

09.03.01 Информатика и вычислительная техника (профиль «Технологии виртуальной и дополненной реальности»)

наименование ОПОП

Б1.О.16.01

шифр дисциплины

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

**Дисциплины
(модуля)**

Основы программирования

Разработчик (и):
Романовская Ю.В.
ФИО

доцент

должность

канд. физ.-мат. наук
ученая степень,
звание

Утверждено на заседании кафедры
информационных технологий (ИТ)
наименование кафедры

протокол № 6 от 01.02.2024

Заведующий кафедрой ИТ


подпись

Ляш О.И.
ФИО

1. Критерии и средства оценивания компетенций и индикаторов их достижения, формируемых дисциплиной (модулем)

Код и наименование компетенции	Код и наименование индикатора(ов) достижения компетенции	Результаты обучения по дисциплине (модулю)			Оценочные средства текущего контроля	Оценочные средства промежуточной аттестации
		<i>Знать</i>	<i>Уметь</i>	<i>Владеть</i>		
ОПК-8 Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ИД-1 _{ОПК-8} Способен использовать алгоритмические языки программирования, операционные системы и оболочки, современные среды разработки программного обеспечения ИД-2 _{ОПК-8} Способен составлять алгоритмы, писать и отлаживать коды на языке программирования, тестировать работоспособность программы, интегрировать программные модули, пригодные для практического применения	алгоритмические языки программирования, современные инструменты разработки программного обеспечения, основные приемы, применяемые при разработке программ.	составлять и формализовывать алгоритмы, писать код на языке программирования и выполнять его отладку, тестировать работоспособность программы, интегрировать программные модули.	навыками разработки на языке Си программ, пригодных для практического применения; навыками отладки и тестирования работоспособности программ на языке Си.	- комплект заданий для выполнения лабораторных работ; - типовые задания по вариантам для выполнения индивидуального проекта	Результаты текущего контроля, Экзаменационные билеты

2. Оценка уровня сформированности компетенций (индикаторов их достижения)

Показатели оценивания компетенций (индикаторов их достижения)	Шкала и критерии оценки уровня сформированности компетенций (индикаторов их достижения)			
	Ниже порогового («неудовлетворительно»)	Пороговый («удовлетворительно»)	Продвинутый («хорошо»)	Высокий («отлично»)
Полнота знаний	Уровень знаний ниже минимальных требований. Имели место грубые ошибки.	Минимально допустимый уровень знаний. Допущены не грубые ошибки.	Уровень знаний в объёме, соответствующем программе подготовки. Допущены некоторые погрешности.	Уровень знаний в объёме, соответствующем программе подготовки.
Наличие умений	При выполнении стандартных заданий не продемонстрированы основные умения. Имели место грубые ошибки.	Продемонстрированы основные умения. Выполнены типовые задания с не грубыми ошибками. Выполнены все задания, но не в полном объеме (отсутствуют пояснения, неполные выводы)	Продемонстрированы все основные умения. Выполнены все основные задания с некоторыми погрешностями. Выполнены все задания в полном объёме, но некоторые с недочетами.	Продемонстрированы все основные умения. Выполнены все основные и дополнительные задания без ошибок и погрешностей. Задания выполнены в полном объеме без недочетов.
Наличие навыков (владение опытом)	При выполнении стандартных заданий не продемонстрированы базовые навыки. Имели место грубые ошибки.	Имеется минимальный набор навыков для выполнения стандартных заданий с некоторыми недочетами.	Продемонстрированы базовые навыки при выполнении стандартных заданий с некоторыми недочетами.	Продемонстрированы все основные умения. Выполнены все основные и дополнительные задания без ошибок и погрешностей. Продемонстрирован творческий подход к решению нестандартных задач.
Характеристика сформированности компетенции	Компетенции фактически не сформированы. Имеющихся знаний, умений, навыков недостаточно для решения практических (профессиональных) задач. ИЛИ Зачетное количество баллов не набрано согласно установленному диапазону	Сформированность компетенций соответствует минимальным требованиям. Имеющихся знаний, умений, навыков в целом достаточно для решения практических (профессиональных) задач. ИЛИ Набрано зачетное количество баллов согласно установленному диапазону	Сформированность компетенций в целом соответствует требованиям. Имеющихся знаний, умений, навыков достаточно для решения стандартных профессиональных задач. ИЛИ Набрано зачетное количество баллов согласно установленному диапазону	Сформированность компетенций полностью соответствует требованиям. Имеющихся знаний, умений, навыков в полной мере достаточно для решения сложных, в том числе нестандартных, профессиональных задач. ИЛИ Набрано зачетное количество баллов согласно установленному диапазону

3. Критерии и шкала оценивания заданий текущего контроля

3.1 Критерии и шкала оценивания лабораторных работ

Перечень лабораторных работ, описание порядка выполнения и защиты работы, требования к результатам работы, структуре и содержанию отчета и т.п. представлены в методических материалах по освоению дисциплины (модуля) и в электронном курсе в ЭИОС МАУ.

