

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования



**Пермский национальный исследовательский  
политехнический университет**

**УТВЕРЖДАЮ**

Проректор по образовательной  
деятельности

 И.Ю.Черникова

« 26 » сентября 20 24 г.

### **РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

**Дисциплина:** Языки и методы программирования 1  
(наименование)

**Форма обучения:** очная  
(очная/очно-заочная/заочная)

**Уровень высшего образования:** бакалавриат  
(бакалавриат/специалитет/магистратура)

**Общая трудоёмкость:** 144 (4)  
(часы (ЗЕ))

**Направление подготовки:** 01.03.02 Прикладная математика и информатика  
(код и наименование направления)

**Направленность:** Математическое моделирование (СУОС)  
(наименование образовательной программы)

## 1. Общие положения

### 1.1. Цели и задачи дисциплины

Основной целью курса является ознакомление будущих специалистов-математиков с программированием на языках высокого уровня для современных вычислительных машин и комплексов, принципами построения сложных программных систем. Создание сложных вычислительных комплексов требует знаний в областях, связанных с построением сложных алгоритмов, в сфере программной инженерии, поэтому в рамках курса студенты знакомятся с основами структурного и объектно-ориентированного программирования. Успешное освоение курса необходимо для решения прикладных инженерных задач в области расчетов прочности, исследования тепловых явлений и эффектов, задач мезомеханики, течения жидкостей, для выполнения курсовых и дипломных проектов.

Программа изучения дисциплины должна обеспечить приобретение знаний, умений и навыков в следующих областях:

- построение алгоритмов и их реализации на современных компьютерах;
- построение языков программирования, способов и механизмов управления данными;
- построение человеко-машинного интерфейса.

Изучение дисциплины обеспечивает:

- формирование знаний формальных способов описания алгоритмов, основных принципов построения алгоритмов решения прикладных задач, языков программирования высокого уровня, современных подходов к программированию, методов разработки программных систем;
- формирование умений разрабатывать прикладные программные продукты для решения задач по специальности с реализацией на алгоритмическом языке высокого уровня в средах разработки или средах визуального программирования;
- формирование навыков разработки алгоритмов решения прикладных задач с использованием современных языков и сред программирования.

### 1.2. Изучаемые объекты дисциплины

Алгоритмические системы, искусственные языки, современные языки программирования (C/C++), среды разработки (MS Visual Studio), компиляторы.

### 1.3. Входные требования

Не предусмотрены

## 2. Планируемые результаты обучения по дисциплине

Компетенция	Индекс индикатора	Планируемые результаты обучения по дисциплине (знать, уметь, владеть)	Индикатор достижения компетенции, с которым соотнесены планируемые результаты обучения	Средства оценки
ОПК-2	ИД-1ОПК-2	Умеет: – разрабатывать и использовать прикладные программные продукты для решения задач по специальности;	Умеет обосновывать выбор и применение современного математического аппарата и систем программирования в исследовательской и прикладной деятельности	Отчёт по практическому занятию

Компетенция	Индекс индикатора	Планируемые результаты обучения по дисциплине (знать, уметь, владеть)	Индикатор достижения компетенции, с которым соотнесены планируемые результаты обучения	Средства оценки
ОПК-2	ИД-2ОПК-2	Владеет: – навыками работы прикладного программиста и продвинутого пользователя ПК.	Владеет навыками применения современного математического аппарата и систем программирования при разработке и реализации алгоритмов решения прикладных задач	Индивидуальное задание
ОПК-2	ИД-3ОПК-2	Знает: – способы и методы разработки программных средств;	Знает современный математический аппарат, особенности применения современных математических методов и систем программирования в областях знаний, связанных с профессиональной деятельностью;	Контрольная работа
ОПК-4	ИД-1ОПК-4	Знает: – основные понятия языков программирования; синтаксис, семантику языков программирования; формальные способы описания алгоритмов; типы данных, способы и механизмы управления данными; методы и основные этапы трансляции; – хотя бы один язык программирования (C/C++).	Знает порядок постановки и распределения задач исполнителям работ и способы комбинирования существующих информационно-коммуникационных технологии для решения задач в области профессиональной деятельности	Коллоквиум
ОПК-4	ИД-2ОПК-4	Умеет: – программировать хотя бы на одном языке высокого уровня (C/C++);	Умеет определять необходимость комбинировать и адаптировать существующие информационно-коммуникационные технологии при решении задач в области профессиональной деятельности и учитывать требования информационной безопасности	Отчёт по практическому занятию
ОПК-4	ИД-3ОПК-4	Владеет: – навыками разработки алгоритмов решения	Владеет навыками использования существующих	Индивидуальное задание

