

Компонент ОПОП

09.03.01 Информатика и

вычислительная техника

наименование ОПОП

Б1.О.14.02

шифр дисциплины

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Дисциплины (модуля) Объектно-ориентированное программирование

Разработчик:

Парфенов С.А.

ФИО

ст. преподаватель

должность

учёная степень, звание

Утверждено на заседании кафедры

Информационных технологий

Наименование кафедры

протокол № 6 от 01.02.2024

Заведующий кафедрой

ИТ

Ляш О.И.

подпись

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	-основы математики, физики, вычислительной техники и программирования. -современные информационные технологии и программные средства, в том числе отечественного производства, которые могут быть использованы при решении задач профессиональной деятельности; -принципы работы современных информационных технологий и программных средств. - алгоритмические языки программирования, современные среды разработки программного обеспечения. -возможности современных средств разработки программного обеспечения. -методологии разработки программного обеспечения и технологии программирования. -методы и средства проектирования программного обеспечения. -методы и приемы формализации задач. - особенности построения объектно-ориентированных программных средств. - отличие процедурного программирования от ООП. - теорию ООП. - некоторые паттерны проектирования	- решать стандартные профессиональные задачи с применением естественнонаучных и общинженерных знаний, методов математического анализа и моделирования. - выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности. - составлять алгоритмы, писать и отлаживать коды на языке программирования, тестировать работоспособность программы, интегрировать программные модули. -проводить анализ требований к программному обеспечению и их исполнению, вырабатывать варианты и средства реализации требований к программному обеспечению. -применять методы и средства проектирования программного обеспечения, структур данных, пользовательских интерфейсов. -проводить оценку и обоснование принимаемых проектных решений. - формулировать цели и определять пути их достижения в рамках ООП.	- навыками теоретического и экспериментального исследования объектов профессиональной деятельности. - навыками применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности. - языком программирования С++; навыками отладки и тестирования работоспособности программы. -навыками формализации задач, выдвижения требований к программному обеспечению; -навыками анализа возможностей реализации требований к программному обеспечению, согласования требований к программному обеспечению с заинтересованными сторонами; -навыками разработки и согласования технических спецификаций на программные компоненты. -навыками проектирования программного обеспечения, структур данных, пользовательских интерфейсов. - приемами ООП.	комплект заданий для лабораторных работ	Экзаменационные билеты Результаты текущего контроля

			<ul style="list-style-type: none">- различать существующие приемы ООП.применять в профессиональной деятельности ООП.-разрабатывать программные средства, используя ООП	<ul style="list-style-type: none">- объектно-ориентированным языком программирования C++		
--	--	--	--	--	--	--

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 умение	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 объёме (отсутствуют пояснения, неполные выводы)	Продемонстрированы все основные умения. Выполнены все основные задания с некоторыми погрешностями. Выполнены все задания в полном объёме, но с некоторыми недочётами.	Продемонстрированы все основные умения. Выполнены все основные и дополнительные задания без ошибок и погрешностей. Задания выполнены в полном объёме без недочётов.
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 недочётами.	Продемонстрированы базовые навыки при выполнении стандартных заданий с некоторыми недочётами.	Продемонстрированы все основные умения. Выполнены все основные и дополнительные задания без ошибок и погрешностей. Продемонстрирован творческий подход к решению нестандартных задач.
Характеристика сформированности компетенций	Компетенции фактически не сформированы. Имеющихся знаний, умений, навыков недостаточно для решения практических (профессиональных) задач. ИЛИ Зачётное 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 зачётное количество баллов согласно установленному диапазону	Сформированность компетенций в целом соответствует требованиям. Имеющихся знаний, умений, навыков достаточно для решения стандартных профессиональных задач. ИЛИ Набрано зачётное 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 зачётное количество баллов согласно установленному диапазону

3. Критерии и шкала оценивания заданий текущего контроля

3.1 Критерии и шкала оценивания лабораторных работ

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

Баллы	Критерии оценивания
10	Задание выполнено полностью и правильно. Отчёт по практической работе подготовлен качественно в соответствии с требованиями. Полнота ответов на вопросы преподавателя при защите работы.
8	Задание выполнено полностью, но нет достаточного обоснования или при верном решении допущена незначительная ошибка, не влияющая на правильную последовательность рассуждений. Все требования, предъявляемые к работе, выполнены.
5	Задания выполнены частично с ошибками. Демонстрирует средний уровень выполнения задания на лабораторную работу. Большинство требований, предъявляемых к заданию, выполнены.
0	Задание выполнено со значительным количеством ошибок на низком уровне. Многие требования, предъявляемые к заданию, не выполнены. ИЛИ Задание не выполнено.