Оценка/баллы	Критерии оценивания
<i>Отлично</i>	Задание выполнено правильно в полном объеме, результаты работы представлены в соответствии с требованиями.
<i>Хорошо</i>	Задание выполнено в полном объеме, но при верном в целом ходе выполнения допущены несколько незначительных ошибок (не влияющих на правильную последовательность действий) ИЛИ не более одной существенной ошибки. Все требования, предъявляемые к представлениям результатов работе, выполнены.
<i>Удовлетворительно</i>	Задание выполнено не в полном объеме ИЛИ с 2-3 существенными ошибками. Большинство требований, предъявляемых к представлению результатов работы, выполнены.
<i>Неудовлетворительно</i>	Задание не выполнено ИЛИ задание выполнено со значительным (более 3) количеством существенных ошибок. Большинство требований, предъявляемых к представлению результатов работы, не выполнены.

3.2 Критерии и шкала оценивания индивидуального проекта

Перечень контрольных заданий, рекомендации по выполнению представлены в методических материалах по освоению дисциплины (модуля) и в электронном курсе в ЭИОС МАУ.

В ФОС включены типовой вариант индивидуального проекта.

Типовой вариант индивидуального проекта (1 семестр)

Формулировка задания. Для идеи разработки программного средства, которая выбирается студентом самостоятельно, необходимо:

- описать постановку задачи на разработку программного средства;
- сформулировать функциональные и нефункциональные требования к программному средству;
- спроектировать и реализовать программное средство.

В результате выполнения работы необходимо предоставить:

1. программное средство в виде исполняемого файла и исходного кода;
2. описание процесса разработки программного средства в виде текстового документа в соответствии с планом, представленном в методических указаниях;
3. презентацию, представляющую результаты работы.

Результаты работы оцениваются в соответствии с критериями, представленными ниже.

№	Критерий оценки	Высокий уровень (4 балла)	Продвинутый уровень (3 балла)	Пороговый уровень (2,5 балла)	Уровень ниже порогового (0 баллов)
1.	Полнота описания всех этапов работы	Представлены результаты по всем трем этапам	Представлены результаты двух этапов	Представлены результаты двух этапов	Представлены результаты одного этапа
2.	Понимание содержания этапов	Полное понимание всех этапов и их последовательность	Полное понимание содержания	Частичное понимание содержания этапов	Непонимание содержания этапов

	разработки ПО	и	некоторых этапов		
3.	Владение средствами формализации результатов работы	Продемонстрированы навыки формализации результатов работы	Результаты частично формализованы, присутствуют небольшие ошибки	Результаты слабо формализованы, присутствуют 1-2 грубые ошибки	Отсутствует формализация результатов, присутствуют многочисленные грубые ошибки
4.	Наличие логической связи между алгоритмами и кодом программы	Связь четко прослеживается	Связь прослеживается, преимущественно не нарушена	Связь прослеживается, но местами частичная	Связь не прослеживается
5.	Владение терминологией	Свободное владение терминологией, язык изложения близок к профессиональному	Владение терминологией, грамотный язык изложения	Затруднения с владением терминологией	Слабое владение терминологией, язык изложения близок к бытовому
6.	Логичность изложения материала	Материал излагается последовательно, структура изложения логична	Прослеживается структура изложения	Незначительные нарушения последовательности и изложения	Последовательность изложения значительно нарушена
7.	Качество презентации	Слайды без избыточного материала, хорошо структурированы	Материал на единичных слайдах в некоторой части дублирует выступление или не соответствует ему	Материал на нескольких слайдах в некоторой части дублирует выступление или не соответствует ему	Материал на слайдах плохо читается, не соответствует выступлению
8.	Соблюдение регламента (8 мин)	Полностью представленный проект за отведенное время	Проект представлен полностью с незначительными (2-3 мин) нарушениями регламента	Проект представлен в большей части с нарушением регламента на 2-3 мин	Проект представлен не полностью со значительным нарушением регламента (более 3 мин)
9.	Ответы на вопросы	Ответы четкие, ясные, аргументированные на все вопросы	Ответы лишь частично аргументированы, получены на большинство вопросов	Ответы слабо аргументированы, получены на половину вопросов	Ответы не аргументированы или получены только на меньшую часть вопросов
10.	Наличие программной реализации	Программно реализованы большинство функций, при тестировании, при тестировании отсутствуют системные ошибки и предупреждения компилятора	Программно реализована половина функций, при тестировании отсутствуют системные ошибки и предупреждения компилятора	Программно реализованы менее половины функций, при тестировании присутствуют предупреждения компилятора при отсутствии системных ошибок	Программная реализация отсутствует или при тестировании присутствуют системные ошибки, препятствующие запуску программы

