

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Гарант Максим Алексеевич  
Должность: Ректор  
Дата подписания: 10.11.2023 09:49:14  
Уникальный программный ключ:  
7708e7a47e66a8ee02711b298d7e78bd1e40bf88

Приложение  
к рабочей программе дисциплины

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ  
ПО ДИСЦИПЛИНЕ (МОДУЛЮ)**

**Системное программирование**

*(наименование дисциплины (модуля))*

Направление подготовки / специальность

**09.03.01 Информатика и вычислительная техника**

*(код и наименование)*

Направленность (профиль) / специализация

**Проектирование АСОИУ на транспорте**

*(наименование)*

## Содержание

1. Пояснительная записка.
2. Типовые контрольные задания или иные материалы для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих уровень сформированности компетенций.
3. Методические материалы, определяющие процедуру и критерии оценивания сформированности компетенций при проведении промежуточной аттестации.

## 1. Пояснительная записка

Цель промежуточной аттестации – оценивание промежуточных и окончательных результатов обучения по дисциплине, обеспечивающих достижение планируемых результатов освоения образовательной программы.

Формы промежуточной аттестации: **зачет - 6 семестр; курсовая работа, экзамен – 7 семестр.**

### Перечень компетенций, формируемых в процессе освоения дисциплины

Код и наименование компетенции	Код индикатора достижения компетенции
ПК-1: Способен разрабатывать требования и проектировать программное обеспечение	ПК-1.1 Разрабатывает программный код на языках программирования низкого уровня
	ПК-1.2 Осуществляет отладку программ, написанных на языке низкого уровня

### Результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Оценочные материалы (семестр )
ПК-1.1: разрабатывать программный код на языках программирования низкого уровня	Обучающийся знает: методы планирования проектных работ; методы концептуального проектирования; технические требования к интерфейсной графике; стандарты регламентирующие требования к эргономике разработки взаимодействия человек-систем;	Вопросы (№ 1–10)
	Обучающийся умеет: планировать проектные работы; выбирать методики разработки требований к системе и шаблоны документов требований к системе; формулировать цели, исходя из анализа проблем, потребностей и возможностей; разрабатывать графический дизайн интерфейсов;	Задания (№ 1–4)
	Обучающийся владеет: навыками составления и согласования перечня поставок требований к системе; выявления потребителей требований к системе и их интересов; определения значимых показателей деятельности объекта автоматизации, на изменение которых направлен проект; создавать графические документы в программах растровых и векторных изображений; разработки блок-схемы драйвера устройства; написания исходного кода драйвера устройства;	
ПК-1.2: осуществлять отладку программ, написанных на языке низкого уровня	Обучающийся знает: синтаксис, особенности программирования и стандартные библиотеки выбранного языка программирования, стандарты информационного взаимодействия систем, используемых на транспорте; государственные стандарты ЕСПД	Вопросы (1–10)
	Обучающийся умеет: создавать блок-схемы алгоритмов функционирования разрабатываемых программных продуктов, оценивать вычислительную сложность алгоритмов функционирования разрабатываемых программных продуктов, формулировать цели, исходя из анализа проблем, потребностей и возможностей, осуществлять отладку программных продуктов для целевой операционной системы, применяемой на транспорте	Задания (№ 5–8)
	Обучающийся владеет: отладки разработанного драйвера устройства, разработки и описания порядка	

	работ по созданию и сдаче системы; разработки и описания порядка работ по созданию и сдаче системы, изучения технической документации по языку программирования, системы команд процессора устройства, адресации памяти и регистров процессора устройства	
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

6 семестр

Промежуточная аттестация (зачет) проводится в одной из следующих форм:

- 1) Собеседование;
- 2) выполнение заданий в ЭИОС СамГУПС.

7 семестр

Промежуточная аттестация (курсовая работа) проводится в одной из следующих форм:

- 1) Собеседование
- 2) выполнение заданий в ЭИОС СамГУПС.

Промежуточная аттестация (Экзамен) проводится в одной из следующих форм

- 1) Собеседование;
- 2) выполнение заданий в ЭИОС СамГУПС.

