

**Компонент ОПОП 01.03.02 Прикладная математика и информатика**  
**Направленность (профиль) Системное программирование и компьютерные технологии**

**Б1.О.15.02**

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**

**Дисциплины  
(модуля)**

**Языки и технологии программирования**

Разработчики:  
Козинец Егор Анатольевич,  
доцент кафедры информационных  
технологий;

Королева Наталья Юрьевна,  
доцент кафедры  
информационных технологий,  
канд. пед. наук, доцент

Утверждено на заседании кафедры  
информационных технологий  
протокол № 6 от 01.02.2024

Заведующий кафедрой ИТ

\_\_\_\_\_ О.И. Ляш

## 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>		
<p><b>ОПК-2.</b> Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач</p> <p><b>ОПК-5.</b> Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения</p>	<p><b>ИД-1оПК-2</b> Использует и адаптирует существующие математические методы для разработки и реализации алгоритмов решения прикладных задач</p> <p><b>ИД-2оПК-2</b> Использует существующие системы программирования для разработки и реализации алгоритмов решения прикладных задач.</p> <p><b>ИД-1оПК-5</b> Разрабатывает алгоритмы решения практических задач в области профессиональной деятельности.</p> <p><b>ИД-2оПК-5</b> Способен</p>	<ul style="list-style-type: none"> <li>– парадигмы программирования и реализующие их конкретные технологии;</li> <li>– основные характеристики языка программирования, определения алфавита, синтаксиса и семантики; основные этапы эволюции языков и технологий программирования; понятие и состав среды разработки;</li> <li>– способы человеко-машинного взаимодействия при решении задач на ЭВМ; этапы решения задач на ЭВМ и жизненного цикла программного продукта;</li> <li>– виды модулей в составе проекта (совокупности файлов, обеспечивающих решение задачи на ЭВМ);</li> <li>– процессы, происходящие при сборке проекта, и реализующие их утилиты;</li> <li>– понятия отладки, тестирования, верификации и валидации программ, виды ошибок; критерии оценки качества программы; понятие спецификации программы;</li> <li>– определение алгоритма и его свойства, способы записи алгоритма, виды структур алгоритмов;</li> <li>– назначение и правила оформления</li> </ul>	<ul style="list-style-type: none"> <li>– анализировать условие задачи на целесообразность применения той или иной технологии программирования;</li> <li>– выделять отдельные подзадачи в соответствии с выбранной технологией;</li> <li>– определять, какие действия, на каком этапе разработки программы выполняются;</li> <li>– выделять при решении задачи наиболее важные критерии качества программы;</li> <li>– осуществлять отладку программ различными способами;</li> <li>– выделять и обрабатывать исключительные ситуации (ошибки) вычислительного процесса;</li> <li>– составлять план тестирования, формировать и</li> </ul>	<ul style="list-style-type: none"> <li>– терминологией для определения и описания этапов жизненного цикла программ;</li> <li>– навыками выделения и формулирования этапов решения задач на ЭВМ в контексте выбранной технологии программирования;</li> <li>– технологией реализации процедурной (структурной) парадигмы программирования средствами выбранной языка программирования;</li> <li>– навыками сборки, отладки, тестирования и</li> </ul>	<ul style="list-style-type: none"> <li>- комплект заданий для выполнения лабораторных работ;</li> <li>- тестовые задания;</li> </ul>	<p>Результаты текущего контроля</p>