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

4.1 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:

1. Стиль программирования.Комментарии. Примеры.
2. Встроенные типы. Примеры.
3. Преобразование типов. Примеры.
4. Операции: арифметические. Примеры.
5. Операции: логические, сравнения. Примеры.
6. Операции:инкремент и декремент.Примеры.
7. Операции: присваивание. Преобразование типов в операции присваивания.

Примеры.

8. Операции: поразрядные. Примеры.
9. Операции: запятая. Примеры.
10. Приоритет операций. Примеры.
11. Директивы препроцессора. Примеры.
12. Инструкции:if, оператор ?:. Примеры.
13. Инструкции: break, continue, return. Примеры.
14. Инструкции: switch. Примеры.
15. Цикл for, цикл с предусловием, цикл с постусловием. Примеры.
16. Массивы. Многомерные массивы. Примеры.
17. Указатели. Бестиповые и типизированные указатели. Арифметика указателей.

Примеры.

18. Указатели. Преобразование типа указателя. Примеры
19. Ссылки. Ссылки gvalue. Примеры.

20. Строки. Примеры.
21. Указатели и многомерные массивы. Примеры.
22. Динамические массивы. Примеры.
23. Структуры. Примеры.
24. Объединения. Примеры.
25. Перечисления. Примеры.
26. Функции: объявление, определение, формальные и фактические параметры. Примеры.
27. Функции: передача аргументов по значению. Примеры.
28. Функции: передача аргументов по ссылке, по адресу. Примеры.
29. Функции: указатели на функцию. Примеры.
30. Функции: аргументы по умолчанию, перегрузка. Примеры.
31. Функции: рекурсия. Примеры.
32. Функции: массивы как параметры. Примеры.
33. Функции: константные параметры. inline – функции. Примеры.
34. Функции: левые функции. Примеры.
35. Функции: с переменным числом параметров. Примеры.
36. Функции: возврат по ссылке, по адресу. Примеры.
37. Аргументы командной строки, assert, static_assert. Примеры.
38. Стандартные потоки C++. Примеры.
39. Потоки и файлы. Примеры.
40. Строковые потоки. Примеры.
41. Классы. Описание. Поля. Методы. Доступ к полям. Роли. Примеры.
42. Классы. Конструктор. Свойства конструктора. Примеры.
43. Классы. Деструктор. Свойства деструктора. Примеры.
44. Классы. Конструктор копирования. Примеры.
45. Классы. Статические элементы. Примеры.
46. Классы. Константные поля, методы и объекты. Примеры.
47. Классы. Указатель this. Примеры.
48. Классы. Встроенное и внешнее определение методов. Примеры.
49. Лямбда функции. Примеры.

Оценка	Баллы	Критерии оценивания
Отлично	17-20	Обучающийся глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, чётко 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	13-16	Обучающийся твёрдо 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	10-12	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	Менее 10	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, переводится в баллы и суммируется с баллами, набранными в ходе текущего контроля.

Оценка	Баллы	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	61-80	Контрольные точки выполнены в неполном объеме. Экзамен сдан
Неудовлетворительно	Менее 61	Контрольные точки не выполнены или не сдан экзамен

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

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, расчетные задачи.

Комплект заданий диагностической работы

ОПК-8 Способен разрабатывать алгоритмы и программы, пригодные для практического применения	
1	<p>1. Выберите верный синоним стандарта C++14:</p> <p>а) c++1x</p> <p>б) c++1y</p> <p>в) c++1z</p> <p>2. Есть ли возможность в Code::Blocks настроить один и тот же стандарт для всех проектов:</p> <p>а) Да</p> <p>б) Нет</p> <p>3. Перечислите не менее двух версий компиляторов с поддержкой C++17.</p> <p>4. Разработать программу, которая позволяет пользователю,</p>

выбрав один из трёх способов вычисления:

- с помощью цикла for,
- цикла do..while,
- цикла while

вычислить произведение: $\prod_{k=1}^{10} \sqrt[3]{k}$.

5. Разработать программу, которая вычисляет сумму представленной последовательности, используя цикл for: 13, 23, 43, 63, ..., 1003.
6. Разработать программу для вывода квадратного корня из натуральных чисел в интервале: от 150 до 200.
7. Разработать программу табуляции функции $y = 5x - x + 3$, $x \in [2; 5]$. Шаг табуляции 0.2.
8. Разработать программу, которая заменяет на ; все пробелы в тексте, заданном пользователем.
9. Разработать программу, которая заменяет исходный текст, заданный пользователем - соответствующими символами азбуки Морзе.
10. Разработать программу, которая в заданной пользователем матрице размера $m \times n$, переставляет строки в порядке убывания максимальных элементов каждой строки.