Компетенция	Индекс индикатора	Планируемые результаты обучения по дисциплине (знать, уметь, владеть)	Индикатор достижения компетенции, с которым соотнесены планируемые результаты обучения	Средства оценки
		прикладных задач с использованием ПК; – навыками разработки прикладных программ на языках программирования высокого уровня и использования сред разработки (MS Visual Studio).	информационно-коммуникационных технологии для решения задач в области профессиональной деятельности	

### 3. Объем и виды учебной работы

Вид учебной работы	Всего часов	Распределение по семестрам в часах	
		Номер семестра	
		1	
1. Проведение учебных занятий (включая проведение текущего контроля успеваемости) в форме:	72	72	
1.1. Контактная аудиторная работа, из них:			
- лекции (Л)	16	16	
- лабораторные работы (ЛР)			
- практические занятия, семинары и (или) другие виды занятий семинарского типа (ПЗ)	34	34	
- контроль самостоятельной работы (КСР)	22	22	
- контрольная работа			
1.2. Самостоятельная работа студентов (СРС)	36	36	
2. Промежуточная аттестация			
Экзамен	36	36	
Дифференцированный зачет			
Зачет			
Курсовой проект (КП)			
Курсовая работа (КР)			
Общая трудоемкость дисциплины	144	144	

### 4. Содержание дисциплины

Наименование разделов дисциплины с кратким содержанием	Объем аудиторных занятий по видам в часах			Объем внеаудиторных занятий по видам в часах
	Л	ЛР	ПЗ	
1-й семестр				

Наименование разделов дисциплины с кратким содержанием	Объем аудиторных занятий по видам в часах			Объем внеаудиторных занятий по видам в часах
	Л	ЛР	ПЗ	СРС
Алгоритмизация	4	0	4	2
Тема 1. Понятие алгоритма и алгоритмической системы, сложность алгоритма. Способы описания алгоритма. Основные типы алгоритмов, их сложность и их использование для решения задач. Языки программирования: причины возникновения, классификация, основные понятия языков программирования. Тема 2. Искусственные и естественные языки. Методы и основные этапы трансляции. Синтаксис и семантика языка программирования. Этапы процесса создания исполняемого файла. Формальные способы описания языков программирования.				
Язык программирования высокого уровня (C/C++)	4	0	12	14
Тема 3. Алфавит, синтаксис и семантика языка программирования. Простейшая программа. Основные типы, операторы, конструкции языка. Бинарные и унарные операции. Приоритет операций. Логические операции, битовые операции. Тема 4. Структуры данных. Основные алгоритмы обработки данных (сортировка, поиск и др.).				
Структурный подход	4	0	10	10
Тема 5. Основные принципы структурного подхода к программированию. Причины появления и элементы структурного программирования. Тема 6. Синтаксическая поддержка структурного подхода, реализованная в языках программирования высокого уровня. Модульность. Процедуры, функции, модули. Файлы. Хранение данных. Операции с файлами.				
Объектно-ориентированный подход	4	0	8	10
Тема 7. Концепция объектно-ориентированного подхода к программированию. Основные принципы. Тема 8. Реализация объектно-ориентированного подхода в современных языках и средах разработки.				
ИТОГО по 1-му семестру	16	0	34	36
ИТОГО по дисциплине	16	0	34	36

#### Тематика примерных практических занятий

№ п.п.	Наименование темы практического (семинарского) занятия

№ п.п.	Наименование темы практического (семинарского) занятия
1	Построение алгоритмов решения прикладных задач с использованием ЭВМ. Способы описания алгоритма.
2	Базовый синтаксис алгоритмического языка C/C++. Структура программы. Виды трансляторов. Этапы создания исполняемого файла программы.
3	Основные типы, операторы, операции, конструкции алгоритмического языка.
4	Массивы данных и их обработка. Строковые типы данных, основные операции над ними.
5	Применение структурного подхода к программированию. Структурирование программы. Функции и процедуры и их применение.
6	Модульность. Создание и использование подключаемых модулей. Хранение данных и обмен данными: файлы, операции с файлами. Решение прикладных задач с использованием принципов структурного подхода к программированию.
7	Неоднородные структурированные типы данных: записи, классы, объекты. Описание класса. Создание и уничтожение объекта. Реализация основных принципов объектно-ориентированного программирования в алгоритмическом языке высокого уровня.
8	Разработка прикладных программ с использованием принципов объектно-ориентированного подхода к программированию.

