrent Versions System) — централизованная СКВ, разработана в 1990 г. (<http://nongnu.org/cvs/>);
- Team Foundation Server — централизованная СКВ, разработана в 2005 г. (<https://visualstudio.microsoft.com/tfs/>);
- Bazaar — распределенная СКВ, разработана в 2004 г. (<https://bazaar.canonical.com>).

2. Выполнить сравнительную характеристику СКВ:

Название	Описание/Характеристики	Достоинства	Недостатки
GIT			
SVN (Subversion)			
Mercurial			
CVS (Concurrent Versions System)			
Team Foundation Server			
Bazaar			
Вывод (какая лучше)			

Практическая работа № 2. Системы контроля версий виды, принципы организации работы.

1. Чем вызвана необходимость использования программ систем контроля версий?
2. Что представляет собой система контроля версий?
3. Приведите примеры программ контроля версий.
4. Что представляет собой репозиторий?
5. Перечислите возможности программ контроля версий.
6. Охарактеризуйте принцип работы централизованной системы контроля версий.
7. Охарактеризуйте принцип работы распределенной системы контроля версий.
8. Укажите представителей централизованной системы контроля версий.
9. Укажите представителей распределенной системы контроля версий.
10. Для каких целей можно использовать системы контроля версий кроме сопровождения разработки программных продуктов?

Практическая работа № 6. Тестирование программ методом "черного ящика".

Выполнение работы предусматривает следующую последовательность действий:

1. Ознакомиться с теоретическими сведениями по стратегиям тестирования.
2. В соответствии с вариантом задачи, подготовить тесты (не менее 5) по методикам стратегии "черного ящика".
3. Предлагаемые тесты свести в таблицу.
4. Разработать программу.
5. Выполнить тестирование. Занести в таблицу результаты.
6. Сделать вывод о роли тестирования с использованием стратегии "черного ящика" и возможностях его применения. Сформулировать его достоинства и недостатки.

Варианты задания

Построить таблицу значений функции $y=f(x)$, x изменяется от x_{min} до x_{max} с шагом dx . Проконтролировать правильность ввода x_{min} , x_{max} , dx и корректность вычисляемого выражения.

Примечание. В протоколе необходимо указать порядок выполнения операций в соответствии с их приоритетом.

$$1. y = \frac{a + 20b}{x^3} * \ln 2x - \frac{1}{(a-1)^2}$$

$$2. y = 3\sqrt{\frac{5x-9}{7.5ab} + 18} + e^{2x + \frac{0.5}{a}}$$

$$3. y = \frac{x^4 - a^3 - b^2}{\sqrt{19x - 3.5} + \ln a}$$

$$4. y = \sin \frac{e^x - 3a}{a^2 + b^2} + \frac{10}{x^3}$$

$$5. y = \cos \frac{(x-a)^2}{x-2a} - \frac{3.5}{\sqrt{xb}}$$

$$6. y = \sin \frac{a + \cos^2 x}{\cos x^2 - b} + 2.5a\sqrt{b}$$

$$7. y = \frac{\operatorname{tg} 3a - 20|b| - \sqrt{ab}}{x^2 + b^3}$$

$$8. y = 2 \operatorname{arctg} \frac{25a}{b} + 3 \cos^2 \frac{9xb}{b-x}$$

$$9. y = |x^2 - a^2| + \frac{9x^3}{(b-x)^3} * \sin \frac{x}{a}$$

$$10. y = |(2a - 7.5x)^3| + e^{\frac{2b}{x} - \frac{a}{2b}}$$

$$11. y = \sin 3x + \cos^2 \frac{x}{2a+b} - \frac{2x}{a}$$

$$12. y = \ln \frac{3x^3 - 2x^2 + x}{(a^2 + b)^2} + \frac{a}{x^3 - 4x^2 - x}$$

$$13. y = e^{|\sin(3ax+b)|} * \frac{x}{(\sqrt{ax} + bx^2)^3}$$

$$14. y = \sin \frac{|x|}{2\sqrt{a}} + \cos^2 \frac{x^3}{a+b}$$

$$15. y = 2 \operatorname{ctg} \frac{x^3 - 2x^2 + |x|}{(a + \sqrt{b})^3} - \frac{1}{x}$$

$$16. y = \sin 5e^{\frac{x}{a} + \frac{b}{2} - \frac{3}{x}} + \cos^2 3ax$$

Решение:

ФИО:

Вариант, условие задачи:

Номер теста	Назначение теста	Значения исходных данных	Ожидаемый результат	Реакция программы	Вывод
1					
2					
3					
4					
5					

Программа

Код:

Вывод:

Практическая работа № 8. Экспорт настроек в командной среде.

