

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**Федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Нижегородский государственный университет
им. Н.И. Лобачевского»**

Институт информационных технологий, математики и механики

(факультет / институт / филиал)

УТВЕРЖДЕНО

решением президиума Ученого совета ННГУ

протокол от

«16» июня 2021 г. № 8

Рабочая программа дисциплины

**Информатика и
программирование**

(наименование дисциплины (модуля))

Уровень высшего образования

бакалавриат

(бакалавриат / магистратура / специалитет)

Направление подготовки / специальность

09.03.04 Программная инженерия

(указывается код и наименование направления подготовки / специальности)

Направленность образовательной программы

Разработка программно-информационных систем

(указывается профиль / магистерская программа / специализация)

Форма обучения

очная

(очная / очно-заочная / заочная)

Нижегород

2021 год

1. Место дисциплины в структуре ОПОП

Дисциплина относится к обязательной части

№ варианта	Место дисциплины в учебном плане образовательной программы	Стандартный текст для автоматического заполнения в конструкторе РПД
1	Блок 1. Дисциплины (модули) Обязательная часть	Дисциплина <i>Б1.О.10 Информатика и программирование</i> относится к обязательной части ООП направления подготовки 09.03.04. Программная инженерия.

2. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции (код, содержание индикатора)	Результаты обучения по дисциплине	
ОПК-2 Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	ОПК-2.1. Знает современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности.	Знать технологии разработки программного обеспечения (структурное и модульное программирование – подробно, объектно-ориентированное программирование – на уровне общих идей).	Собеседование тест
	ОПК-2.2. Умеет выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности.	Уметь разрабатывать и отлаживать программы средней сложности с использованием технологий структурного, модульного и объектно-ориентированного программирования, применять информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности.	Практические задания, Проект
	ОПК-2.3. Имеет навыки применения современных информационных технологий и программных средств, в том числе	Владеть современными интегрированными средами разработки программ, в том числе отечественного производства, при решении задач профессиональной деятельности.	Практические задания, Проект

	<i>отечественного производства, при решении задач профессиональной деятельности.</i>		
<p>ОПК-6</p> <p><i>Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов</i></p>	<p>ОПК-6.1.</p> <p><i>Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий.</i></p>	<p><i>основные идеи, лежащие в основе современных языков программирования и основные составляющие современного языка программирования на примере C и C++.</i></p> <p><i>32 принципы реализации технологий разработки программного обеспечения: структурное, модульное и объектно-ориентированное программирование – в языках программирования C и C++;</i></p> <p><i>33 методы конструирования новых типов данных (C, C++).</i></p> <p><i>34 методы и приемы работы с динамической памятью (C, C++).</i></p> <p><i>35 методы работы с внешней памятью (C, C++).</i></p>	<p><i>Собеседование тест</i></p>
	<p>ОПК-6.2.</p> <p><i>Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ.</i></p>	<p><i>Уметь разрабатывать и отлаживать программы с использованием языков программирования C, C++ и базами данных,</i></p> <p><i>применять современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ.</i></p>	<p><i>Практические задания, Проект</i></p>
	<p>ОПК-6.3.</p> <p><i>Имеет навыки программирования, отладки и тестирования прототипов программно-технических комплексов задач.</i></p>	<p><i>Владеть навыками реализации, тестирования и отладки программных систем средней сложности (на базовом уровне).</i></p>	<p><i>Практические задания, Проект</i></p>
<p>ПК-4. <i>Способен применять методологии и средствами разработки программного обеспечения</i></p>	<p>ПК-4.1. <i>Знает современные методологии и технологии разработки ПО</i></p>	<p><i>Знать основные методологии и технологии разработки ПО</i></p>	<p><i>Сбеседование</i></p>
	<p>ПК-4.2. <i>Умеет использовать современные технологии и средства</i></p>	<p><i>Уметь применять технологии и средства разработки ПО</i></p>	<p><i>Практическое задание</i></p>

	разработки ПО		
	ПК-4.3. Имеет навыки использования современных технологий разработки ПО	Владеть навыками работы в интегрированной среде разработки ПО	Практическое задание

3. Структура и содержание дисциплины

3.1. Трудоемкость дисциплины

	Очная форма обучения
Общая трудоемкость	12 ЗЕТ
Часов по учебному плану	432
в том числе	
аудиторные занятия (контактная работа):	229
- занятия лекционного типа	96
- занятия семинарского типа	64
- занятия лабораторного типа	64
- текущий контроль (КСР)	5
самостоятельная работа	131
Промежуточная аттестация – экзамен	72