## 5. Организационно-педагогические условия

### 5.1. Образовательные технологии, используемые для формирования компетенций

Проведение лекционных занятий по дисциплине основывается на активном методе обучения, при котором учащиеся не пассивные слушатели, а активные участники занятия, отвечающие на вопросы преподавателя. Вопросы преподавателя нацелены на активизацию процессов усвоения материала, а также на развитие логического мышления. Преподаватель заранее намечает список вопросов, стимулирующих ассоциативное мышление и установление связей с ранее освоенным материалом.

Практические занятия проводятся на основе реализации метода обучения действием: определяются проблемные области, формируются группы. При проведении практических занятий преследуются следующие цели: применение знаний отдельных дисциплин и креативных методов для решения проблем и принятия решений; отработка у обучающихся навыков командной работы, межличностных коммуникаций и развитие лидерских качеств; закрепление основ теоретических знаний.

Проведение лабораторных занятий основывается на интерактивном методе обучения, при котором обучающиеся взаимодействуют не только с преподавателем, но и друг с другом. При этом доминирует активность учащихся в процессе обучения. Место преподавателя в интерактивных занятиях сводится к направлению деятельности обучающихся на достижение целей занятия.

При проведении учебных занятий используются интерактивные лекции, групповые дискуссии, ролевые игры, тренинги и анализ ситуаций и имитационных моделей.

## 5.2. Методические указания для обучающихся по изучению дисциплины

При изучении дисциплины обучающимся целесообразно выполнять следующие рекомендации:

1. Изучение учебной дисциплины должно вестись систематически.
2. После изучения какого-либо раздела по учебнику или конспектным материалам рекомендуется по памяти воспроизвести основные термины, определения, понятия раздела.
3. Особое внимание следует уделить выполнению отчетов по практическим занятиям, лабораторным работам и индивидуальным комплексным заданиям на самостоятельную работу.
4. Вся тематика вопросов, изучаемых самостоятельно, задается на лекциях преподавателем. Им же даются источники (в первую очередь вновь изданные в периодической научной литературе) для более детального понимания вопросов, озвученных на лекции.

## 6. Перечень учебно-методического и информационного обеспечения для самостоятельной работы обучающихся по дисциплине

### 6.1. Печатная учебно-методическая литература

№ п/п	Библиографическое описание (автор, заглавие, вид издания, место, издательство, год издания, количество страниц)	Количество экземпляров в библиотеке
<b>1. Основная литература</b>		
1	Бизли Д. Python. Исчерпывающее руководство : пер. с англ. Санкт-Петербург [и др.] : Питер, 2023. 366 с. 29,670 усл. печ. л.	1
2	Гордиенко А. П. Функциональное программирование : учебник. Москва : КНОРУС, 2023. 277 с. 17,5 усл. печ. л.	1
3	Лоспинозо Д. С++ для профи. Молниеносный старт : пер. с англ. Санкт-Петербург [и др.] : Питер, 2021. 815 с. 67,080 усл. печ. л.	1
4	Павловская Т. А. С/С++. Программирование на языке высокого уровня : для магистров и бакалавров учебник для вузов. Санкт-Петербург [и др.] : Питер, 2020. 460 с. 37,410 усл. печ. л.	50
5	Павловская Т. А. С/С++. Программирование на языке высокого уровня : для магистров и бакалавров учебник для вузов. Санкт-Петербург [и др.] : Питер, 2021. 460 с. 37,410 усл. печ. л.	50
6	Парфенов Д. В. Язык Си: кратко и ясно : учебное пособие. Москва : ИНФРА-М, 2024. 320 с. 20,0 усл. печ. л.	1
7	Страуструп Б. Язык программирования С++ : пер. с англ. Спец. изд. Москва : БИНОМ, 2012. 1135 с. 92,3 усл. печ. л.	6
8	Шилдт Г. Полный справочник по С++ : пер. с англ. 4-е изд. Москва [и др.] : Вильямс, 2012. 796 с. 64,5 усл. печ. л.	2
<b>2. Дополнительная литература</b>		
<b>2.1. Учебные и научные издания</b>		
1	Новиков Ф. А. Дискретная математика для программистов : учебное пособие. Санкт-Петербург : Питер, 2004. 301 с.	8
<b>2.2. Периодические издания</b>		
	Не используется	
<b>2.3. Нормативно-технические издания</b>		
	Не используется	
<b>3. Методические указания для студентов по освоению дисциплины</b>		
	Не используется	