## 2. Типовые<sup>1</sup> контрольные задания или иные материалы для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих уровень сформированности компетенций

### 2.1 Типовые вопросы (тестовые задания) для оценки знаниевого образовательного результата

Проверяемый образовательный результат:

Код и наименование индикатора достижения компетенции	Образовательный результат
ПК-1.1: разрабатывать программный код на языках программирования низкого уровня	Обучающийся знает: методы планирования проектных работ; методы концептуального проектирования; технические требования к интерфейсной графике; стандарты регламентирующие требования к эргономике разработки взаимодействия человек-систем;
<p>1. Предложения языка ассемблера состоят из следующих компонент:</p> <p>А) метка или имя;</p> <p>Б) мнемоника;</p> <p>В) операнды;</p> <p>Г) комментарии;</p> <p>Д) константы;</p> <p>Е) литералы;</p> <p>Ж) не знаю...</p> <p>2. Схема трансляции ассемблерного модуля состоит из следующих этапов:</p> <p>А) исходный модуль на языке ассемблера – объектный модуль – подключение библиотек и других объектных модулей – исполняемый модуль;</p> <p>Б) исходный модуль на языке ассемблера - подключение библиотек и других объектных модулей – объектный модуль – исполняемый модуль;</p> <p>В) подключение библиотек и других объектных модулей - исходный модуль на языке ассемблера – объектный модуль – исполняемый модуль;</p> <p>Г) нет правильного ответа;</p> <p>3. Для указания ассемблеру того, что в программе используются числа в двоичной системе исчисления необходимо:</p> <p>А) в конце каждого двоичного числа ставить букву «b»;</p> <p>Б) в конце каждого двоичного числа ставить обозначение «bit»;</p> <p>В) в начале каждого двоичного числа ставить букву «b», а в конце 2;</p> <p>Г) в начале каждого двоичного числа ставить цифру «2», а в конце букву «b»;</p> <p>Д) в начале каждого двоичного числа ставить букву «b»;</p> <p>Е) в конце каждого двоичного числа ставить цифру «2»;</p> <p>Ж) ничего не ставить, ассемблер сам разберётся, где двоичная запись, а где шестнадцатеричная;</p> <p>4. Шестнадцатеричное 96h в двоичной системе исчисления равно:</p> <p>А) 10010110;</p> <p>Б) 01101001;</p> <p>В) 0000011000001001;</p> <p>Г) 150;</p> <p>Д) нет правильно варианта;</p> <p>5. Для представления отрицательного числа в компьютере выполняются следующие операции:</p> <p>А) инверсия положительного числа– прибавление 1 к результату инверсии = отрицательное число;</p> <p>Б) прибавление 1 к положительному числу – инверсия результата = отрицательное число;</p> <p>В) побитовое сложение положительного числа с ним же самим – инверсия результата сложения плюс 1 = отрицательное число;</p> <p>Г) инверсия положительного числа - побитовое сложение инвертированного результата с ним</p>	

<sup>1</sup> Приводятся типовые вопросы и задания. Оценочные средства, предназначенные для проведения аттестационного мероприятия, хранятся на кафедре в достаточном для проведения оценочных процедур количестве вариантов. Оценочные средства подлежат актуализации с учетом развития науки, образования, культуры, экономики, техники, технологий и социальной сферы. Ответственность за нераспространение содержания оценочных средств среди обучающихся университета несут заведующий кафедрой и преподаватель – разработчик оценочных средств.

же самим плюс 1 = отрицательное число;

6. Чему будет равен результат при выполнении операции  $96h \text{ AND } 0Fh$  =:

- а)  $A5h$ ;
- б)  $10100101b$ ;
- в)  $110b$ ;
- г)  $06h$ ;
- д)  $6$ ;
- е)  $8CA$ ;
- ж)  $100011001010b$ ;

7. Процессор – это:

- А) кремневая плата или подложка с логическими цепями, состоящими из транзисторов, скрытая в пластмассовом корпусе, снабжённом контактными ножками; +
- Б) кремневая плата, обеспечивающая механизм страничной организации памяти, которая необходима для любой многозадачной операционной системы;
- В) кремневая плата, хранящая инструкции и данные в виде двоичных сигналов в двоичной системе исчисления;
- Г) надо подумать...