3.2. Содержание дисциплины

Наименование и краткое содержание разделов и тем дисциплины	Всего (часы)			В том числе																
				Контактная работа (работа во взаимодействии с преподавателем), часы												Самостоятельная работа обучающегося, часы				
	из них																			
	Занятия лекционного типа			Занятия семинарского типа			Занятия лабораторного типа						Всего							
Очная	Очно-заочная	Заочная	Очная	Очно-заочная	Заочная	Очная	Очно-заочная	Заочная	Очная	Очно-заочная	Заочная				Очная	Очно-заочная	Заочная	Очная	Очно-заочная	Заочная
Семестр 1																				
Введение в предмет.	6			2												2			4	

Структура и содержание курса. Литература.																				
Решение задач с использованием вычислительной техники.	12		4			2			2						8			4		
Современная система разработки программного обеспечения.	12		4			2			2						8			4		
Среда исполнения программ. Программа в среде Microsoft Windows.	10		2			2			2						6			4		
Основные понятия языков программирования. Синтаксис, семантика, формальные способы описания языков программирования.	14		4			3			3						10			4		
Типы данных, способы и механизмы управления данными.6	14		4			3			3						10			4		
Программа на языке С. Методы и основные этапы трансляции.	16		4			3			3						10			6		
Структурное программирование и операторы языка С.	16		4			3			3						10			6		
Конструирование новых типов данных.	16		4			3			3						10			6		
Модульное программирование	16		4			3			3						10			6		

ие.																				
Элементы анализа и разработки алгоритмов.	16			4			3			3						10			6	
Методы работы с внешней памятью. Файлы.	16			4			3			3						10			6	
Динамическое управление памятью.	14			4			2			2						8			6	
Текущий контроль (КСР)	2															2				
Промежуточная аттестация Экзамен	36																			
итого	216			48			32			32						114			66	
Семестр 2																				
Отличия языка C++ от языка C	12			3			2			2						7			4	
Введение в объектно-ориентированное программирование	9			4												4			4	
Классы и объекты	14			4			3			3						10			4	
Конструкторы и деструктор	14			4			3			3						10			4	
Обработка исключений	15			4			3			3						10			4	
Перегрузка операций	15			4			3			3						10			5	
Пример. Класс Cdate	13			3			2			2						7			5	
Наследование и иерархия классов	15			4			3			3						10			5	
Специальные поля и методы классов	15			4			3			3						10			6	
Виртуальные методы.	24			4			3			3						10			6	

Абстрактные виртуальные методы и классы																				
Шаблоны. Шаблоны функций и шаблоны классов	23		4			3			3						10			6		
Пример. Класс Container	23		3			2			2						7			6		
Пример. Класс Dictionary	21		3			2			2						7			6		
	213		48			32			32						112			65		
Текущий контроль (КСР)	3														3					
Промежуточная аттестация – экзамен	36																			
Итого	216		48			32			32						115			65		

Практические занятия (семинарские занятия) организуются, в том числе в форме практической подготовки, которая предусматривает участие обучающихся в выполнении отдельных элементов работ, связанных с будущей профессиональной деятельностью.

Практическая подготовка предусматривает: разработку программного обеспечения (приложений)

На проведение практических занятий (семинарских занятий) в форме практической подготовки отводится 30 часов.

Практическая подготовка направлена на формирование и развитие:

практических навыков в соответствии с профилем ОП:

- Формирование требований к информатизации и автоматизации прикладных процессов, формализация предметной области проекта; технико-экономическое обоснование проектных решений и составление технического задания на разработку программного продукта; проектирование программно- аппаратных средств в соответствии с техническим заданием; применение современных инструментальных средств при разработке программного обеспечения; документирование компонентов информационной системы на стадии жизненного цикла
- компетенций – ОПК-2, ОПК-6, ПК-4.

Текущий контроль успеваемости реализуется в формах опросов на занятиях семинарского типа, лабораторного типа.

Промежуточная аттестация проходит в традиционных формах (зачет, экзамен).

4. Учебно-методическое обеспечение самостоятельной работы обучающихся

Самостоятельная работа студента при изучении дисциплины «Информатика и программирование» включает выполнение практических заданий контролем преподавателя и подготовку к зачету и экзамену.

Тематика практических и лабораторных занятий (Семестр 1)

1. Знакомство. Обзор содержания курса и плана практики. Отчетность. Литература.
2. Введение в языки программирования. Языки высокого/низкого уровня. Понятия программы, переменной, способа интерпретации, типа данных. Тривиальная программа “Hello, world” на C.