4. Учебно-методическое обеспечение самостоятельной работы студента		
1	Введение в математическое моделирование : учебное пособие / Ашихмин В. Н., Гитман М. Б., Келлер И. Э., Наймарк О.Б., Столбов В. Ю., Трусов П. В., Фрик П.Г. Москва : Логос, 2007. 439 с.	2
2	Введение в математическое моделирование : учебное пособие для вузов / Ашихмин В.Н., Гитман М.Б., Келлер И.Э., Наймарк О.Б. Москва : Логос, 2005. 439 с.	31

## 6.2. Электронная учебно-методическая литература

Вид литературы	Наименование разработки	Ссылка на информационный ресурс	Доступность (сеть Интернет / локальная сеть; авторизованный / свободный доступ)
Учебно-методическое обеспечение самостоятельной работы студентов	Введение в математическое моделирование: Учеб. пособие / Под ред. П.В. Трусова. — М.: Университетская книга, Логос, 2007. - 440 с. ISBN 978-5-98704-037-X.	<a href="http://www.mmsp.pstu.ru/userfiles/Posobie/MM_intro_PVTrusov.pdf">http://www.mmsp.pstu.ru/userfiles/Posobie/MM_intro_PVTrusov.pdf</a>	сеть Интернет; свободный доступ
Учебно-методическое обеспечение самостоятельной работы студентов	Документация по языку C++	<a href="https://cplusplus.com/">https://cplusplus.com/</a>	сеть Интернет; свободный доступ
Учебно-методическое обеспечение самостоятельной работы студентов	Парфёнов Д. В. Объектно-ориентированные средства математического моделирования. Москва : РТУ МИРЭА, 2021. 145 с.	<a href="https://e.lanbook.com/book/26556">https://e.lanbook.com/book/26556</a>	сеть Интернет; авторизованный доступ
Учебно-методическое обеспечение самостоятельной работы студентов	Парфенов, Д. В. Промышленное программирование с использованием языка C++ : учебное пособие / Д. В. Парфенов, Д. А. Петрусевич. — Москва : РТУ МИРЭА, 2023. — 131 с. — ISBN 978-5-7339-1708-5. — Текст : электронный // Лань : электронно-библиотечная система.	<a href="https://e.lanbook.com/book/31619">https://e.lanbook.com/book/31619</a>	сеть Интернет; авторизованный доступ
Учебно-методическое обеспечение самостоятельной работы студентов	Сайт с описанием алгоритмов	<a href="http://algotlist.manual.ru">http://algotlist.manual.ru</a>	сеть Интернет; свободный доступ



Вид литературы	Наименование разработки	Ссылка на информационный ресурс	Доступность (сеть Интернет / локальная сеть; авторизованный / свободный доступ)
Учебно-методическое обеспечение самостоятельной работы студентов	Страуструп Б. Язык программирования С++ для профессионалов. 2-е изд. Москва : ИНТУИТ, 2016. 670 с.	<a href="https://e.lanbook.com/book/100542">https://e.lanbook.com/book/100542</a>	сеть Интернет; авторизованный доступ
Учебно-методическое обеспечение самостоятельной работы студентов	Хаггарти Р. Дискретная математика для программистов. 2-е изд., испр. Москва : Техносфера, 2012. 40 с.	<a href="https://e.lanbook.com/book/73011">https://e.lanbook.com/book/73011</a>	сеть Интернет; авторизованный доступ

### 6.3. Лицензионное и свободно распространяемое программное обеспечение, используемое при осуществлении образовательного процесса по дисциплине

Вид ПО	Наименование ПО
Операционные системы	MS Windows 11 (подп. Azure Dev Tools for Teaching )
Офисные приложения.	Microsoft Office Professional 2007. лиц. 42661567
Прикладное программное обеспечение общего назначения	Dr.Web Enterprise Security Suite, 3000 лиц, ПНИПУ ОЦНИТ 2017
Среды разработки, тестирования и отладки	Eclipse (Open Source, лиц. EPL)
Среды разработки, тестирования и отладки	jet brains pycharm ( Apache License 2.0 )
Среды разработки, тестирования и отладки	MS Visual studio 2019 community (Free)