	<p>разрабатывать компьютерные программы пригодные для практического применения.</p> <p><b>ИД-30пк-5</b></p> <p>Применяет алгоритмы и компьютерные программы для решения практических задач в области профессиональной деятельности.</p>	<p>основных алгоритмических конструкций процедурного программирования;</p> <ul style="list-style-type: none"> <li>– основные типы данных, их назначение, ограничения, допустимые операции;</li> <li>– способы определения подпрограмм и передачи им параметров, ограничения на типы возвращаемых значений;</li> <li>– способы организации массивов;</li> <li>– приведение и вывод базовых типов данных;</li> <li>– модель работы с выделяемой для программы памятью: статический раздел, стек, куча;</li> <li>– динамическое выделение и освобождение памяти;</li> <li>– механизмы ссылок и указателей;</li> <li>– агрегатные (контейнерные) пользовательские типы данных (перечисления, структуры, массивы, строки, различные виды списков, словари, хэш-таблицы);</li> <li>– стандартные алгоритмы работы с агрегатными типами;</li> <li>– операции для работы с файлами входных и выходных данных (текстовыми и бинарными);</li> <li>– модули стандартных библиотек для выполнения операций структурного программирования;</li> <li>– примеры сторонних (инструментальных) библиотек и задач, которые могут быть решены с их помощью средствами языка программирования</li> </ul>	<p>документировать тестовые наборы;</p> <ul style="list-style-type: none"> <li>– представлять алгоритмы различными способами;</li> <li>– оценивать вычислительную (временную) и объёмную (пространственную) сложность алгоритмов;</li> <li>– осуществлять программную реализацию алгоритмов на выбранном языке программирования в среде разработки;</li> <li>– выполнять рефакторинг, оптимизацию производительности, реинжиниринг уже существующей программы;</li> <li>– осуществлять работу с базовыми и пользовательскими типами данных;</li> <li>– реализовывать алгоритмы чтения, обработки, записи данных во внешний файл;</li> <li>– создавать многомодульные проекты, подключать модули стандартной библиотеки, подключать сторонние библиотеки.</li> </ul>	<p>документировани я программ;</p> <p>навыками работы с отдельными типами данных.</p>		
--	---	--	---	---	--	--

## 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»)
<b>Полнота знаний</b>	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.
<b>Наличие умений</b>	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.
<b>Наличие навыков (владение опытом)</b>	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.
<b>Характеристика сформированности компетенции</b>	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 полностью и правильно. Отчет по лабораторной/практической работе подготовлен качественно в соответствии с требованиями. Полнота ответов на вопросы преподавателя при защите работы.
<i>Хорошо</i>	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.
<i>Удовлетворительно</i>	Задания выполнены частично с ошибками. Демонстрирует средний уровень выполнения задания на лабораторную/практическую работу. Большинство требований, предъявляемых к заданию, выполнены.
<i>Неудовлетворительно</i>	Задание выполнено со значительным количеством ошибок на низком уровне. Многие требования, предъявляемые к заданию, не выполнены. ИЛИ Задание не выполнено.

#### 3.2 Критерии и шкала оценивания тестирования

Перечень тестовых вопросов и заданий, описание процедуры тестирования представлены в методических материалах по освоению дисциплины (модуля) и в электронном курсе в ЭИОС МАУ. В ФОС включен типовой вариант тестового задания:

Какая из перечисленных инструкций приведёт к ошибке компиляции в случае версии языка C++14?

- 1) `int a = 0.5;`
- 2) `int a(0.5);`
- 3) `int a{0.5};`
- 4) ни одна из перечисленных.

Оценка/баллы	Критерии оценки
<i>Отлично</i>	90-100 % правильных ответов
<i>Хорошо</i>	70-89 % правильных ответов
<i>Удовлетворительно</i>	50-69 % правильных ответов
<i>Неудовлетворительно</i>	49% и меньше 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 зачетом

Если обучающийся набрал зачетное количество баллов согласно установленному диапазону по дисциплине (модулю), то он считается аттестованным с оценкой согласно шкале баллов для определения итоговой оценки:

Оценка	Баллы	Критерии оценивания
<i>Отлично</i>	91 - 100	Набрано зачетное количество баллов согласно установленному диапазону
<i>Хорошо</i>	81 - 90	Набрано зачетное количество баллов согласно установленному диапазону
<i>Удовлетворительно</i>	60 - 80	Набрано зачетное количество баллов согласно установленному диапазону
<i>Неудовлетворительно</i>	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