3. Структура программы на C, синтаксис. Операторы ввода-вывода. Типы данных. Операторы ветвления. Программы:

- a. Определить сумму налога за участок треугольной формы (задаётся ставка налога).
- b. Определить вид треугольника по заданным длинам сторон. Обеспечение контроля ввода.
- c. С.Р. Определить взаимное расположение квадрата и круга. Даны длина стороны квадрата и радиусу круга. Центры круга и квадрата совпадают (5 вариантов взаимного расположения).

ТК. Изучение среды разработки VisualStudio. Простые программы:

- вычисление корней квадратного уравнения по заданным коэффициентам;
- реализовать программу, которая выводит цифры числа (замечание: предполагается, что вводится число с фиксированным количеством знаков).

С.Р. в Т/К. Определить взаимное расположение двух окружностей. Даны координаты центров и радиусы окружностей.

4. Обсуждение результатов С.Р. Обсуждение правильного использования операторов if и if...else. Оператор множественного выбора. Циклы. Контроль ввода (циклы с постусловием). Программы:

- a. По заданным оценкам студента (3 оценки) определить тип студента: ОТЛ, ХОР, ТРО, ДВ.
- b. Реализовать калькулятор с поддержкой операций: +, -, *, /.
- c. Задано целое положительное число k ($k \geq 1$). Напечатать все его целые делители (пример: $k=6 \rightarrow 1, 2, 3, 6$).
- d. Программа «угадай число».

ТК. Отладка программы (дать программу с несколькими типами ошибок). Программы:

- вычисление простых сомножителей числа;
- вывод таблицы умножения;
- реализовать программу для определения суммы цифр числа;

- пусть пользователь вводит число N ($0 < N < 27$), реализовать программу, которая выводит все числа из диапазона от 0 до 999, сумма цифр которых равна N .

С.Р. в ТК. Задана последовательность чисел: $1, (1+1/2), (1+1/2+1/3), \dots$. Найти первый элемент последовательности, который больше, чем заданное число a .

5. Обсуждение результатов С.Р. Массивы (синтаксис, примеры объявления, индексация).

Генерация псевдослучайных данных. Прерывание циклов (break, continue). Бинарный поиск, простейшая сортировка. Программы:

- Вывести на экран коды всех кодов символов A..Z (необходимо использовать массив для хранения кодов, в качестве индекса использовать соответствующий символ).
- Найти min и max элементы в массиве и их позицию. Найти заданный элемент в массиве и его позицию.
- Сортировка. Найти элемент в упорядоченном массиве.
- Удаление/добавление элемента в массив (упорядоченный, неупорядоченный).

ТК. Базовые задачи:

- реализовать программу, которая располагает элементы массива в обратном порядке;
- подсчитать количество разных элементов в массиве;
- определить количество разных элементов в упорядоченном массиве;
- построить таблицу значений функции (двумерные массивы).

С.Р. в ТК. Поиск подстроки в строке.

6. Обсуждение результатов С.Р. Постановка задачи по ЛР 1. Схема сдачи ЛР, обсуждение реализации (организация диалога с пользователем). Изучение следующих сортировок (оценка эффективности):

- Сортировка пузырьком.

ТК. Работа в MS Word. Примеры отчётов.

- Обсуждение разделов отчета.
- Работа со стилями в MS Word.
- Работа с рисунками в MS Word.

7. Изучение следующих сортировок (оценка эффективности):

- Сортировка выбором.
- Сортировка вставками.
- Сортировка слиянием.

ТК. Реализация рассмотренных методов сортировки:

- Сортировка пузырьком.

- b. Сортировка выбором.
- c. Сортировка вставками.
- d. Сортировка слиянием.

С.Р.в ТК. Поиск седловых точек в квадратной матрице.

8. Подпрограммы (назначение, виды). Передача параметров по значению и по ссылке.

Рекурсия. Программы:

- a. Реализовать функцию вычисления расстояния между двумя точками на плоскости.
- b. Дан массив, реализовать функцию, которая выводит индексы элементов массива, соответствующие возрастающему порядку элементов.
- c. На плоскости отмечено $N = 3K$ точек. Будем рассматривать такие варианты построения K невырожденных треугольников с вершинами в этих точках, при которых каждая из заданных точек является вершиной какого-либо треугольника. Точки расположены так, что хотя бы одно построение с указанным свойством существует. Требуется определить тот вариант, при котором суммарная площадь полученных K треугольников минимальна. Реализовать функцию, которая принимает на вход координаты точек.
- d. Задано целое число n ($n \geq 1$). Реализовать функцию для вычисления числа Фибоначчи с номером n : $f_0 = 0$, $f_1 = 1$, $f_k = f_{k-1} + f_{k-2}$, $k \geq 2$.
- e. в прямоугольной таблице $N \times M$ вначале игрок находится в левой верхней клетке. За один ход ему разрешается перемещаться в соседнюю клетку либо вправо, либо вниз (влево и вверх перемещаться запрещено). Посчитайте, сколько есть способов у игрока попасть в правую нижнюю клетку.