### 6.4. Современные профессиональные базы данных и информационные справочные системы, используемые при осуществлении образовательного процесса по дисциплине

Наименование	Ссылка на информационный ресурс
База данных научной электронной библиотеки (eLIBRARY.RU)	<a href="https://elibrary.ru/">https://elibrary.ru/</a>
Научная библиотека Пермского национального исследовательского политехнического университета	<a href="https://elib.pstu.ru/">https://elib.pstu.ru/</a>
Образовательная платформа Юрайт	<a href="https://urait.ru/">https://urait.ru/</a>
Электронно-библиотечная система Лань	<a href="https://e.lanbook.com/">https://e.lanbook.com/</a>
Электронно-библиотечная система IPRsmart	<a href="http://www.iprbookshop.ru/">http://www.iprbookshop.ru/</a>
Информационные ресурсы Сети КонсультантПлюс	локальная сеть

## **7. Материально-техническое обеспечение образовательного процесса по дисциплине**

Вид занятий	Наименование необходимого основного оборудования и технических средств обучения	Количество единиц
Лекция	Аудитории, оборудованные ноутбуком, видеопроектором	1
Практическое занятие	Персональные компьютеры (локальная компьютерная сеть)	12

## **8. Фонд оценочных средств дисциплины**

Описан в отдельном документе

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
**«Пермский национальный исследовательский политехнический  
университет»**

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**  
для проведения промежуточной аттестации обучающихся по дисциплине  
**«Языки и методы программирования 1»**

*Приложение к рабочей программе дисциплины*

**Направление подготовки:** 01.03.02 Прикладная математика и  
информатика

Пермь 2024

**Фонд оценочных средств** для проведения промежуточной аттестации обучающихся по дисциплине является частью (приложением) к рабочей программе дисциплины. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине разработан в соответствии с общей частью фонда оценочных средств для проведения промежуточной аттестации основной образовательной программы, которая устанавливает систему оценивания результатов промежуточной аттестации и критерии выставления оценок. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине устанавливает формы и процедуры текущего контроля успеваемости и промежуточной аттестации обучающихся по дисциплине.

Предусмотрены аудиторные лекционные и практические занятия, а также самостоятельная работа студентов. В рамках освоения учебного материала дисциплины формируются компоненты компетенций знать, уметь, владеть, указанные в РПД, которые выступают в качестве контролируемых результатов обучения по дисциплине.

Контроль уровня усвоенных знаний, усвоенных умений и приобретенных владений осуществляется в рамках текущего, рубежного и промежуточного контроля при изучении теоретического материала и в ходе практических занятий, а также на экзамене. Итоговой оценкой достижения результатов обучения по дисциплине является промежуточная аттестация в виде экзамена, проводимая с учетом результатов текущего и рубежного контроля.

### **1. Виды контроля, типовые контрольные задания и шкалы оценивания результатов обучения**

Текущий контроль успеваемости имеет целью обеспечение максимальной эффективности учебного процесса, управление процессом формирования заданных компетенций обучаемых, повышение мотивации к учебе и предусматривает оценивание хода освоения дисциплины. В соответствии с Положением о проведении текущего контроля успеваемости и промежуточной аттестации обучающихся по образовательным программам высшего образования – программам бакалавриата, специалитета и магистратуры в ПНИПУ предусмотрены следующие виды и периодичность текущего контроля успеваемости обучающихся:

- входной контроль, проверка исходного уровня подготовленности обучаемого и его соответствия предъявляемым требованиям для изучения данной дисциплины;

- текущий контроль усвоения материала (уровня освоения компонента «знать» заданных компетенций) на каждом групповом занятии и контроль посещаемости лекционных занятий;

- промежуточный и рубежный контроль освоения обучаемыми отдельных компонентов «знать», «уметь» заданных компетенций путем компьютерного или бланочного тестирования, контрольных опросов, контрольных работ (индивидуальных домашних заданий), защиты отчетов по лабораторным работам, рефератов, эссе и т.д.

Рубежный контроль по дисциплине проводится на следующей неделе после прохождения модуля дисциплины, а промежуточный – во время каждого контрольного мероприятия внутри модулей дисциплины;

- межсессионная аттестация, единовременное подведение итогов текущей успеваемости не менее одного раза в семестр по всем дисциплинам для каждого направления подготовки (специальности), курса, группы;

- контроль остаточных знаний.