8. К регистрам общего назначения относят регистры:

- А)  $EAX$ ;
- Б)  $EBX$ ;
- В)  $ECX$ ;
- Г)  $EDX$ ;
- Д)  $EES$ ;
- Е)  $EDS$ ;
- Ж)  $ESS$ ;
- З)  $ECS$ ;

9.  $BH$  – это:

- А) один из регистров общего назначения;
- Б) верхние 16 разрядов регистра общего назначения;
- В) нижние 16 разрядов регистра общего назначения;
- Г) один из сегментных регистров;
- Д) часть сегментного регистра;
- Е) верхние 8 разрядов регистра общего назначения;
- ж) нижние 8 разрядов регистра общего назначения;

10. Выберите правильные записи команд:

- А)  $\text{mov ah}, 123h$ ;
- Б)  $\text{mov bx}, 12345h$ ;
- В)  $\text{mov dl}, 100h$ ;
- Г)  $\text{mov cx}, 1234h$ ;
- Д)  $\text{mov al}, 56h$ ;
- Е)  $\text{mov es}, ds$ ;
- Ж)  $\text{mov dx}, 0DEF0h$ ;

ПК-1.2: осуществлять отладку программ, написанных на языке низкого уровня

Обучающийся знает: синтаксис, особенности программирования и стандартные библиотеки выбранного языка программирования, стандарты информационного взаимодействия систем, используемых на транспорте; государственные стандарты ЕСПД

1. Сегментные регистры в архитектуре  $x86\_32$  имеют:

- А) 16 разрядов;
- Б) 20 разрядов;
- В) 8 разрядов;
- Г) 32 разряда;
- Д) 64 разряда;

2. Если  $SA$  – адрес начала сегмента,  $OA$  – смещение искомого байта относительно этого начала, то физический адрес ячейки памяти можно получить по формуле:

- А)  $SA * 16 + OA$ ;
- Б)  $SA * 4 + OA$ ;
- В)  $OA * 16 + SA$ ;

Г)  $OA*4+SA$ ;

Д) надо подумать...

3. Сегментные регистры:

А) хранят начальные адреса сегментов программы и обеспечивают возможность обращения к этим сегментам; +

Б) используются для хранения данных. В эти регистры может быть записан адрес возврата в основную программу после завершения работы процедуры;

В) хранят машинные коды команд после трансляции программы;

Г) хранят адрес инструкции, которая должна быть выполнена следующей;

Д) не знаю ((;

4. Выберите правильные трактовки:

А) флаг ZF – признак нуля;

Б) флаг CF – признак переноса;

В) флаг SF – признак знака;

Г) флаг TF – признак полупереноса;

Д) забыл(а);

5. Имя метки – это:

А) идентификатор, значением которого является адрес первого байта того предложения исходного текста программы, которое он обозначает;

Б) идентификатор, отличающий данную директиву от других одноимённых директив;

В) мнемоническое обозначение соответствующей области памяти для хранения машинной команды или директивы транслятора;

Г) идентификатор, который обозначает поименованную область памяти для хранения адреса следующей выполняемой команды;

6. Когда асемблер встречает в программе команду `jmp $+3` то:

А) прибавляет к переменной \$ цифру 3;

Б) прибавляет к машинному коду операции цифру 3;

В) к текущему смещению прибавляет 3 и переходит к команде, имеющей полученный адрес;

Г) прибавляет к содержимому регистра AX цифру 3 и переходит к команде, имеющей полученный адрес;

7. Системы для разработки новых программ на конкретном языке программирования:

а) системы программирования

б) программированные системы

в) системы ретуширования

8. К какому уровню языков относятся языки ассемблера:

а) среднего уровня

б) низкого уровня

в) высокого уровня

9. Какой язык программирования был создан в 1979 году и назван в честь первого в мире программиста:

а) Паскаль

б) Фортран

в) Ада

10. Для чего используют программы-ассемблеры:

а) для перевода программ в машинные коды

б) для обеспечения бесперебойной работы программ

в) для создания программ

## 2.2 Типовые задания для оценки навыкового образовательного результата

Проверяемый образовательный результат:

Код и наименование индикатора достижения компетенции	Образовательный результат
ПК-1.1: разрабатывать программный код на языках программирования низкого	Обучающийся умеет: планировать проектные работы; выбирать методики разработки требований к системе и шаблоны документов требований к системе; формулировать цели, исходя из анализа проблем, потребностей и возможностей; разрабатывать