ТК. Реализация подпрограмм, которые разбирались на практическом занятии.

С.Р. в ТК. Реализовать рекурсивную функцию вычисления n -го числа Фибоначчи.

9. Обсуждение результатов С.Р. Изучение следующих сортировок (оценка эффективности):

- a. Сортировка Хоара.
- b. Сортировка Шелла.
- c. Сортировка подсчётом.

С.Р. Матричное умножение.

ТК. Реализация рассмотренных методов сортировки:

- a. Сортировка Хоара.
- b. Сортировка Шелла.
- c. Сортировка подсчётом.

10. Обсуждение результатов С.Р. Передача параметров в подпрограммы (параметры-константы, параметры без типа, массивы и строки открытого типа). Программы:

- a. Функция поиска элемента в массиве.
- b. Функция сортировки массива произвольного размера.
- c. С.Р. Задан массив $Y[n]$, элементами которого являются целые числа. Преобразовать массив так, чтобы все его нечетные элементы оказались в конце. Порядок элементов в четной и нечетной частях не должен измениться.

ТК. Решение задач, направленных на реализацию функций:

- a. Функция определения расстояния между парой точек плоскости.
- b. Функция поиска элемента в массиве.
- c. Функция поиска максимального элемента массива.

С.Р.в ТК. Найти в массиве слов слово по шаблону (шаблон содержит не более одного символа "?").

11. Обсуждение результатов С.Р. Постановка задачи по ЛР 2.

ТК. Решение задач, направленных на реализацию функций:

- a. Функция сортировки массива произвольного размера.
- b. Функция умножения плотной матрицы на вектор.
- c. Функция определения количества файлов в указанной директории.

С.Р. в ТК. Реализовать функцию выделения разрядов десятичного числа.

12. Типданных «указатель на функцию». Пример использования указателей на функцию на примере сортировки (функция сравнения элементов).

ТК. Решение задач, разобранных на практическом занятии.

С.Р. Задан массив целых чисел. Необходимо реализовать функцию для вычисления количества различных элементов в этом массиве.

13. Обсуждение результатов С.Р. Конструирование составных типов данных. Модули. Пример реализации модуля для работы с комплексными числами.

ТК. Решение задач. Программы:

- a. Реализация модуля работы с комплексными числами.
- b. Реализация модуля операций над отрезками (взаимное расположение, расстояние, точка пересечения и т.п.).

14. Файловый ввод-вывод (основные понятия, организация файлового ввода-вывода в С).

Решение задач с использованием файлового ввода-вывода:

- a. Файл содержит двумерные координаты точек, определяющих вершины углов многоугольника. Реализовать подпрограмму, которая определяет, является ли многоугольник выпуклым или нет.
- b. Файл содержит список фамилий студентов и оценок, полученных на экзамене (каждая строка файла <Фамилия>[\t]*<Оценка>). Предполагается, что оценки – рациональные числа (с половинками). Реализовать подпрограмму, которая выводит фамилию студента с наивысшим баллом за экзамен.

ТК. Решение задач, разобранных на практическом занятии.

15. Обработка ошибок (примеры типичных ошибок). Обсуждение проблем реализации метода Гаусса (прямой, обратный ход, проверка деления на ноль, антипереполнение). Тест по пройденному материалу. Ответы на вопросы.

Тематика практических и лабораторных занятий (Семестр 2)