Цель и задачи исследования. Цель — получение навыков управления параметрами настройки IDE Visual Studio.

Задачи. Студенту необходимо изменить настройки среды разработки Visual Studio, осуществить их экспорт в файл и импортировать на другую вычислительную машину. В ходе выполнения работы следует делать снимки экрана, после чего оставить отчет в виде инструкции по выполнению необходимых операций для экспорта и импорта настроек

среды Visual Studio.

Практическая часть. Исследование выполняется в несколько этапов.

1. С учетом того что IDE Visual Studio обладает достаточно обширными возможностями в рамках настройки параметров, обеспечивающих удобство работы, студенту предлагается выполнить настройку наиболее заметных из них, отвечающих за внешний вид среды разработки, что, в свою очередь, позволит с легкостью фиксировать результат работы. Для этого необходимо осуществить следующие действия:

- в строке меню необходимо выбрать Инструменты → Параметры;
- в списке параметров необходимо выбрать Окружение → Общие;
- для изменения цветовой схемы среды в окошке с выпадающим списком Цветовая схема необходимо выбрать соответствующую вкусу схему и нажать кнопку Ок;
- для изменения настроек применяемых шрифтов необходимо выполнить первые два пункта и выбрать в списке пункт Шрифты и цвета, выполнить необходимые настройки и нажать кнопку Ок.

2. Для экспорта параметров среды в файл необходимо выбрать пункт главного меню Средства, подпункт Импорт и экспорт параметров. После запуска мастера импорта и экспорта параметров нужно выбрать, установив соответствующий флажок, вариант действий (импорт или экспорт) и нажать кнопку Далее. Выполняя действия, предлагаемые мастером, пометить необходимые для импорта/экспорта параметры и записать их в файл в указанное место.

3. Сохранить файл с параметрами и перенести его на другую машину с установленной средой Visual Studio.

4. Вызвав мастер импорта и экспорта параметров, импортировать настройки в новую среду разработки.

5. Оформить отчет, описав свои действия и разместив соответствующие снимки экрана.

Практическая работа № 12. Интегрированная среда разработки.

- 1) Ознакомьтесь с обзором и краткой историей ИСР;
- 2) Дайте обзор ИСР направленным на работу с одним ЯП:
Интегрированные среды разработки Visual Basic;
Интегрированные среды разработки Delphi;
Интегрированные среды разработки C/C++;
Интегрированные среды разработки Python;
Интегрированные среды разработки C#;
- 3) Проведите сравнительную характеристику ИСР предназначенных для нескольких ЯП:
Интегрированные среды разработки предназначенные для нескольких языков программирования (IntelliJ IDEA, NetBeans, Eclipse, Qt Creator, Geany, Embarcadero RAD Studio, Code::Blocks, Xcode, Microsoft Visual Studio);
- 4) Сделайте выводы, какое ИСР лучше и почему?? Какое ИСР вам показалось худшим выбором для разработки? Почему??

Практическая работа № 16. Проверки на стороне сервера.

Практическая часть. Исследование выполняется в несколько этапов.

1. Осуществить проверку с помощью сервиса validator.w3.org. Для проверки необходимы веб-ресурсы, доступные на клиентском компьютере, один из которых размещен в Интернете.

Для начала проверки необходимо перейти по адресу validator.w3.org. Откроется страница, на которой три вкладки (рис. 2.4).