уровня	<p>графический дизайн интерфейсов;</p> <p>Обучающийся владеет: навыками составления и согласования перечня поставок требований к системе; выявления потребителей требований к системе и их интересов; определения значимых показателей деятельности объекта автоматизации, на изменение которых направлен проект; создавать графические документы в программах растровых и векторных изображений; разработки блок-схемы драйвера устройства; написания исходного кода драйвера устройства;</p>
<p>Разработка программной реализации драйвера русификатора клавиатуры (переключение по выбранной клавише)</p> <p>Задание 1. Создать проект, настроить платформу и порты ввода вывода.</p> <p>Задание 2. Разработать структурную схему.</p> <p>Задание 3. Разработать функциональную схему.</p> <p>Задание 4. Произвести отладку созданной программы</p>	
ПК-1.2: осуществлять отладку программ, написанных на языке низкого уровня	<p>Обучающийся умеет: создавать блок-схемы алгоритмов функционирования разрабатываемых программных продуктов, оценивать вычислительную сложность алгоритмов функционирования разрабатываемых программных продуктов, формулировать цели, исходя из анализа проблем, потребностей и возможностей, осуществлять отладку программных продуктов для целевой операционной системы, применяемой на транспорте</p> <p>Обучающийся владеет: отладки разработанного драйвера устройства, разработки и описания порядка работ по созданию и сдаче системы; разработки и описания порядка работ по созданию и сдаче системы, изучения технической документации по языку программирования, системы команд процессора устройства, адресации памяти и регистров процессора устройства</p>
<p>Разработка программной реализации драйвера преобразования к верхнему регистру</p> <p>Задание 5. Создать проект, настроить платформу и порты ввода вывода.</p> <p>Задание 6. Разработать структурную схему.</p> <p>Задание 7. Разработать функциональную схему.</p> <p>Задание 8. Произвести отладку созданной программы</p>	

### 2.3. Перечень вопросов для подготовки обучающихся к промежуточной аттестации

1. Программы и программное обеспечение. Основные понятия и определения системного программирования.
2. Этапы подготовки программы.
3. Структура драйвера устройства.
4. Структура компилятора, компиляторы и интерпретаторы, объектная программа, трансляция в ассемблер.
5. Входной язык, целевой язык и язык реализации. Т-диаграммы.
6. Методы создания компилятора.
7. Фазы компиляции.
8. Языки и их представление.
9. Операции над языками.
10. Дерево разбора цепочек языка.
11. Конечный автомат
12. Режим командной строки и его назначение
13. Разновидности командных интерпретаторов
14. Запуск и завершение работы режима командной строки. Запуск команд и программ в режиме командной строки
15. Получение справок о командах в режиме командной строки
16. Исходный текст программы на Ассемблере
17. Компиляция и листинг программ Ассемблера
18. Редактирование связей и его результаты
19. Исполнение программы Ассемблера и его результаты
20. Запоминание результатов в отдельном файле
21. Отладка программ, начальные сведения



22. Ввод и редактирование текста программы Ассемблера
23. Запуск на компиляцию программы Ассемблера
24. Запуск на редактирование связей программы Ассемблера
25. Запуск на отладку программы Ассемблера
26. Запуск на выполнение программы Ассемблера
27. Применение командных файлов для работы с Ассемблером
28. Назначение и состав языка БНФ
29. Правила, нетерминальные переменные и метасимволы
30. Примеры описания на БНФ
31. Утилита BE , назначение и использование
32. Утилита CHOICE, назначение и использование
33. Утилита GREP, назначение и использование
34. Процедуры, их назначение и применение
35. Описание процедур
36. Параметры процедур и вызов процедур
37. Вложенные вызовы процедур
38. Пример программы с процедурами
39. Листинг программы с процедурами
40. Блок-схемы и описания данных
41. Вектор прерываний
42. Вложенные вызовы макрокоманд
43. Выгрузка резидента
44. Вызов старого обработчика прерывания
45. Директива EXITM
46. Директивы написания макрокоманд
47. Завершение основной программы при проверке повторной загрузки
48. Задание нового обработчика прерывания
49. Запуск части инициализации
50. Контроль наличия резидента
51. Локальные метки и переменные макрокоманд
52. Макрокоманды – процедуры этапа компиляции
53. Макрорасширения
54. Макросы и их применение
55. Назначение блок-схем программ
56. Назначение макросов
57. Обработка в одном резиденте нескольких прерываний
58. Обработка прерываний в процессоре (программных и аппаратных)
59. Описание данных и процедур резидента
60. Описание макросов
61. Определение и запоминание старого обработчика
62. Освобождение памяти внешнее из отдельной программы
63. Отладка макрокоманд
64. Оформление блок-схемы программы
65. Параметры макросов и макровыводы
66. Параметры по умолчанию
67. Понятие прерывания и их назначение
68. Построение резидентных программ
69. Пример простейшего резидента
70. Примеры блок-схем программ