- 16. *Некоторые отличия C++ от C.* Ввод/вывод, константы, работа с динамической памятью, ссылки, передача параметров по ссылке, перегрузка функций, константные параметры.
- 17. *Введение в объектно-ориентированное программирование.* Парадигмы ООП. Процедурное и объектно-ориентированное программирование. Сравнительный пример применения двух подходов.
- 18. *Классы и объекты.* Объявление класса, поля и методы, спецификаторы доступа. Описание класса. Объявление и использование объектов класса.
- 19. *Конструкторы и деструктор.* Понятие и типы конструкторов. Понятие деструктора. Когда и как вызываются конструкторы и деструктор.
- 20. *Обработка исключений.* Ошибки и способы их обработки. Обработка исключений. Правила написания обработчика. Вложенные исключения.
- 21. *Перегрузка операций.* Введение. Перегрузка операций, два способа перегрузки операций. Правила перегрузки операций различных типов. Динамические поля и перегрузка операций.
- 22. *Перегрузка операций и конструктор преобразования типа.* Пример класса CDate.
- 23. *Агрегация и наследование.* Агрегация, наследование и спецификаторы доступа. Создание и удаление объектов при наследовании. Перегрузка и перекрытие методов.
- 24. *Специальные поля и методы классов.* Константные поля и методы класса. Константные объекты. Статические поля и методы.
- 25. *Виртуальные методы.* Виртуальное перекрытие методов. Абстрактные методы и классы.
- 26. *Шаблоны.* Шаблоны функций. Введение в шаблоны классов
- 27. *Примеры применения наследования и шаблонов.* Класс Container. Класс Dictionary
- 28. *Введение в STL.*

Для обеспечения самостоятельной работы обучающихся используется электронный курс (Языки и методы программирования, <https://e-learning.unn.ru/course/view.php?id=1509>), созданный в системе электронного обучения ННГУ - <https://e-learning.unn.ru/>

Контрольные вопросы и задания для проведения текущего контроля и промежуточной аттестации по итогам освоения дисциплины приведены в п. 5.2.

5. Фонд оценочных средств для промежуточной аттестации по дисциплине (модулю), включающий:

5.1. Описание шкал оценивания результатов обучения по дисциплине

Уровень сформированности компетенций (индикатора достижения компетенций)	Шкала оценивания сформированности компетенций						
	плохо	неудовлетворительно	удовлетворительно	хорошо	очень хорошо	отлично	превосходно
	Не зачтено		Зачтено				
<u>Знания</u>	Отсутствие знаний теоретического материала. Невозможность оценить полноту знаний вследствие отказа обучающегося от ответа	Уровень знаний ниже минимальных требований. Имели место грубые ошибки.	Минимально допустимый уровень знаний. Допущено много негрубых ошибок.	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько незначительных ошибок	Уровень знаний в объеме, соответствующем программе подготовки, без ошибок.	Уровень знаний в объеме, превышающем программу подготовки.
<u>Умения</u>	Отсутствие минимальных умений. Невозможность оценить наличие умений вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы основные умения. Имели место грубые ошибки.	Продemonстрированы основные умения. Решены типовые задачи с негрубыми ошибками. Выполнены все задания, но не в полном объеме.	Продemonстрированы все основные умения. Решены все основные задачи с негрубыми ошибками. Выполнены все задания, в полном объеме, но некоторые с недочетами.	Продemonстрированы все основные умения. Решены все основные задачи. Выполнены все задания, в полном объеме, но некоторые с недочетами.	Продemonстрированы все основные умения, решены все основные задачи с отдельными незначительными недочетами, выполнены все задания в полном объеме.	Продemonстрированы все основные умения, решены все основные задачи. Выполнены все задания, в полном объеме без недочетов
<u>Навыки</u>	Отсутствие владения материалом. Невозможность оценить наличие навыков	При решении стандартных задач не продемонстрированы базовые	Имеется минимальный набор навыков для решения стандартных задач с	Продemonстрированы базовые навыки при решении стандартных задач с	Продemonстрированы базовые навыки при решении стандартных задач без	Продemonстрированы навыки при решении нестандартных задач без ошибок и	Продemonстрирован творческий подход к решению нестандартных

	вследствие отказа обучающегося от ответа	навыки. Имели место грубые ошибки.	некоторыми недочетами.	некоторыми недочетами	ошибок и недочетов.	недочетов.	ых задач.
--	--	---------------------------------------	------------------------	-----------------------	---------------------	------------	-----------

Шкала оценки при промежуточной аттестации

Оценка		Уровень подготовки
зачтено	Превосходно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «превосходно»
	Отлично	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «отлично», при этом хотя бы одна компетенция сформирована на уровне «отлично»
	Очень хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «очень хорошо», при этом хотя бы одна компетенция сформирована на уровне «очень хорошо»
	Хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «хорошо», при этом хотя бы одна компетенция сформирована на уровне «хорошо»
	Удовлетворительно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «удовлетворительно», при этом хотя бы одна компетенция сформирована на уровне «удовлетворительно»
не зачтено	Неудовлетворительно	Хотя бы одна компетенция сформирована на уровне «неудовлетворительно», ни одна из компетенций не сформирована на уровне «плохо»
	Плохо	Хотя бы одна компетенция сформирована на уровне «плохо»