3.2 Критерии и шкала оценивания тестирования

Тестирование проводится на каждой лекции по пройденному материалу этой лекции. Тест содержит не более 10 вопросов. Тестирование проводится в электронном виде в курсе дисциплины в ЭИОС МАУ.

В ФОС включен типовой вариант тестового задания:

Вопрос 1. В чем отличие функций gets() и fgets()?

- Функция gets() отбрасывает символ '\n'
- Отличий нет
- Функция fgets() отбрасывает символ '\n'

Вопрос 2. Какой заголовочный файл содержит прототипы функций ввода/вывода?

- stdio.h
- windows.h
- system.h

Вопрос 3. Какое значение возвращает функция getch() в случае ошибки?

- 1
- NULL
- 0
- EOF

Вопрос 4. Какому типу принадлежит указатель на поток?

- const char *
- int *
- FILE
- FILE *

Вопрос 5. Что является параметром функции, которая используется для вывода на экран сообщения о конкретном типе ошибки?

- Символьная строка и код ошибки
- Код ошибки
- Символьная строка

Оценка/баллы	Критерии оценки
<i>Освоено / 1</i>	50-100 % правильных ответов
<i>Не освоено / 0</i>	Менее 50% правильных ответов

4. Критерии и шкала оценивания результатов обучения по дисциплине (модулю) при проведении промежуточной аттестации

Критерии и шкала оценивания результатов освоения дисциплины (модуля) с экзаменом

Для дисциплин (модулей), заканчивающихся экзаменом, результат промежуточной аттестации складывается из баллов, набранных в ходе текущего контроля и при проведении экзамена:

В ФОС включен список вопросов и заданий к экзамену и типовой вариант экзаменационного билета.

Список вопросов к экзамену:

1. Перечислите этапы решения задачи на ЭВМ.
2. Что понимается под жизненным циклом программного обеспечения?

