

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

УТВЕРЖДЕНО
решением Ученого совета ННГУ
протокол № 10 от 27.08.2025

Рабочая программа дисциплины

Основы программирования

Уровень высшего образования
Бакалавриат

Направление подготовки / специальность
02.03.02 - Фундаментальная информатика и информационные технологии

Направленность образовательной программы
Системное программирование

Форма обучения
очная

г. Нижний Новгород

2025 год начала подготовки

1. Место дисциплины в структуре ОПОП

Дисциплина Б1.О.13 Основы программирования относится к обязательной части образовательной программы.

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 | Для текущего контроля успеваемости | Для промежуточной аттестации |
| ОПК-3: Способен к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям | ОПК-3.1: Пишет программный код с использованием языков определения и манипулирования данными в базах данных ОПК-3.2: Проектирует базы данных для программных модулей и компонентов ОПК-3.3: Оптимизирует производительность работы с базами данных | ОПК-3.1: Знать основные идеи, лежащие в основе современных языков программирования и основные составляющие современного языка программирования на примере C и C++. принципы реализации технологий разработки программного обеспечения: структурное, модульное и объектно-ориентированное программирование – в языках программирования C и C++; методы конструирования новых типов данных(C, C++). методы и приемы работы с динамической памятью(C, C++). методы работы с внешней памятью (C, C++). технологии разработки программного обеспечения (структурное и модульное программирование – подробно, объектно-ориентированное программирование – на уровне общих идей). ОПК-3.2: Уметь разрабатывать и отлаживать программы средней сложности с использованием языков | Практическое задание Проект Собеседование Тест | Экзамен: Контрольные вопросы |

| | | | | |
|--|--|---|--|--|
| | | <p>программирования C, C++. использовать современные инструментальные и вычислительные средства.</p> <p>ОПК-3.3: Владеть современными интегрированными средами разработки программ навыками реализации, тестирования и отладки программных систем средней сложности</p> | | |
|--|--|---|--|--|

3. Структура и содержание дисциплины

3.1 Трудоемкость дисциплины

| | |
|--|-----------------------------|
| | очная |
| Общая трудоемкость, з.е. | 10 |
| Часов по учебному плану | 360 |
| в том числе | |
| аудиторные занятия (контактная работа): | |
| - занятия лекционного типа | 96 |
| - занятия семинарского типа (практические занятия / лабораторные работы) | 128 |
| - КСР | 4 |
| самостоятельная работа | 60 |
| Промежуточная аттестация | 72 Экзамен |

3.2. Содержание дисциплины

(структурированное по темам (разделам) с указанием отведенного на них количества академических часов и виды учебных занятий)

| Наименование разделов и тем дисциплины | Всего (часы) | в том числе | | | |
|---|--------------|--|--|-------------|---|
| | | Контактная работа (работа во взаимодействии с преподавателем), часы из них | | | Самостоятельная работа обучающегося, часы |
| | | Занятия лекционного типа | Занятия семинарского типа (практические занятия/лабораторные работы), часы | Всего | |
| 0 Ф 0 | 0 Ф 0 | 0 Ф 0 | 0 Ф 0 | 0 Ф 0 | |
| 1.1 Введение в предмет. Структура и содержание курса. Литература. | 2 | 2 | 0 | 2 | 0 |
| 1.2 Решение задач с использованием вычислительной техники | 4 | 2 | 2 | 4 | 0 |