5.2. Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения

5.2.1 Контрольные вопросы 1 семестр

Вопрос	Код компетенции (согласно РПД)
1. Программа на языке С. Объявления, операторы, комментарии, директивы компилятору, Структура программы на языке С. Примеры.	ОПК-2
2. Понятие типа данных. Логический тип и логические выражения в языке С. Перечислимый тип в языке С. Операции над объектами этих типов. Примеры использования.	ОПК-3, ОПК-6
3. Понятие типа данных. Данные целых типов в языке С. Представление беззнаковых и знаковых целых чисел в оперативной памяти. Операции над целочисленными данными. Примеры использования.	ОПК-3, ОПК-6
4. Понятие типа данных. Данные вещественных типов. Представление в оперативной памяти. Операции над вещественными данными. Примеры использования.	ОПК-2

5. Понятие типа данных. Данные строкового типа в языке С. Представление в оперативной памяти. Операции. Примеры использования.	ОПК-3, ОПК-6
6. Понятие типа данных. Данные символьного типа в языке С. Значение и код символа. Операции. Примеры использования.	ОПК-2
7. Понятие переменной. Объявление, определение, инициализация. Понятие константы. Объявление констант в языке С. Примеры использования.	ОПК-3, ОПК-6
8. Оператор присваивания и выражения. Преобразования встроенных типов данных. Некоторые стандартные математические функции.	ОПК-3, ОПК-6
9. Комбинированные типы данных. Массивы. Объявление массивов. Операции над массивами. Двумерные массивы. Объявления, размещение элементов массивов в памяти. Примеры использования.	ОПК-3, ОПК-6
10. Комбинированные типы данных. Объявление и использование объектов структурного типа. Объявление и использование объектов типа «объединение». Примеры.	ОПК-3, ОПК-6
11. Этапы решения задач с использованием вычислительной техники.	ОПК-2, ПК-4
12. Технология структурного программирования. Теорема о базисных алгоритмических конструкциях. Составной оператор. Примеры использования.	ОПК-3, ОПК-6, ПК-4
13. Технология структурного программирования. Теорема о базисных алгоритмических конструкциях. Программирование выбора. Условный оператор. Полный и неполный варианты. Примеры использования.	ОПК-3, ОПК-6, ПК-4
14. Технология структурного программирования. Теорема о базисных алгоритмических конструкциях. Программирование выбора. Оператор множественного выбора. Примеры использования.	ОПК-3, ОПК-6, ПК-4
15. Технология структурного программирования. Теорема о базисных алгоритмических конструкциях. Программирование повторяющихся действий. Операторы цикла. Цикл с предусловием while. Операторы break и continue. Примеры использования.	ОПК-3, ОПК-6, ПК-4
16. Технология структурного программирования. Теорема о базисных алгоритмических конструкциях. Программирование повторяющихся действий. Операторы цикла. Цикл с постусловием dowhile. Операторы break и continue. Примеры использования.	ОПК-3, ОПК-6, ПК-4
17. Технология структурного программирования. Теорема о базисных алгоритмических конструкциях. Программирование повторяющихся действий. Операторы цикла. Цикл с известным числом повторений for. Операторы break и continue. Примеры использования.	ОПК-3, ОПК-6, ПК-4
18. Концепция модульного программирования. Необходимость, средства поддержки в языках. Функции в языке С. Объявление, реализация и использование функций.	ОПК-3, ОПК-6
19. Концепция модульного программирования. Функции. Локальные и глобальные переменные. Способы передачи параметров в языке С. Рекурсивные функции.	ОПК-3, ОПК-6
20. Концепция модульного программирования. Особенности использования массивов в качестве параметров процедур и функций.	ОПК-3, ОПК-6
21. Тип указатель на функцию. Примеры использования.	ОПК-2
22. Методы работы с внешней памятью. Файлы. Основные понятия, записи, физический и логический файл, связывание. Текстовые файлы и работа с ними.	ОПК-2
23. Методы работы с внешней памятью. Файлы. Основные понятия, записи, физический и логический файл, связывание. Типизированные файлы и работа с ними.	ОПК-2
24. Концепция модульного программирования. Модули. Назначение, построение, использование. Многомодульные программы – общие принципы организации и сборки.	ОПК-2