3. Что представляет собой жизненный цикл разработки программного обеспечения?
4. Какие существуют модели, реализующие жизненный цикл разработки программного обеспечения?
5. Что представляет собой каскадная (водопадная) модель процесса разработки программного обеспечения?
6. Поясните разницу между итеративной и инкрементной моделями процесса разработки программного обеспечения.
7. На каких принципах основана гибкая методология разработки программного обеспечения?
8. В чем состоят особенности метода управления проектами Scrum?
9. В чем состоят сходства и различия методов Kanban и Scrum?
10. Что представляют собой процессы исследования и анализа предметной области?
11. Какие способы выявления требований к программному средству Вы знаете?
12. Какой цикл проходят требования к программному средству в процессе своей разработки?
13. Каким критериям должен удовлетворять список требований к программному средству?
14. Приведите примеры проверяемых и непроверяемых требований.
15. На какие группы можно разбить список требований к программному средству?
16. Что представляют собой требования к данным?
17. Что представляет собой этап проектирования программного средства?
18. Какие вопросы рассматриваются при выполнении проектирования данных?
19. Как выглядит процесс проектирования функций?
20. Перечислите свойства алгоритма.
21. Перечислите способы документирования результатов формализации алгоритма.
22. Что представляет собой блок-схема алгоритма?
23. Как выполняется нисходящее проектирование программы?
24. Перечислите современные виды пользовательских интерфейсов.
25. Что понимается под термином usability?
26. Перечислите преимущества хорошего пользовательского интерфейса.
27. Перечислите основные критерии эргономичности пользовательского интерфейса
28. Назовите несколько психологических и физиологических факторов, которые следует учитывать при проектировании пользовательского интерфейса.
29. Назовите основные этапы разработки пользовательских интерфейсов
30. С помощью каких функций языка Си можно настроить язык консоли?
31. Сколько существует класс лексем в языке Си? Перечислите их.
32. Что может выступать идентификатором в языке Си? Какие идентификаторы не рекомендуется использовать в качестве имен в прикладной программе?
33. Какие виды констант определены синтаксисом языка Си?
34. Как выглядит шаблон определения и инициализации переменной?
35. Что представляет собой первый параметр функций форматизированного ввода/вывода scanf() и printf()? Если ли различия в его содержимом у этих функций?
36. Какая особенность функции scanf() следует помнить при задании списка элементов для считывания?
37. На какие группы делятся операторы языка Си?
38. Как в языке Си интерпретируются отличные от нуля числовые значения, если они являются операндами логических операций?
39. В чем отличие блока от составного оператора?
40. Приведите пример полной формы условного оператора.
41. Для чего может быть использован оператор switch?
42. Каким требованиям должны отвечать метки оператора switch?
43. Что произойдет, если забыть поставить оператор break для какой-либо из меток оператора switch?
44. Какие три вида цикла определены в языке Си? Какой из них гарантирует выполнение тела цикла хотя бы 1 раз?
45. Может ли какая-нибудь из секций цикла с параметром быть пустой? Могут ли все секции быть пустыми?
46. Почему цикл с предусловием считается более предпочтительным для реализации по сравнению с циклом с постусловием?
47. Почему язык Си не считается сильно типизированным языком?
48. Какие типы, производные от базовых типов, определены в языке Си?
49. Чем отличаются друг от друга целочисленные типы языка Си?
50. Какие ключевые слова предназначены для модификации базового целочисленного типа?

51. Могут ли типы short, int, long и long long пересекаться по количеству выделяемых байт?
52. Почему тип char формально считается целочисленным?
53. Каким образом символьные константы записываются в памяти?
54. Какие типы данных предназначены для представления чисел с плавающей точкой?
55. Какой тип данных с плавающей точкой предоставляет максимальную точность?
56. Какой тип по умолчанию присваивается компилятором константе с плавающей точкой? Как можно задать требуемый тип константы с плавающей точкой?
57. Какие нештатные ситуации могут возникнуть при работе с операциями с плавающей точкой?
58. Какая ситуация называется потерей значимости?
59. Сформулируйте определение переменной.
60. Какие переменные называются локальными, а какие - глобальными?
61. Какие значения будут иметь неинициализированные глобальные и локальные переменные?
62. Какие существуют способы определения констант?
63. Дайте определение массива. Может ли массив содержать элементы разных типов?
64. Есть ли в языке Си ограничения на размерность массива?
65. В чем разница между массивами постоянной длины и массивами переменной длины?
66. При работе с массивом переменной длины может ли его размерность быть изменена после создания массива?
67. Как вычисляется размер памяти, отводимой под хранение массива?
68. Какая существует принципиальная особенность языка Си при работе с массивами?
69. Какое ключевое слово следует указать при объявлении массива, предназначенного только для чтения?
70. Что произойдет при попытке использовать неинициализированный массив?
71. Какова должна быть размерность символьного массива, предназначенного для работы со строкой из 20 символов? В чем состоит разница между 'С' и "С"?
72. Перечислите функции Си, позволяющие считывать строки.
73. В чем недостаток функции gets()?
74. Перечислите несколько функций обработки строк языка Си.
75. Как выглядит шаблон для объявления многомерного массива в языке Си?
76. Сформулируйте определение структурного типа. Для чего он используется?
77. Что могут представлять собой элементы структуры?
78. В чем разница между структурой (структурным типом) и экземпляром структуры?
79. С помощью каких операторов осуществляется доступ к конкретному элементу структуры?
80. Как можно инициализировать экземпляр структуры?
81. Какая операция позволяет получить адрес переменной? К каким объектам эту операцию нельзя применить?
82. Сформулируйте определение указателя в языке Си.
83. Для чего применяется специальная константа NULL?
84. Для чего используют операцию разыменования или косвенной адресации? Что является ее операндом?
85. Как выглядит шаблон для объявления указателя?
86. Какие основные операции допустимы над указателями в языке Си?
87. Какие особенности имеют операции инкремента и декремента при работе с указателями? Проявляются ли эти особенности при работе с аддитивными операциями?
88. Можно ли с помощью имени массива получить адрес первого элемента этого массива?
89. Какие два метода обращения к элементу массива существуют в Си? Какой из них обладает большим быстродействием?
90. Может ли быть составлен массив из указателей?
91. Как в языке Си определяется строка символов?
92. Что называется многоуровневой адресацией?
93. Есть ли ограничения на количество уровней в многоуровневой адресации?
94. Какой тип данных может возвращать функция в языке Си?
95. Если тип возвращаемого значения у функции не будет указан, как может интерпретировать такую ситуацию компилятор?
96. Какие требования предъявляются к списку параметров?
97. Что называется прототипом функции?
98. В каком случае допускается не использовать явно оператор return в теле функции?
99. В чем разница между формальными и фактическими параметрами функции?