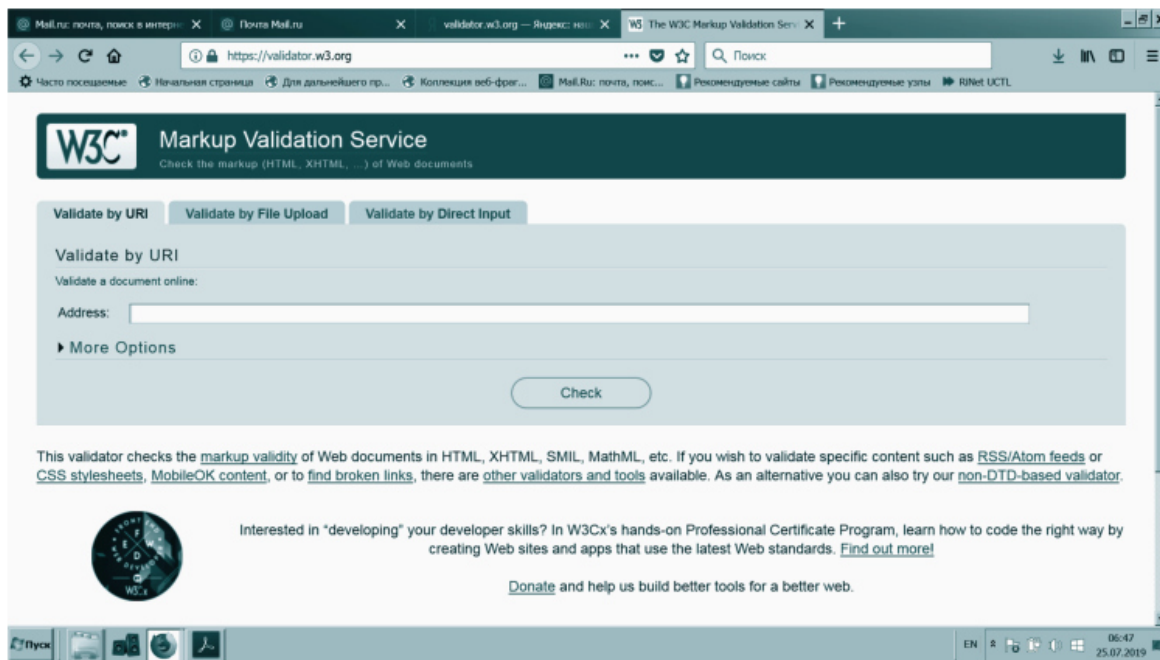


Рис. 2.4. Внешний вид окна валидатора

2. Первоначально проверить веб-сайт, размещенный в Интернете. Необходимо выбрать опцию на первой вкладке **Validate by URI**.

3. Далее нажать на кнопку **More options** и выставить следующие значения: **Character Encoding** — кодировка исследуемого сайта. Если она уже есть между тегами `<head>`, то выполнить действия следующего пункта.

4. На сайте в браузере на клиентском компьютере нажать на сочетание клавиш **CTRL+U** и найти в начале документа строку

```
<meta charset="UTF-8" />
```

Оставить выбранной опцию **detect automatically**. Если первой строкой указан тип текущего документа — **Document Type**, то оставить опцию — **detect automatically**. В поле **Address** ввести адрес исследуемого сайта. Нажать кнопку **Check**, которая расположена посередине серого блока.

5. Далее идет процесс валидации сайта, и через некоторое время появится результат валидации. Будет выведена страница с сообщением, например: «This document was successfully checked as HTML5!». Это значит, что ваш сайт успешно прошел проверку на валидность определенному типу документа — **HTML5**.

Если появилась надпись на красном фоне — это значит, что в **HTML**-документе присутствуют ошибки. Их необходимо исправить. Для этого следует просто выделить наз-

вание ошибки и попытаться проанализировать ее с помощью поискового сервера, изучить рекомендации, представленные в Интернете, для исправления ошибки либо обсудить пути возможного исправления в группе.

6. Проверить валидность сайта, загруженного с компьютера. На первой вкладке страницы валидатора выбрать опцию Validate by File Upload и провести валидацию ресурса. Проверку валидности сайта осуществить с помощью опции Validate by Direct Input. При этом содержимое файла вставить непосредственно в форму для ввода.