25. Виртуальное адресное пространство прикладной программы. Сегменты данных, кода, стек, куча. Особенности и техника передачи параметров. Побочный эффект.	ОПК-2
26. Ввод/вывод. Функция вывода printf. Управление форматом вывода данных. Функция ввода scanf.	ОПК-2
27. Динамическое управление памятью в языке C. Типизированные и бестиповые указатели. Статическое и динамическое распределение памяти. Функции работы с динамической памятью.	ОПК-2
28. Динамическое управление памятью в языке C. Динамические массивы.	ОПК-2
29. Алгоритмы сортировки. Сортировки, использующие сравнения. Устойчивые и неустойчивые. Классы сортировок. Сортировки пузырьком, выбором, вставками.	ОПК-2
30. Алгоритмы сортировки. Сортировки, использующие сравнения. Устойчивые и неустойчивые. Классы сортировок. Сортировка слиянием. Быстрая сортировка (Хоара).	ОПК-2

2 семестр

Вопрос	Код компетенции (согласно РПД)
1. Обзор операций языка C++. Приоритеты операций. Операции низкого и высокого уровней. Примеры операций.	ОПК-2
2. Выражения в языке C++. Правила вычисления выражений. Приоритеты операций и правило ассоциативности. Приведение типов в C++.	ОПК-2
3. Ссылки. Константные ссылки. Примеры использования ссылок.	ОПК-2
4. Парадигмы ООП. Примеры применения парадигм ООП.	ОПК-3, ОПК-6, ПК-4
5. Абстрактные типы данных. Понятие класса. Инкапсуляция. Объявление класса. Поля и методы. Скрытие членов класса: секции public и private. Описание класса. Классы и структуры. Пример разработки класса «Комплексное число».	ОПК-3, ОПК-6
6. Объявление объектов класса. Указатели на объекты. Взаимодействие объектов и методов класса. Пример разработки класса «Строка».	ОПК-3, ОПК-6
7. Конструкторы. Типы конструкторов. Конструкторов классов «Комплексное число» и «Строка». Когда и как вызываются конструкторы.	ОПК-3, ОПК-6
8. Конструкторы копирования и преобразования типов. Примеры разработки конструкторов копирования и преобразования типов.	ОПК-3, ОПК-6
9. Деструкторы. Когда и как вызываются деструкторы.	ОПК-3, ОПК-6
10. Константные поля. Инициализация константных полей. Константные методы. Примеры константных полей и методов. Константные объекты.	ОПК-2
11. Статические поля и методы. Примеры статических полей и методов.	ОПК-2
12. Перегрузка операций. Общие правила и ограничения. Два способа перегрузки операций. Дружественные функции класса.	ОПК-2

13. Правила перегрузки операции присваивания.	ОПК-2
14. Правила перегрузки операций +, -, *, /.	ОПК-2
15. Правила перегрузки операций типа +=.	ОПК-2
16. Перегрузка операции индексации и унарных операций.	ОПК-2
17. Перегрузка операции ввода/вывода в поток.	ОПК-2
18. Перегрузка операций и конструкторы преобразования типов. Класс CData	ОПК-2
19. Наследование. Перекрытие методов. Спецификаторы доступа при наследовании.	ОПК-3, ОПК-6
20. Конструирование и деструктурирование объектов при наследовании.	ОПК-3, ОПК-6
21. Множественное наследование и иерархия «родитель-ребенок»	ОПК-3, ОПК-6
22. Наследование и перекрытие методов. Классы Container и Dictionary.	ОПК-3, ОПК-6
23. Обработка ошибок. Возбуждение и обработка исключений.	ОПК-3, ОПК-6
24. Шаблоны функций. Механизм конкретизации (инстанцирования) шаблона функций. Вызов шаблонов функций. Перегрузка шаблонов функций. Специализация шаблонов функций. Модели компиляции шаблонов. Примеры шаблонов функций.	ОПК-3, ОПК-6
25. Шаблоны классов. Объявление и описание классов с шаблонами. Специализация класса с шаблоном. Пример класса с шаблоном.	ОПК-3, ОПК-6
26. Виртуальные методы. Правила виртуального перекрытия методов. Приведение типов при виртуальном перекрытии методов.	ОПК-3, ОПК-6
27. Механизм работы виртуальных методов. Ранее и позднее связывание. Примеры применения виртуальных методов.	ОПК-3, ОПК-6
28. Абстрактные методы и классы. Понятия, правила и ограничения. Примеры абстрактных классов.	ОПК-3, ОПК-6

5.2.2. Типовые задания/задачи для оценки сформированности компетенции ОПК-6 и ОПК-2

Написать функцию (и тестирующую функцию main), которая:

1. по номеру года выводит его название в восточном календаре. В восточном календаре принят 60-летний цикл, состоящий из 12-летних подциклов, обозначаемых названиями цвета: зеленый, красный, желтый, белый и черный. В каждом подцикле год носит название животного: крысы, коровы, тигра, зайца, дракона, змеи, лошади, овцы, обезьяны, курицы, собаки и свиньи. Известно, что 1984 год был началом цикла — годом зеленой крысы.
2. определяет, может ли шар радиуса r пройти через ромбообразное отверстие с диагоналями p и q .