100. В каком случае функция не может входить в выражения, требующие значения?
101. Какие существуют практические рекомендации по использованию оператора return?
102. Почему считается желательным использование прототипов функций?
103. Существуют ли функции, для которых не требуется прототип?
104. Какой способ передачи параметров предусматривает синтаксис языка Си?
105. Опишите процесс передачи параметров в функцию.
106. Что понимается под передачей параметров по ссылке?
107. Будет ли при передаче параметров по ссылке применен вызов по значению?
108. Что называется областью видимости?
109. Какие переменные называются глобальными, а какие локальными?
110. Какие рекомендации можно сформулировать по использованию глобальных и локальных переменных?
111. При совпадении имен локальной и глобальной переменных внутри блока к какой переменной произойдет обращение внутри этого блока?
112. Что называется единицей трансляции?
113. В чем особенность переменных, в объявлении которых присутствует спецификатор static?
114. Сколько раз при инициализации задается значение статической локальной переменной?
115. Для чего спецификатор static используется в объявлении глобальной переменной?
116. Какие продолжительности хранения могут иметь объекты в Си?
117. В чем разница между статической и автоматической продолжительностями хранения?
118. Сколько аргументов может иметь функция main()? Что представляют собой эти аргументы?
119. В каком случае объект будет называться рекурсивным?
120. Когда рекурсия будет называться множественной?
121. На что нужно обратить внимание при создании рекурсивной функции, чтобы ее выполнение было конечным?
122. Что называется деревом рекурсии?
123. Чем определяется объем памяти, требуемой для реализации рекурсивной подпрограммы?
124. Сформулируйте рекомендации по использованию рекурсии.
125. Приведите примеры задач, которые рекурсивны по своей природе.
126. Приведите определение потока.
127. Какие действия можно производить с потоком?
128. Какой заголовочный файл содержит прототипы функций ввода/вывода?
129. Что происходит при инициализации потока?
130. Какому типу принадлежит указатель на поток?
131. Сколько существует стандартных режимов работы с файлом? Чем они отличаются?
132. Для чего необходимо явным образом закрывать открытые файлы после работы с ними?
133. Какие ошибки могут возникнуть при открытии потока?
134. Как определить, что произошла ошибка при открытии потока?
135. Какая функция используется для вывода на экран сообщения о конкретном типе ошибки? Что является ее параметром?
136. Как проверить, успешно ли произошло закрытие файла?
137. Какие функции используются для записи и чтения символа при работе с файлом?
138. Какие функции используются для записи и чтения строк при работе с файлом?
139. В чем отличие функций gets() и fgets()?
140. Какие два способа представления файлов предлагаются в языке Си? В чем их особенности?
141. Что называется стандартным вводом, стандартным выводом и стандартным выводом ошибок?
142. С помощью какой функции можно перенаправить стандартные потоки?
143. Какие функции форматированного файлового ввода-вывода определены в Си?
144. Каким образом можно получать информацию о типе ошибки при работе с файлами?
145. Что понимают под динамическим распределением памяти? Как называется область памяти, которая может быть динамически выделена?
146. Перечислите функции динамического распределения памяти, определенные в языке Си.
147. Что представляет собой значение, возвращаемое функцией malloc(), и какого оно типа?
148. Как определить, что функция malloc() не смогла выделить память?
149. Какая функция предназначена для освобождения памяти? Почему важно освобождать память? В каких случаях говорят про утечку памяти?
150. В чем отличие функции calloc() от malloc()?
151. В каких случаях применяют функцию realloc()?