7. Оформить отчет о результатах валидации, привести скриншоты выполненного задания, результаты проверки и исправления обнаруженных погрешностей.

8. Дополнительно: провести проверку веб-ресурсов с помощью плагинов. Загрузить плагины для соответствующего браузера со следующих ресурсов:

[http://users.skynet.be/mgueury/mozilla/;](http://users.skynet.be/mgueury/mozilla/)

[https://chrome.google.com/webstore/detail/html-tidy-browser;](https://chrome.google.com/webstore/detail/html-tidy-browser;extensi/gljdonhfjnfdklljmfaabfpjlonflnm;)

[extensi/gljdonhfjnfdklljmfaabfpjlonflnm;](https://chrome.google.com/webstore/detail/html-tidy-browser;extensi/gljdonhfjnfdklljmfaabfpjlonflnm;)

[https://addons.opera.com/en/extensions/details/validator/.](https://addons.opera.com/en/extensions/details/validator/)

Практическая работа № 17. Практическая работа. Тест-план.

Вы являетесь руководителем команды QA в компании, занимающейся разработкой мобильного мессенджера. Вашей задачей является написание тест-плана, максимально полно описывающего процесс тестирования программного продукта.

В качестве программного продукта, для которого вы пишете тест-план, можете взять любой коммерческий мессенджер (Whatsapp, ICQ или что-то еще) согласно своему варианту.

Варианты:

1. Telegram
2. WhatsApp
3. Viber
4. WeChat
5. Discord
6. Skype
7. Signal
8. Microsoft Teams
9. VK Messenger
10. «ТамТам»
11. TenChat
12. Yandex Messenger
13. Agent.Mail.ru
14. TDM
15. ICQ
16. Line
17. Threema
18. Imo

Если для описания каких-то пунктов плана вам не будет хватать информации (например, вам требуется придумать сотрудников, которые занимаются тестированием или сослаться на несуществующий документ требований) — описывайте гипоте-

тический (но реалистичный) сценарий, как если бы эта информация была вам доступна.

Тест-план

1. Цель - предельно краткое описание цели разработки приложения (частично это напоминает бизнес-требования, но здесь информация подаётся в ещё более сжатом виде и в контексте того, на что следует обращать первостепенное внимание при организации тестирования и повышения качества);	
2. Области, подвергаемые тестированию — перечень функций и/или нефункциональных особенностей приложения, которые будут подвергнуты тестированию. В некоторых случаях здесь также приводится приоритет соответствующей области;	
3. Области, не подвергаемые тестированию — перечень функций и/или нефункциональных особенностей приложения, которые не будут подвергнуты тестированию. Причины исключения той или иной области из списка тестируемых могут быть самыми различными — от предельно низкой их важности для заказчика до нехватки времени или иных ресурсов. Этот перечень составляется, чтобы у проектной команды и иных заинтересованных лиц было чёткое единое понимание, что тестирование таких-то особенностей приложения не запланировано — такой подход позволяет исключить появление ложных ожиданий и неприятных сюрпризов;	
4. Тестовая стратегия и подходы — описание процесса тестирования с точки зрения применяемых методов, подходов, видов тестирования, технологий, инструментальных средств и т. д.;	
5. Критерии — этот раздел включает следующие подразделы: • приёмочные критерии, критерии качества — любые объективные показатели качества, которым разрабатываемый продукт должен соответствовать с точки зрения заказчика или пользователя, чтобы считаться готовым к эксплуатации; • критерии начала тестирования — перечень условий, при выполнении которых команда приступает к тестированию. Наличие этого критерия страхует команду от бес-	