### **2.1. Текущий контроль усвоения материала**

Текущий контроль усвоения материала в форме собеседования или выборочного теоретического опроса студентов проводится по каждой теме. Результаты по 4-балльной шкале оценивания заносятся в книжку преподавателя и учитываются в виде интегральной оценки при проведении промежуточной аттестации.

### **2.2. Рубежный контроль**

Рубежный контроль для комплексного оценивания усвоенных знаний, освоенных умений и приобретенных владений проводится в форме защиты практических занятий и рубежных контрольных работ (после изучения каждого модуля учебной дисциплины).

#### **2.2.1. Защита лабораторных и практических занятий**

Всего запланировано 17 практических занятий. Типовые темы практических занятий приведены в РПД.

Защита практического занятия проводится индивидуально каждым студентом или группой студентов. Типовые шкала и критерии оценки приведены в общей части ФОС образовательной программы.

#### **2.2.2. Рубежная контрольная работа**

Согласно РПД запланировано 2 рубежные контрольные работы (КР) после освоения студентами учебных модулей дисциплины. Первая КР по модулям 1 и 2 «Алгоритмизация и язык программирования высокого уровня (C/C++)», вторая КР – по модулям 3 и 4 «Современные подходы к программированию: структурный подход, объектно-ориентированный подход».

Типовые шкала и критерии оценки результатов рубежной контрольной работы приведены в общей части ФОС образовательной программы.

### **2.3. Промежуточная аттестация (итоговый контроль)**

Допуск к промежуточной аттестации осуществляется по результатам текущего и рубежного контроля. Условиями допуска являются успешная сдача всех лабораторных работ и положительная интегральная оценка по результатам текущего и рубежного контроля.

Промежуточная аттестация, согласно РПД, проводится в виде экзамена по дисциплине устно по билетам. Билет содержит теоретические вопросы (ТВ) для проверки усвоенных знаний, практические задания (ПЗ) для проверки освоенных умений и комплексные задания (КЗ) для контроля уровня приобретенных владений всех заявленных компетенций.

Билет формируется таким образом, чтобы в него попали вопросы и

практические задания, контролирующие уровень сформированности *всех* заявленных компетенций. Форма билета представлена в общей части ФОС образовательной программы.

### **2.3.2. Шкалы оценивания результатов обучения на экзамене**

Оценка результатов обучения по дисциплине в форме уровня сформированности компонентов *знать, уметь, владеть* заявленных компетенций проводится по 4-х балльной шкале оценивания путем выборочного контроля во время экзамена.

Типовые шкала и критерии оценки результатов обучения при сдаче экзамена для компонентов *знать, уметь и владеть* приведены в общей части ФОС образовательной программы.

## **3. Критерии оценивания уровня сформированности компонентов и компетенций**

### **3.1. Оценка уровня сформированности компонентов компетенций**

При оценке уровня сформированности компетенций в рамках выборочного контроля при экзамене считается, что *полученная оценка за компонент проверяемой в билете компетенции обобщается на соответствующий компонент всех компетенций, формируемых в рамках данной учебной дисциплины.*

Типовые критерии и шкалы оценивания уровня сформированности компонентов компетенций приведены в общей части ФОС образовательной программы.

### **3.2. Оценка уровня сформированности компетенций**

Общая оценка уровня сформированности всех компетенций проводится путем агрегирования оценок, полученных студентом за каждый компонент формируемых компетенций, с учетом результатов текущего и рубежного контроля в виде интегральной оценки по 4-х балльной шкале. Все результаты контроля заносятся в оценочный лист и заполняются преподавателем по итогам промежуточной аттестации.

Форма оценочного листа и требования к его заполнению приведены в общей части ФОС образовательной программы.

При формировании итоговой оценки промежуточной аттестации в виде экзамена используются типовые критерии, приведенные в общей части ФОС образовательной программы.