71. Работа с вектором прерываний напрямую
72. Разбор параметров командной строки
73. Разработка блок-схем программ
74. Расчет размера резидента
75. Резидентные программы
76. Русификация сообщений резидента
77. Связь с резидентной программой из программы
78. Связь с резидентом с помощью клавиатуры
79. Служебные символы макрокоманд
80. Сравнение макросов и процедур
81. Структура резидентной программы
82. Условная компиляция
83. Установка резидента
84. Циклическая компиляция
85. Элементы блок-схем программ

#### **2.4. Перечень примерных тем курсовых работ**

Курсовая работа заключается в разработке и реализации трехуровневой системы клиент-сервер. Результатами курсовой работы являются программная реализация данных подсистем и пояснительная записка, оформленная в соответствии с требованиями стандартов и задания на курсовую работу.

Основной сервер выполнен в виде службы windows, предназначен для выполнения длительных операций и хранения данных. Клиент – Silverlight приложение. Клиент предназначен для организации взаимодействия пользователя с сервером с использованием графического интерфейса. Программы разработать с использованием языка C#. Должна быть реализована возможность одновременной работы клиентов через сеть с одним сервером.

Вариант задания

Сервер - планировщик задач (запуск программ и пакетных файлов, установка соединения с глобальной сетью, отсылка приложениям сообщений через определенное время). Также сервер может выключать компьютер по нескольким условиям: наступление определенного времени, загрузка процессора ниже определенного порога, интенсивность сетевого обмена ниже определенного порога.

1. Назначение макросов
2. Обработка в одном резиденте нескольких прерываний
3. Обработка прерываний в процессоре (программных и аппаратных)
4. Описание данных и процедур резидента
5. Описание макросов
6. Определение и запоминание старого обработчика
7. Освобождение памяти внешнее из отдельной программы
8. Отладка макрокоманд

### **3. Методические материалы, определяющие процедуру и критерии оценивания сформированности компетенций при проведении промежуточной аттестации**

#### **Критерии формирования оценок по ответам на вопросы, выполнению тестовых заданий**

- оценка «отлично» выставляется обучающемуся, если количество правильных ответов на вопросы составляет 100 – 90 % от общего объема заданных вопросов;
- оценка «хорошо» выставляется обучающемуся, если количество правильных ответов на вопросы – 89 – 76 % от общего объема заданных вопросов;
- оценка «удовлетворительно» выставляется обучающемуся, если количество правильных ответов на тестовые вопросы – 75–60 % от общего объема заданных вопросов;
- оценка «неудовлетворительно» выставляется обучающемуся, если количество правильных ответов – менее 60 % от общего объема заданных вопросов.

### **Критерии формирования оценок по результатам выполнения заданий**

**Зачтено»** – ставится за работу, выполненную полностью без ошибок и недочетов в соответствии с заданием. Обучающийся полностью владеет информацией по теме работы, решил все поставленные в задании задачи.

**«Не зачтено»** - ставится за работу, если обучающийся правильно выполнил менее 2/3 всего задания, использовал при выполнении неправильные алгоритмы, допустил грубые ошибки при программировании, сформулировал неверные выводы по результатам работы.