смысленной траты усилий в условиях, когда тестирование не принесёт ожидаемой пользы; • критерии приостановки тестирования — перечень условий, при выполнении которых тестирование приостанавливается. Наличие этого критерия также страхует команду от бессмысленной траты усилий в условиях, когда тестирование не принесёт ожидаемой пользы; • критерии возобновления тестирования — перечень условий, при выполнении которых тестирование возобновляется (как правило, после приостановки); • критерии завершения тестирования — перечень условий, при выполнении которых тестирование завершается. Наличие этого критерия страхует команду как от преждевременного прекращения тестирования, так и от продолжения тестирования в условиях, когда оно уже перестаёт приносить ощутимый эффект;	
6. Ресурсы. В данном разделе тест-плана перечисляются все необходимые для успешной реализации стратегии тестирования ресурсы, которые в общем случае можно разделить на: • программные ресурсы; • аппаратные ресурсы; • человеческие ресурсы; • временные ресурсы; • финансовые ресурсы (во многих компаниях финансовые ресурсы могут быть представлены отдельным документом, так как являются конфиденциальной информацией).	
7. Расписание — фактически это календарь, в котором указано, что и к какому моменту должно быть сделано. Особое внимание уделяется так называемым ключевым точкам, к моменту наступления которых должен быть получен некий значимый ощутимый результат;	
8. Роли и ответственность — перечень необходимых ролей (например, «ведущий	

тестирующий», «эксперт по оптимизации производительности») и область ответственности специалистов, выполняющих эти роли;	
9. Оценка рисков — перечень рисков, которые с высокой вероятностью могут возникнуть в процессе работы над проектом. По каждому риску даётся оценка представляемой им угрозы и приводятся варианты выхода из ситуации;	
10. Документация — перечень используемой тестовой документации с указанием, кто и когда должен её готовить и кому передавать;	
11. Метрики — числовые характеристики показателей качества, которые могут включать в себя описание способов оценки и анализа результата. На этот раздел, как правило, формируется множество ссылок из других разделов тест-плана. В тестировании существует большое количество общепринятых метрик, многие из которых могут быть собраны автоматически с использованием инструментальных средств управления проектами. (Виды метрик для тестирования ПО можно посмотреть в предыдущей лекции)	
Вывод	

Практическая работа № 18. Практическая работа № 1. Управление персоналом проекта.

1. Вы являетесь руководителем проекта QA в компании, занимающейся разработкой мобильного мессенджера. Вашей задачей является собрать команду в соответствии с темой вашего проекта. Тему проекта можно выбрать по желанию.

Тема проекта			
	«Должность»	Описание должностных обязанностей (аргументируйте, зачем этот сотрудник нужен и какие задачи он будет выполнять)	На каком этапе работы с проектом (вспомните этапы разработки ПО) этот сотрудник будет участвовать в проекте

Если для описания каких-то пунктов плана вам не будет хватать информации (например, вам требуется придумать сотрудников, которые занимаются тестированием или сослаться на несуществующий документ требований) — описывайте гипотетический (но реалистичный) сценарий, как если бы эта информация была вам доступна.

2. Ответьте на вопрос, какие «Должности» мог бы выполнить один и тот же сотрудник, можно ли их совместить и почему???

Практическая работа № 19. Управление стоимостью проекта.

З а д а ч а 1.1. Как можно охарактеризовать состояние проекта продолжительностью 24 месяца и стоимостью 500 тыс. руб., если на отчетную дату показатель CPI равен 1,7, а SPI – 0,9? Какие прогнозы можно сделать исходя из имеющихся данных? Зная, что процент завершенности проектных работ составил 45 %, определите фактические затраты и освоенный объем.

З а д а ч а 1.2. По итогам 3-месячной работы над проектом (планируемый срок выполнения проекта 12 месяцев) были определены следующие данные. Прогнозная стоимость проекта составила 450 тыс. руб. При этом отклонение по завершении – 32 тыс. руб. Освоенный объем составил 100 тыс. руб., плановый – 110 тыс. руб. Определите прогнозную продолжительность проекта, фактические затраты, отклонение по стоимости, отклонение по срокам, а также индексы выполнения расписания, бюджета и эффективности. Дайте характеристику состояния данного проекта.

Перечень практических работ МДК.03.02 «Управление проектами»

Практическая работа № 1. Практикум. Метрика размера программы - метрика Холстеда.

Метрика Холстеда относится к метрикам, вычисляемым на основании анализа числа строк и синтаксических элементов исходного кода программы.

Метрика Холстеда позволяет оценить размер (в словах) и объем в битах программы на стадии анализа требований. Используя нормы выработки операторов в день можно оценить время на разработку.