152. Сформулируйте определение структуры данных.
153. Что относят к фундаментальным структурам данных?
154. Какие два элементарные решения лежат в основе многих важных структур данных?
155. Чем обусловлена фундаментальность такой структуры данных как массив?
156. Как может быть решена проблема расположения в памяти больших массивов, для которых не может быть выделена одна сплошная область памяти?
157. Каким преимуществом перед массивами обладают связанные списки?
158. Что может представлять собой ссылка в последнем узле связанного списка?
159. Какая ошибка наиболее часто встречается при работе со связными списками?
160. Как расшифровывается аббревиатура LIFO? Какой структуре данных она соответствует?
161. Как расшифровывается аббревиатура FIFO? Какой структуре данных она соответствует?
162. Каким образом может быть выделена память для стека?
163. Как называется та сторона очереди, откуда производится удаление элемента?
164. Какие задачи могут потребовать использование стека, а какие – очереди?
165. Перечислите основные причины ошибок в программах.
166. Поясните, в каком порядке будут выполнены операторы в следующей строке: $x=++y$;
167. На какие основные категории делят ошибки при работе с указателями?
168. В чем состоит особенность ошибок при работе с указателями?
169. Какие ошибки часто возникают при работе с массивами?
170. Можно ли на этапе компиляции обнаружить ошибку при задании аргументов функции с переменным количеством параметров?
171. Когда может возникнуть переполнение стека, который используется для хранения локальных переменных?
172. В чем состоит разница в определении понятия отладки программ в отечественной и зарубежной литературе?
173. Дайте определение тестирования программы.
174. Какие приемы можно использовать для отладки программ?
175. Какие выделяют методы тестирования?
176. Объясните принцип работы инкрементного (нарастающего тестирования).
177. Что понимают под трассировкой программы? Как выглядит трассировочная таблица?
178. В чем состоят принципы стиля программирования?
179. Какие существуют рекомендации по выбору идентификаторов объектов?
180. Какие существуют рекомендации по составлению выражений и операторов?
181. Какие существуют рекомендации по использованию комментариев?
182. Перечислите критерии оценки стиля программирования.
183. Какие существуют популярные средства автоматического форматирования кода, поддерживающие язык Си?
184. Дайте определения сложности программы, временной сложности и пространственной (объемной) сложности программы.
185. Как можно оценить теоретическую временную сложность программы?
186. Что называется аналитическим профилем программы?
187. Сформулируйте алгоритм вычисления пространственной (объемной) сложности программы.

Пример экзаменационного билета

Вопросы для проверки сформированности компетенции ОПК-8:

1. Что понимается под жизненным циклом программного обеспечения?
2. Перечислите несколько функций обработки строк языка Си.
3. Какая функция предназначена для освобождения памяти? Почему важно освобождать память? В каких случаях говорят про утечку памяти?
4. Как можно оценить теоретическую временную сложность программы?

Оценка / баллы	Критерии оценки ответа на вопрос экзаменационного билета
----------------	--

Отлично / 5	Обучающийся глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, не затрудняется с ответом при видоизменении вопроса. Владеет специальной терминологией, демонстрирует общую эрудицию в предметной области, использует при ответе ссылки на материал специализированных источников, в том числе на Интернет-ресурсы.
Хорошо / 4	Обучающийся твердо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, владеет специальной терминологией на достаточном уровне; могут возникнуть затруднения при ответе на уточняющие вопросы по рассматриваемой теме; в целом демонстрирует общую эрудицию в предметной области.
Удовлетворительно / 3	Обучающийся имеет знания только основного материала, но не усвоил его деталей, допускает неточности, недостаточно правильные формулировки, плохо владеет специальной терминологией, допускает существенные ошибки при ответе, недостаточно ориентируется в источниках специализированных знаний.
Неудовлетворительно / 0	Обучающийся не знает значительной части программного материала, допускает существенные ошибки, нарушения логической последовательности в изложении программного материала, не владеет специальной терминологией, не ориентируется в источниках специализированных знаний. ИЛИ Нет ответа на поставленный вопрос.

Оценка, полученная на экзамене, складывается из баллов за каждый вопрос экзаменационного билета, переводится в баллы («5» - 20 баллов, «4» - 15 баллов, «3» - 10 баллов) и суммируется с баллами, набранными в ходе текущего контроля.