*Виды ошибок:*

- *грубые ошибки: незнание основных понятий, правил, норм; незнание приемов решения задач; ошибки, показывающие неправильное понимание условия предложенного задания.*

- *негрубые ошибки: неточности формулировок, определений; нерациональный выбор хода решения.*

- *недочеты: нерациональные приемы выполнения задания; отдельные погрешности в формулировке выводов; небрежное выполнение задания.*

### **Критерии формирования оценок по зачету**

**«Зачтено»** - обучающийся демонстрирует знание основных разделов программы изучаемого курса: его базовых понятий и фундаментальных проблем; приобрел необходимые умения и навыки, освоил вопросы практического применения полученных знаний, не допустил фактических ошибок при ответе, достаточно последовательно и логично излагает теоретический материал, допуская лишь незначительные нарушения последовательности изложения и некоторые неточности.

**«Не зачтено»** - выставляется в том случае, когда обучающийся демонстрирует фрагментарные знания основных разделов программы изучаемого курса: его базовых понятий и фундаментальных проблем. У экзаменуемого слабо выражена способность к самостоятельному аналитическому мышлению, имеются затруднения в изложении материала, отсутствуют необходимые умения и навыки, допущены грубые ошибки и незнание терминологии, отказ отвечать на дополнительные вопросы, знание которых необходимо для получения положительной оценки.

### **Критерии формирования оценок по написанию и защите курсовой работы**

**«Отлично»** (5 баллов) – получают обучающиеся, оформившие курсовую работу (курсовой проект) в соответствии с предъявляемыми требованиями, в которой отражены все необходимые результаты проведенного анализа, сделаны обобщающие выводы и предложены рекомендации в соответствии с тематикой курсовой работы, а также грамотно и исчерпывающе ответившие на все встречные вопросы преподавателя.

**«Хорошо»** (4 балла) – получают обучающиеся, оформившие курсовую работу (курсовой проект) в соответствии с предъявляемыми требованиями, в которой отражены все необходимые результаты проведенного анализа, сделаны обобщающие выводы и предложены рекомендации в соответствии с тематикой курсовой работы. При этом при ответах на вопросы преподавателя обучающийся допустил не более двух ошибок.

**«Удовлетворительно»** (3 балла) – получают обучающиеся, оформившие курсовую работу (курсовой проект) в соответствии с предъявляемыми требованиями. При этом при ответах на вопросы преподавателя обучающийся допустил более трёх ошибок.

**«Неудовлетворительно» (0 баллов)** – ставится за курсовую работу, если число ошибок и недочетов превысило удовлетворительный уровень компетенции

### **Критерии формирования оценок по экзамену**

**«Отлично»** (5 баллов) – обучающийся демонстрирует знание всех разделов изучаемой дисциплины: содержание базовых понятий и фундаментальных проблем; умение излагать программный материал с демонстрацией конкретных примеров. Свободное владение материалом должно характеризоваться логической ясностью и четким видением путей применения полученных знаний в практической деятельности, умением связать материал с другими отраслями знания.

**«Хорошо»** (4 балла) – обучающийся демонстрирует знания всех разделов изучаемой дисциплины: содержание базовых понятий и фундаментальных проблем; приобрел необходимые умения и навыки, освоил вопросы практического применения полученных знаний, не допустил фактических ошибок

при ответе, достаточно последовательно и логично излагает теоретический материал, допуская лишь незначительные нарушения последовательности изложения и некоторые неточности. Таким образом данная оценка выставляется за правильный, но недостаточно полный ответ.

**«Удовлетворительно»** (3 балла) – обучающийся демонстрирует знание основных разделов программы изучаемого курса: его базовых понятий и фундаментальных проблем. Однако знание основных проблем курса не подкрепляется конкретными практическими примерами, не полностью раскрыта сущность вопросов, ответ недостаточно логичен и не всегда последователен, допущены ошибки и неточности.

**«Неудовлетворительно»** (0 баллов) – выставляется в том случае, когда обучающийся демонстрирует фрагментарные знания основных разделов программы изучаемого курса: его базовых понятий и фундаментальных проблем. У экзаменуемого слабо выражена способность к самостоятельному аналитическому мышлению, имеются затруднения в изложении материала, отсутствуют необходимые умения и навыки, допущены грубые ошибки и незнание терминологии, отказ отвечать на дополнительные вопросы, знание которых необходимо для получения положительной оценки.