3. определяет, пройдет ли шкаф с заданными размерами через заданное прямоугольное отверстие.

5.2.3. Типовые задания/задачи для оценки сформированности компетенции ПК-4

Создать класс Liquid (жидкость), имеющий поля названия и плотности. Определить методы переназначения и изменения плотности. Определить необходимые конструкторы, вывод параметров объекта на консоль.

5.2.4. Типовые тестовые задания для оценки сформированности компетенции ОПК-6 и ОПК-2

1. Модульное программирование – парадигма программирования в соответствии с которой компьютерная программа
 - a. Представляется в виде совокупности взаимодействующих процедур (модулей) +
 - b. Проектируется в соответствии со структурной (нисходящей) технологией проектирования без использования оператора goto.
 - c. Формируется как совокупность взаимодействующих объектов, обменивающихся сообщениями
2. Структурное программирование – парадигма программирования в соответствии с которой компьютерная программа
 - a. Представляется в виде совокупности взаимодействующих процедур (модулей)
 - b. Проектируется в соответствии со структурной (нисходящей) технологией проектирования без использования оператора goto +
 - c. Формируется как совокупность взаимодействующих объектов, обменивающихся сообщениями
3. Объектно-ориентированное программирование – парадигма программирования в соответствии с которой компьютерная программа
 - a. Представляется в виде совокупности взаимодействующих процедур (модулей)
 - b. Проектируется в соответствии со структурной (нисходящей) технологией проектирования без использования оператора goto.

6. Учебно-методическое и информационное обеспечение дисциплины

а) основная литература:

1. Грудзинский А.О., Мееров И.Б., Сысоев А.В. Методы программирования. Курс на основе языка ObjectPascal. – Н.Новгород, изд. ННГУ, 2006. – 392 с. 50 экз.
2. Карпенко С.Н. и др. Методы объектно-ориентированного программирования. URL: <http://e-learning.unn.ru/course/view.php?id=251>.
3. Страуструп Б. Курс «Язык программирования C++ для профессионалов». – <http://www.intuit.ru/studies/courses/98/98/info>

б) дополнительная литература:

1. Кетков Ю.Л. Введение в языки программирования С и С++. Интернет-университет информационных технологий - ИНТУИТ.ру, БИНОМ. Лаборатория знаний, 2006 г. – 344 с. <http://www.intuit.ru/studies/courses/1039/231/info>.
2. Огнева, М. В. Программирование на языке с++: практический курс : учебное пособие для бакалавриата и специалитета / М. В. Огнева, Е. В. Кудрина. — М. : Издательство Юрайт,

2017. — 335 с. — (Серия : Бакалавр и специалист). — ISBN 978-5-534-05123-0. — Режим доступа: www.biblio-online.ru/book/7670D7EC-AC37-4675-8EAE-DD671BC6D0E4.

3. Подбельский В.В. Язык С++. М.: Финансы и статистика, 5-е изд. – 560 с. 12 экз. (+ 2004 -75 экз., 2006 - 7 экз., +2007 – 6 экз., 2008 – 14 экз.)
4. Калинина Н., Костюкова Н. Курс «Основы программирования на языке С». – <http://www.intuit.ru/studies/courses/43/43/info>
5. Фридман А. Курс «Язык программирования С++». – <http://www.intuit.ru/studies/courses/17/17/info>
6. Баженова И., Сухомлин В. Курс «Введение в программирование». – <http://www.intuit.ru/studies/courses/27/27/info>

в) Интернет-ресурсы

1. Единое окно доступа к образовательным ресурсам (Программирование) <http://window.edu.ru>
2. Электронно-библиотечная система <https://e.lanbook.com/>

7. Материально-техническое обеспечение дисциплины

Помещения представляют собой учебные аудитории для проведения учебных занятий, предусмотренных программой (лекционного и семинарского типа), оснащенные оборудованием и техническими средствами обучения.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечены доступом в электронную информационно-образовательную среду.

Программа составлена в соответствии с требованиями ОС ННГУ **09.03.04 Программная инженерия**.

Авторы А.В. Сысоев, С.Н. Карпенко

Рецензент (ы) _____

Заведующий кафедрой _____ Р.Г. Стронгин

Программа одобрена на заседании методической комиссии института информационных технологий, математики и механики
от 2 июня 2021 года, протокол № 8