Итоговая оценка по дисциплине (модулю)	Суммарные баллы по дисциплине (модулю), в том числе	Критерии оценивания
Отлично	91 - 100	Выполнены все контрольные точки текущего контроля на высоком уровне. Экзамен сдан
Хорошо	81-90	Выполнены все контрольные точки текущего контроля. Экзамен сдан
Удовлетворительно	70- 80	Контрольные точки выполнены в неполном объеме. Экзамен сдан
Неудовлетворительно	69 и менее	Контрольные точки не выполнены или не сдан экзамен

Критерии и шкала оценивания результатов освоения дисциплины (модуля) с зачетом

Если обучающийся набрал зачетное количество баллов согласно установленному диапазону по дисциплине (модулю), то он считается аттестованным.

Оценка	Баллы	Критерии оценивания
Зачтено	60 - 100	Набрано зачетное количество баллов согласно установленному диапазону
Незачтено	менее 60	Зачетное количество согласно установленному диапазону баллов не набрано

5. Задания диагностической работы для оценки результатов обучения по дисциплине (модулю) в рамках внутренней и внешней независимой оценки качества образования

ФОС содержит задания для оценивания знаний, умений и навыков, демонстрирующих уровень сформированности компетенций и индикаторов их достижения в процессе освоения дисциплины (модуля).

Комплект заданий разработан таким образом, чтобы осуществить процедуру оценки каждой компетенции, формируемых дисциплиной (модулем), у обучающегося в письменной форме.

Содержание комплекта заданий включает: *тестовые задания*.

Комплект заданий диагностической работы

ОПК-8. Способен разрабатывать алгоритмы и программы, пригодные для практического применения	
1.	Выполнение действий друг за другом в алгоритме называется: <ul style="list-style-type: none"> – ветвлением; – циклом; – действием; – следованием.
2.	Выполнение действий друг за другом в алгоритме называется: <ul style="list-style-type: none"> – ветвлением; – циклом; – действием; – следованием.
3.	<div style="display: flex; align-items: flex-start;"> <div style="flex: 1;"> <pre> graph TD Start([НАЧАЛО]) --> Assign[k = 102] Assign --> Calc[k = k - 12] Calc --> Dec{k > 6?} Dec -- Да --> Calc Dec -- Нет --> End([КОНЕЦ]) </pre> </div> <div style="flex: 2; padding-left: 20px;"> <p>Каково значение будет иметь переменная k в конце работы программы, описываемой блок-схемой:</p> <ul style="list-style-type: none"> – 4; – 5; – 6; – 7 </div> </div>
4.	Этап, на котором определяются способы решения поставленной задачи, называется: <ul style="list-style-type: none"> – проектирование; – тестирование; – анализ предметной области; – реализация.
5.	Тело какого цикла выполнится по меньшей мере один раз в любом случае: <ul style="list-style-type: none"> – цикл с предусловием; – цикл с постусловием; – цикл с параметром.
6.	Этап, на котором проверяется соответствие реализованной программы

	<p>предъявляемым требованиям, называется:</p> <ul style="list-style-type: none"> – проектирование; – тестирование; – анализ предметной области; – реализация.
7.	<p>При создании простой программы на Си Вы написали следующую строчку:</p> <pre style="text-align: center;">int main()</pre> <p>Какая информация заложена в ней?</p> <ul style="list-style-type: none"> – Главная функция с целочисленным аргументом; – Главная функция с двумя параметрами; – Главная функция с целым типом возвращаемого значения; – Главная функция с целочисленным аргументом и целым типом возвращаемого значения.
8.	<p>Какой оператор языка Си Вы используете для сопоставления ромбовидному блоку в блок-схеме?</p> <ul style="list-style-type: none"> – while; – break; – if...else; – scanf;
9.	<p>При составлении блок-схемы алгоритма прямоугольный блок с одним входом и одним выходом называется:</p> <ul style="list-style-type: none"> – Блоком ветвления; – Блоком цикла; – Блоком действия; – Блоком-разделителем.
10.	<p>Чему будут равны значения переменных s и t после выполнения следующего фрагмента кода:</p> <pre>t = 2; s = t++;</pre> <ul style="list-style-type: none"> – s = 2, t = 2 – s = 3, t = 3 – s = 2, t = 3 – s = 3, t = 2