Основу метрики Холстеда составляют четыре измеряемые характеристики программы:

n1 — число уникальных операторов программы, включая символы-разделители, имена процедур и знаки операций (словарь операторов);

n2 — число уникальных операндов программы (словарь операндов);

N1 — общее число операторов в программе;

N2 — общее число операндов в программе.

+, *, /, - это операторы

x, y, z, 999, -25, number1 - это операнды

На основании этих характеристик рассчитываются оценки:

Словарь программы (Halstead Program Vocabulary, HPVoc): $n = n1 + n2$;

Длина программы (Halstead Program Length, HPLen): $N = N1 + N2$;

Объем программы (Halstead Program Volume, HPVol): $V = N \log_2 n$;

Сложность программы (Halstead Difficulty, HDiff): $D = (n1/2) \times (N2 / n2)$;

На основе показателя HDiff предлагается оценивать усилия программиста при разработке при помощи показателя HEff (Halstead Effort): $H = D \times V$.

Таблица 1.1. Уровни языков программирования

Язык	λ	Отклонения
Естественный язык	2,16	0,74
PL/I	1,53	0,92
Алгол	1,21	0,74
Паскаль	1,25	0,76
Бейсик	1,22	0,72
Фортран	1,14	0,81
Ассемблер	0,88	0,42

Пример вычисления метрики Холстеда:

Опреаторы	Число операторов	Операнды	Число операндов
::	7	a	2
=	8	b	2
==	1	c	2
.	6	mAparam	5
!=	1	mBparam	5
<	1	mCparam	3
-	3	result.x1	3
/	3	result.x2	3
*	2	result.status	2
- binary	3	det	3
return	1	4	3
{	12	2	2
}	12		
(	13		
)	13		

$n1 = 15$

$n2 = 12$

$N1 = 86$

$N2 = 35$

$N = N1 + N2 = 86 + 35 = 121$

$n = n1 + n2 = 15 + 12 = 27$

$V = N * \log_2(n) = 121 * \log_2(27) = 575$

Задание:

- Решите задачу своего варианта, используя только элементарные конструкции (последовательность, ветвления, циклы). Программа должна быть рабочей! (Программа 1)
- Оптимизировать программу (можно использовать процедуры или функции) (Программа 2).

Внимание! Оптимизированная программа должна содержать проверки всех переменных, которые вводятся с клавиатуры.

- Для созданных программ оценить метрические характеристики по Холстеду;
- Сравнить полученные результаты. Оформить результаты в таблицу. Сделать соответствующие выводы.

Вариант задания №	
Код программы:	Код оптимизированной программы:
Расчет метрики:	Расчет метрики:

Практическая работа № 2. Практикум. Метрики сложности потока управления программ - метрика Чепина.

Задача 1 «Простые числа в матрице»

Дана целочисленная матрица размером $N \times M$. Вычислить и записать в одномерный массив количество простых чисел в каждом столбце матрицы. Размерность матрицы задается с клавиатуры, заполнение матрицы осуществляется посредством датчика случайных чисел. Разработать программу для решения задачи. На основе лексического анализа исходного текста программы определить значение метрики Чепина.

Реализация программы

Текст программы для реализации возможного алгоритма решения поставленной задачи представлен на языке C# (таблица 27).

Таблица 27 - Реализация программы для задачи «Простые числа в матрице»

Но- мера строк	Строки программы
1	using System;

```

2      namespace EX2
3      {
4      class Program
5      {
6      static void Main()
7      {
8      int n,m
9
10     int[,] a;
11     int[] b;
12     ConsoleKeyInfo клавиша;
13     int i,j,k,d;
14     bool p=false;
15     Random g;
16     do
17     {
18     Console.Clear();
19     Console.Write("Сколько строк: ");
20     n = int.Parse(Console.ReadLine());
21     Console.Write("Сколько столбцов: ");
22     m = int.Parse(Console.ReadLine());
23     g=new Random();
24     a = new int[n, m];
25     for (i = 0; i < n; i++)
26     for (j = 0; j < m; j++)
27     {
28     a[i, j] = int.Parse(g.Next(0,101));
29     }
30
31     Console.WriteLine("\nИсходная матрица");
32     for (i = 0; i < n; i++, Console.WriteLine())
33     for (j = 0; j < m; j++)
34     Console.Write("{0,8:d}", a[i, j]);
35     b = new int[m];
36
37     for (j = 0; j < m; j++)
38     {
39     for (i = k = 0; i < n; i++)
40     {
41     p = true;
42     for (d = 2; d < a[i,j]; d++)
43     if (a[i,j] % d == 0) p = false;
44
45
46     if (p) k++;
47     b[j] = k;
48     }
49     }
50
51
52     Console.WriteLine("\nКоличество простых");
53     for (i = 0; i < b.Length; i++)
54     Console.Write("{0,8:d}", b[i]);