<b>ОПК-2. Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач</b>	
1	<p>Каким будет результат запуска утилиты <i>qmake</i> без параметров в директории проекта?</p> <ol style="list-style-type: none"> <li>1) создан файл проекта .pro;</li> <li>2) создан <i>Makefile</i>;</li> <li>3) выполнен скрипт в имеющемся <i>Makefile</i>;</li> <li>4) создан исполняемый модуль.</li> </ol>
2	<p>Какая утилита используется в <i>Qt</i> для компиляции файлов ресурсов?</p> <ol style="list-style-type: none"> <li>1) <i>cpp</i>;</li> <li>2) <i>uic</i>;</li> <li>3) <i>rcc</i>;</li> <li>4) <i>moc</i>.</li> </ol>
3	<p>Какое словосочетание не относится к описанию процесса отладки?</p> <ol style="list-style-type: none"> <li>1) <i>стражи включения</i>;</li> <li>2) точки останова;</li> <li>3) пошаговое выполнение;</li> <li>4) дампы памяти.</li> </ol>
4	<p>Какой из перечисленных терминов означает процесс разбора символьной последовательности по шаблону с целью поиска и извлечения определённой информации?</p> <ol style="list-style-type: none"> <li>1) сборка;</li> <li>2) аллоцирование;</li> <li>3) инициализация;</li> </ol>

	4) <i>парсинг</i>
5	<p><i>Что НЕ может содержаться в заголовочном файле?</i></p> <ol style="list-style-type: none"> <li>1) директивы включения;</li> <li>2) директивы условной компиляции;</li> <li>3) макроопределения;</li> <li>4) объявления типов;</li> <li>5) объявления функций;</li> <li>6) определения встраиваемых функций;</li> <li>7) объявление функции <code>main</code>;</li> <li>8) <i>определение функции <code>main</code></i>;</li> <li>9) объявления констант;</li> <li>10) определения пространств имён.</li> </ol>
6	<p><i>Как можно назвать функцию <code>main</code>, говоря о процессе передачи управления от операционной системы к программе, написанной на языке C++?</i></p> <ol style="list-style-type: none"> <li>1) контрольная точка;</li> <li>2) точка экстремума;</li> <li>3) <i>точка останова</i>;</li> <li>4) точка входа.</li> </ol>
7	<p><i>Какая из перечисленных инструкций недопустима в версиях языка, предшествующих C++14?</i></p> <ol style="list-style-type: none"> <li>1) <code>int a = 1;</code></li> <li>2) <code>int a(1);</code></li> <li>3) <code>int a{1};</code></li> <li>4) все инструкции допустимы.</li> </ol>
8	<p><i>Какая из перечисленных инструкций приведёт к ошибке компиляции в случае версии языка C++14?</i></p> <ol style="list-style-type: none"> <li>1) <code>int a = 0.5;</code></li> <li>2) <code>int a(0.5);</code></li> <li>3) <code>int a{0.5};</code></li> <li>4) ни одна из перечисленных.</li> </ol>
9	<p><i>Какие из следующих высказываний тождественно истинны (перечислите ВСЕ варианты) с учётом следующего определения: <code>bool b(10)</code>:</i></p> <ol style="list-style-type: none"> <li>1) <code>b == true;</code></li> <li>2) <code>b == false;</code></li> <li>3) <code>b == 10;</code></li> <li>4) <code>b == 1;</code></li> </ol>
10	<p><i>Какой из побитовых операторов позволяет получить целую часть от деления целого числа на 2 без использования других операторов?</i></p> <ol style="list-style-type: none"> <li>1) <code>&gt;&gt;</code>;</li> <li>2) <code>&amp;</code>;</li> <li>3) <code> </code>;</li> <li>4) <code>&lt;&lt;</code>.</li> </ol>
<b>ОПК-5. Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения</b>	
1	<p><i>Какой из следующих циклов является в C++ циклом с постусловием?</i></p> <ol style="list-style-type: none"> <li>1) <code>for</code>;</li> <li>2) <code>foreach</code>;</li> <li>3) <i>do-while</i>;</li> <li>4) <code>while</code>.</li> </ol>
2	<p><i>Какая подпрограмма на языке C++ наиболее точно соответствует процедуре языка Pascal?</i></p>