## ЗАДАНИЯ ПО ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЕ

Правильный ответ	Содержание вопроса	Компетенция
модернизация <b>инкапсуляция</b> инициализация монетизация идентификация родительский контроль антропоморфизм <b>наследование</b> изоморфизм <b>полиморфизм</b>	Выберите три основных принципа объектно-ориентированного подхода к программированию.	ОПК-2
транслятор	Системная программа, переводящая текст программы на искусственном языке в текст эквивалентной программы в машинных кодах, называется ...	ОПК-2
интерпретация, компиляция	Существует два принципиально различных метода трансляции, назовите их.  <i>В ответе методы следует указать через запятую в алфавитном порядке, слова разделять пробелом.</i>	ОПК-2
Разветвляющийся алгоритм / ветвление	Как называется алгоритм вычислительного процесса, в рамках которого после проверки условия в разных ситуациях исполняются разные наборы команд (инструкций)?	ОПК-2
цикла	Дополните утверждение: «Согласно теоремам структурного программирования любой алгоритм может быть представлен при помощи операций присваивания, ветвления и ... ».	ОПК-2
Гвидо Ван Россум <b>Деннис Ритчи</b> Билл Гейтс Ада Лавлейс	Одним из создателей языка программирования С был(а) ...	ОПК-2
#define	Запишите название директивы препроцессора, которая указывает правила замены в тексте программы.  <i>Название директивы следует записать начиная с символа #.</i>	ОПК-4
#include	Запишите название директивы препроцессора, которая позволяет вставить в текст программы описания из указанного заголовочного файла.  <i>Название директивы следует записать начиная с символа #.</i>	ОПК-4
компоновки	Подключение кодов библиотеки осуществляется на этапе ..., т. е. после компиляции.  <i>Какое слово пропущено? Укажите его в качестве ответа.</i>	ОПК-2
переменная <b>константа</b>	Лексема, представляющая изображение фиксированного числового, строкового или	ОПК-2

функция директива	символьного значения – это ...	
int bool <b>const</b> main double	Какое зарезервированное слово используется для объявления константы в тексте программы	ОПК-2
long, short, signed, unsigned	В языке C++ существует четыре спецификатора типа, уточняющих внутреннее представление и диапазон значений стандартных типов. Назовите их.  <i>В ответе спецификаторы следует указать через запятую в алфавитном порядке, слова разделять пробелом.</i>	ОПК-4
мантисса, порядок	Внутреннее представление вещественного числа в ЭВМ состоит из двух частей. Каких?  <i>В ответе части следует указать через запятую в алфавитном порядке, слова разделять пробелом.</i>	ОПК-2
{ }	Группа операторов, составляющая некоторое целое (тело цикла, варианты у оператора <b>if</b> , тело функции и т.д.), выделяется в C++ с помощью этих двух символов. Укажите открывающий и закрывающий символы через пробел.	ОПК-2
namespace	Ключевое слово ... позволяет разделить глобальное пространство имен путем создания некоторой декларативной области (пространство имен определяет область видимости).	ОПК-2
1 0 3 2 5 4 7 6 9 8 10	В программе описан одномерный целочисленный массив с индексами от 0 до 10 и целочисленные переменные k, i. В приведенном ниже фрагменте программы массив сначала заполняется, а потом изменяется:  <pre>for (int i = 0; i &lt;= 10; i++) {     A[i] = i; } for (int i = 0; i &lt;= 4; i++) {     k = A[2*i];     A[2*i] = A[2*i+1];     A[2*i+1] = k; }</pre> <i>В ответе запишите все элементы массива через пробел.</i>	ОПК-4
124875487	Ниже записан рекурсивный алгоритм F.  <pre>void F(int n) {     std::cout &lt;&lt; n;     if (n &lt; 5) {         F(2 * n);     } }</pre>	ОПК-4



	<pre> F(n + 3); } } </pre> <p>Запишите подряд без пробелов и разделителей все числа, которые будут показаны на экране при выполнении вызова F(1). Числа должны быть записаны в том же порядке, в котором они выводятся на экран.</p>	
-143	<p>Определите значение переменной c после выполнения следующего фрагмента программы:</p> <pre> a = 5; a = 12 - a * a; b = -a; c = 10 * a - b; </pre>	ОПК-4
14	<p>Определите значение переменной c после выполнения следующего фрагмента программы. Ответ запишите в виде целого числа.</p> <pre> a = 28; b = 10; a = 3 * b - a; if (a &gt; b)     c = 2 * a - b; else     c = 2 * a + b; </pre>	ОПК-4
27	<p>В программе используется одномерный целочисленный массив A с индексами от 0 до 10. Ниже представлен фрагмент этой программы.</p> <pre> s = 0; n = 10; for (int i = 0; i &lt; n; i++) {     s = s + A[i] - A[i+1]; } </pre> <p>В начале выполнения этого фрагмента в массиве находились числа 27, 17, 7, 0, 7, 17, 27, 17, 10, 7, 0, т.е. A[0] = 27, A[1] = 17 и т.д.</p> <p>Чему будет равно значение переменной s после выполнения данного фрагмента программы?</p>	ОПК-4
4	<p>Элементы двумерного массива A размером 10x10 первоначально были равны 1. Затем значения некоторых из них меняют с помощью следующего фрагмента программы:</p> <pre> for (int n = 1; n &lt;= 4; n++) {     for (int k = 1; k &lt;= n + 1; k++){         A[n][k]= A[n][k] - 1;         A[n][k+1]= A[n][k] - 1;     } } </pre>	ОПК-4