```

```

55
56     Console.WriteLine("\nДля выхода нажмите клавишу ESC");
57     клавиша = Console.ReadKey(true);
58     } while (клавиша.Key!= ConsoleKey.Escape);
59
60     }
61     }
62     }

```

Оценка характеристик программы

Рассмотрим текст программы для оценки ее качества с помощью метрики Чепина, которая позволяет оценить меру трудности понимания программы на основе входных и выходных данных. Результат анализа объявленных переменных представлен в таблице 28.

Таблица 28 - Результат анализа объявленных переменных

№ п/ п	Наименование перемен- ных	Номера строк
<i>P</i> (для расчётов и обеспечения вывода)		
1	<i>m</i>	8
2	<i>n</i>	8
3	<i>a</i>	10
<i>M</i> (модифицируемые или создаваемые)		
1	<i>i</i>	13
2	<i>j</i>	13
3	<i>k</i>	13
4	<i>d</i>	13
5	<i>b</i>	11
<i>C</i> (управляющие переменные)		
1	<i>g</i>	15
2	<i>p</i>	14
3	<i>клавиша</i>	12
<i>T</i> (не используемые в программе)		
	Отсутствуют	

Переменные *m*, *n* и *a* используются в качестве исходных данных.

Переменные *i*, *j*, *k*, *d* и *b* в процессе выполнения программы создаются и модифицируются.

Переменные *g*, *p* и *клавиша* используются для управления выполнением программы.

Таким образом, исходя из результатов анализа исходного текста программы, получаем следующие значения характеристик: $P = 3$, $M = 5$, $C = 3$, $T = 0$.

Метрика Чепина: $Q = P + 2M + 3C + 0,5T = 3 + 2 \cdot 5 + 3 \cdot 3 + 0,5 \cdot 0 = 22$.

Выводы. На основе полученных значений метрики Чепина уровень сложности данного решения можно считать сравнительно низким, как так в исходном тексте программы используется незначительное количество переменных, что не затрудняет понимание программы.

Задание:

- Решите задачу своего варианта, используя только элементарные конструкции (последовательность, ветвления, циклы). Программа должна быть рабочей! (Программа 1)
- Оптимизировать программу (можно использовать процедуры или функции) (Программа 2).

Внимание! Оптимизированная программа должна содержать проверки всех переменных, которые вводятся с клавиатуры.

- Заполнить таблицу Анализа объявленных переменных для каждой задачи.
- Для созданных программ оценить метрические характеристики по Чепину;
- Сравнить полученные результаты. Оформить результаты в таблицу. Сделать соответствующие выводы.

Вариант задания № Условие задачи:	
Код программы:	Код оптимизированной программы:
Расчет метрики:	Расчет метрики:

Задача 1.

№ п/ п	Наименование перемен- ных	Номера строк
<i>P</i> (для расчётов и обеспечения вывода)		
1		
2		
3		
<i>M</i> (модифицируемые или создаваемые)		
1		
2		

3		
4		
5		
<i>C</i> (управляющие переменные)		
1		
2		
3		
<i>T</i> (не используемые в программе)		

Задача 2.

№ п/п	Наименование перемен- ных	Номера строк
<i>P</i> (для расчётов и обеспечения вывода)		
1		
2		
3		
<i>M</i> (модифицируемые или создаваемые)		
1		
2		
3		
4		
5		
<i>C</i> (управляющие переменные)		
1		
2		
3		
<i>T</i> (не используемые в программе)		