	<ol style="list-style-type: none"> <li>1) встраиваемая функция;</li> <li>2) функция, заданная макроопределением;</li> <li>3) функция со спецификатором <code>static</code>;</li> <li>4) функция с типом возвращаемого значения <code>void</code>;</li> <li>5) функция с пустым списком входных параметров.</li> </ol>
3	<p><i>В программе определены две функции:</i></p> <pre>int sum1 (int a, int b) {return a + b;} constexpr int sum 2(int a, int b){return a + b;}</pre> <p><i>Какая из перечисленных инструкций приведёт к ошибке компиляции:</i></p> <ol style="list-style-type: none"> <li>1) <code>int a1(sum2 (5, 12));</code></li> <li>2) <code>constexpr int a2(sum1 (5, 12));</code></li> <li>3) <code>int a3(sum2 (5, 12));</code></li> <li>4) <code>int a4(sum1 (5, 12));</code></li> <li>5) ни одна из инструкций не приведёт к ошибке.</li> </ol>
4	<p><i>Что такое rvalue?</i></p> <ol style="list-style-type: none"> <li>1) именованный объект;</li> <li>2) значение, доступное только в пределах выражения, где оно было использовано как литерал или вычислено;</li> <li>3) локальная копия переданного в функцию значения ей параметра;</li> <li>4) нулевой указатель.</li> </ol>
5	<p><i>Какой из перечисленных операторов имеет наивысший приоритет:</i></p> <ol style="list-style-type: none"> <li>1) постфиксный инкремент/декремент;</li> <li>2) префиксный инкремент/декремент;</li> <li>3) взятие адреса;</li> <li>4) унарный плюс/минус.</li> </ol>
6	<p><i>Имеется массив <code>a</code> (<code>int a[5]</code>), целочисленная переменная <code>i</code> и указатель <code>p</code>: <code>int* p(a)</code>. Какие из перечисленных высказываний тождественно истинны (перечислите ВСЕ варианты)?</i></p> <ol style="list-style-type: none"> <li>1) <code>p == a;</code></li> <li>2) <code>p == &amp;a[1];</code></li> <li>3) <code>a[i] == *(p + i);</code></li> <li>4) <code>a[i] == *(a + i).</code></li> </ol>
7	<p><i>Какие из следующих типов могут быть использованы для возвращаемого значения функции (перечислите ВСЕ варианты)</i></p> <ol style="list-style-type: none"> <li>1) <code>void</code>;</li> <li>2) <code>int</code>;</li> <li>3) <code>int*</code>;</li> <li>4) <code>int**</code>;</li> <li>5) <code>int&amp;</code>;</li> <li>6) <code>int[]</code>;</li> <li>7) <code>int[5]</code>.</li> </ol>
8	<p><i>Какой из перечисленных задач целесообразно использовать статическую переменную?</i></p> <ol style="list-style-type: none"> <li>1) подсчёт количества вызовов функции;</li> <li>2) выделение памяти фиксированного размера вне зависимости от компилятора;</li> <li>3) организация цикла со счётчиком;</li> <li>4) просмотр значений нестатических переменных в режиме отладки.</li> </ol>
9	<p><i>Рекурсия - это</i></p> <ol style="list-style-type: none"> <li>1) инициализация значением по умолчанию;</li> <li>2) исключительная ситуация, приводящая к завершению работы программы;</li> <li>3) вычислительный процесс, при котором функция вызывает саму себя;</li> <li>4) функция, которая возвращает значение типа <code>void</code>.</li> </ol>

10	<p><i>Рефакторинг - это</i></p> <ol style="list-style-type: none"><li>1) расширение функциональной спецификации программы;</li><li>2) оптимизация исходного кода по количеству используемой памяти;</li><li>3) оптимизация исходного кода по количеству операций;</li><li>4) <i>преобразование исходного кода с целью сделать его более понятным для программиста</i></li></ol>
----	---