	<pre> } } </pre> <p>Сколько элементов массива в результате будут равны 0?</p>	
17	<p>Напишите в ответе число, которое будет выведено в результате выполнения следующего алгоритма</p> <pre> #include &lt;iostream&gt; using namespace std;  long F(long x) {     return 2 * (x * x - 100) * (x * x - 100) + 5; }  int main() {     long a, b, t, M, R;     a = -20; b = 20;     M = a; R = F(a);     for (t = a; t &lt;= b; t++) {         if (F(t) &lt; R) {             M = t; R = F(t);         }     }     cout &lt;&lt; M + 27 &lt;&lt; endl;     return 0; } </pre>	ОПК-4
4	<p>Определите, какое число будет напечатано в результате выполнения следующего алгоритма:</p> <pre> #include &lt;iostream&gt; using namespace std;  int F(int x) { return -4*(x+4)*(x+2); }  int main() { int a, b, t, M, R; a = -20; b = 20; M = a; R = F(a); for (t = a; t &lt;= b; t++) { if (F(t) &gt; R) { M = t; R = F(t); } } cout &lt;&lt; R &lt;&lt; endl; } </pre>	ОПК-4

19	<p>Ниже записана программа, которая вводит натуральное число <math>x</math>, выполняет преобразования, а затем выводит два числа. Укажите наименьшее возможное значение <math>x</math>, при вводе которого программа выведет сначала 3, а потом 2.</p> <pre>#include &lt;iostream&gt; using namespace std; int main() {     int x, a, b;     cin &gt;&gt; x;     a = 0; b = 0;     while (x &gt; 0) {         if (x % 2 &gt; 0) ++a;         else ++b;         x = x / 2;     }     cout &lt;&lt; a &lt;&lt; ' ' &lt;&lt; b &lt;&lt; endl;     return 0; }</pre>	ОПК-4
5	<p>В какой (по счету) строчке программы допущена ошибка?</p> <pre>#include &lt;iostream&gt; using namespace std; int main() {     cout &gt;&gt; "Hello, World" &lt;&lt; '\n'; }</pre>	ОПК-4
<p>сортировка вставками сортировка обменом <b>сортировка Хоара</b> сортировка выбором</p>	<p>Из перечисленных алгоритмов сортировки выберите тот, который имеет вычислительную сложность <math>O(n \log n)</math></p>	ОПК-2
<p>12341 <b>189.07</b> <b>573455567456</b> -278 0</p>	<p>Выберите из предложенных вариантов числа, которые нельзя сохранить в типе <code>int</code>?</p>	ОПК-4
<p><b>пузырьком</b> топориком <b>вставками</b> <b>выбором</b> переворотом сдвигом</p>	<p>Выберите из предложенных вариантов названия алгоритмов квадратичных сортировок.</p>	ОПК-2
<p>long long</p>	<p>Как называется тип данных в языке программирования C++, как правило, занимающий 8 байт памяти и хранящий в себе «очень длинное целое число»?</p>	ОПК-2
15	<p>Дан массив <math>a = \{28, 37, 6, 4, 2, 9, 15, 52, 59, 15\}</math>, <math>n</math> – размер массива. Далее был запущен алгоритм:</p>	ОПК-4

	<pre>for (int i = 0; i &lt; n - 1; i++) {     for (int j = 0; j &lt; n - 1; j++){         if (a[j] &gt; a[j + 1]) {             swap(a[j], a[j + 1]);         }     } }</pre> <p>Каким будет пятый (по счету) элемент полученного массива?</p>	
--	--	--