| | | | | | |
|--|------|------|-----|------|----|
| 1.3 Программа на языке С – введение. Представление информации в вычислительной технике. Понятие типа данных. Целые типы в языке С. | 8 | 2 | 4 | 6 | 2 |
| 1.4 Типы данных. Представление чисел с плавающей запятой. Программа на языке С – продолжение. | 8 | 2 | 4 | 6 | 2 |
| 1.5 Структурное программирование. Теория, идеи и принципы, последовательные действия, ветвления. | 8 | 2 | 4 | 6 | 2 |
| 1.6 Структурное программирование. Циклы while, do while. | 8 | 2 | 4 | 6 | 2 |
| 1.7 Контрольная работа по теории №1. | 0.66 | 0.66 | 0 | 0.66 | 0 |
| 1.8 Конструирование новых типов данных. Одномерные массивы. | 8 | 2 | 4 | 6 | 2 |
| 1.9 Структурное программирование. Цикл for. | 8 | 2 | 4 | 6 | 2 |
| 1.10 Типы данных. Представление логической, символьной информации. Ввод/вывод в языке С. | 6 | 2 | 2 | 4 | 2 |
| 1.11 Модульное программирование – часть 1. Концепции, функции. | 8 | 2 | 4 | 6 | 2 |
| 1.12 Модульное программирование – часть 2. Передача параметров. | 8 | 2 | 4 | 6 | 2 |
| 1.13 Модульное программирование – часть 3. Указатели. | 8 | 2 | 4 | 6 | 2 |
| 1.14 Контрольная работа по теории №2. | 0.67 | 0.67 | 0 | 0.67 | 0 |
| 1.15 Сортировки – часть 1. Сортировки за время $O(N^2)$. | 8 | 2 | 4 | 6 | 2 |
| 1.16 Сортировки – часть 2. Рекурсия. Сортировки за время $O(N \log N)$. | 8 | 2 | 4 | 6 | 2 |
| 1.17 Сортировки – часть 3. Сортировки за линейное время. | 8 | 2 | 4 | 6 | 2 |
| 1.18 Строки в языке С. | 8 | 2 | 4 | 6 | 2 |
| 1.19 Конструирование новых типов данных. Многомерные массивы. Структуры. Объединения. Перечисления. | 6 | 2 | 2 | 4 | 2 |
| 1.20 Контрольная работа по теории №3. | 0.67 | 0.67 | 0 | 0.67 | 0 |
| 1.21 Среда исполнения программ. Программа в среде Microsoft Windows. | 4 | 2 | 0 | 2 | 2 |
| 1.22 Современная система разработки программного обеспечения. | 4 | 2 | 0 | 2 | 2 |
| 1.23 Модульное программирование – часть 4. Модули. | 8 | 2 | 4 | 6 | 2 |
| 1.24 Методы работы с внешней памятью. Файлы. | 6 | 2 | 2 | 4 | 2 |
| 1.25 Переписывание пропущенных КР по теории. | 2 | 2 | 0 | 2 | 0 |
| 1.26 Контрольная работа по практике. | 2 | 2 | 0 | 2 | 0 |
| 2.1 Отличия языка С++ от языка С | 8 | 3 | 4 | 7 | 1 |
| 2.2 Введение в объектно-ориентированное программирование | 5 | 4 | 0 | 4 | 1 |
| 2.3 Классы и объекты | 12 | 4 | 6 | 10 | 2 |
| 2.4 Конструкторы и деструктор | 12 | 4 | 6 | 10 | 2 |
| 2.5 Обработка исключений | 12 | 4 | 6 | 10 | 2 |
| 2.6 Перегрузка операций | 12 | 4 | 6 | 10 | 2 |
| 2.7 Пример. Класс Cdate | 8 | 3 | 4 | 7 | 1 |
| 2.8 Наследование и иерархия классов | 12 | 4 | 6 | 10 | 2 |
| 2.9 Специальные поля и методы классов | 12 | 4 | 6 | 10 | 2 |
| 2.10 Виртуальные методы. Абстрактные виртуальные методы и классы | 12 | 4 | 6 | 10 | 2 |
| 2.11 Шаблоны. Шаблоны функций и шаблоны классов | 11 | 4 | 6 | 10 | 1 |
| 2.12 Пример. Класс Container | 9 | 3 | 4 | 7 | 2 |
| 2.13 Пример. Класс Dictionary | 9 | 3 | 4 | 7 | 2 |
| Аттестация | 72 | | | | |
| КСР | 4 | | | 4 | |
| Итого | 360 | 96 | 128 | 228 | 60 |

Содержание разделов и тем дисциплины

Цели и задачи изучения дисциплины

Целью освоения дисциплины «Основы программирования» является изучение следующих аспектов:

- общие вопросы создания программ, включая основные этапы процесса разработки и используемые средства;
- краткие сведения о среде исполнения программ;
- краткие сведения об инструментах программирования;
- основные элементы и принципы построения языков программирования высокого уровня на примере языков С и С++;
- базовые алгоритмы и основы алгоритмизации с примерами на языке С;
- различные способы описания моделей объектов предметной области с помощью конструирования типов данных с использованием средств С и С++;
- вопросы динамического управления памятью и работы с файлами с использованием средств С и С++;
- основы технологии объектно-ориентированного программирования на примере С++;
- углубленные элементы технологии объектно-ориентированного программирования: наследование, виртуальные методы, шаблоны функций и классов.

Задачи

- 1) Изучить теоретические основы дисциплины «Основы программирования».
- 2) Обеспечить формирование компетенции ОПК-3 в соответствии с требованиями образовательной программы.
- 3) Сформировать умения использовать современные интегрированные среды разработки программ, разрабатывать, тестировать и отлаживать программные системы средней сложности с применением языков программирования С и С++, а также использовать современные инструментальные и вычислительные средства.

Содержание:

Семестр 1

- 1.1. Введение в предмет. Структура и содержание курса. Литература по курсу.
- 1.2. Решение задач с использованием вычислительной техники.
- 1.3. Программа на языке С – введение. Представление информации в вычислительной технике. Понятие типа данных. Целые типы в языке С.
- 1.4. Типы данных. Представление чисел с плавающей запятой. Программа на языке С – продолжение.
- 1.5. Структурное программирование. Теория, идеи и принципы, последовательные действия, ветвления.
- 1.6. Структурное программирование. Циклы while, do while.
- 1.7. Контрольная работа по теории №1.
- 1.8. Конструирование новых типов данных. Одномерные массивы.
- 1.9. Структурное программирование. Цикл for.
- 1.10. Типы данных. Представление логической, символьной информации. Ввод/вывод в языке С.
- 1.11. Модульное программирование – часть 1. Концепции, функции.
- 1.12. Модульное программирование – часть 2. Передача параметров.
- 1.13. Модульное программирование – часть 3. Указатели.
- 1.14. Контрольная работа по теории №2.
- 1.15. Сортировки – часть 1. Сортировки за время $O(N^2)$.
- 1.16. Сортировки – часть 2. Рекурсия. Сортировки за время $O(N \log N)$.
- 1.17. Сортировки – часть 3. Сортировки за линейное время.
- 1.18. Строки в языке С.
- 1.19. Конструирование новых типов данных. Многомерные массивы. Структуры. Объединения.

Перечисления.

- 1.20.Контрольная работа по теории №3.
- 1.21.Среда исполнения программ. Программа в среде Microsoft Windows.
- 1.22.Современная система разработки программного обеспечения.
- 1.23.Модульное программирование – часть 4. Модули.
- 1.24.Методы работы с внешней памятью. Файлы.

Семестр 2

- 2.1. Отличия языка C++ от языка C
- 2.2. Введение в объектно-ориентированное программирование
- 2.3. Классы и объекты
- 2.4. Конструкторы и деструктор
- 2.5. Обработка исключений
- 2.6. Перегрузка операций
- 2.7. Пример. Класс Cdate
- 2.8. Наследование и иерархия классов
- 2.9. Специальные поля и методы классов
- 2.10. Виртуальные методы. Абстрактные виртуальные методы и классы
- 2.11. Шаблоны. Шаблоны функций и шаблоны классов
- 2.12. Пример. Класс Container
- 2.13. Пример. Класс Dictionary

4. Учебно-методическое обеспечение самостоятельной работы обучающихся

Самостоятельная работа обучающихся включает в себя подготовку к контрольным вопросам и заданиям для текущего контроля и промежуточной аттестации по итогам освоения дисциплины приведенным в п. 5.

Для обеспечения самостоятельной работы обучающихся используются:
Электронные курсы, созданные в системе электронного обучения ННГУ:

Языки и методы программирования, <http://e-learning.unn.ru/course/view.php?id=1509>.

Иные учебно-методические материалы:

1. Электронный курс "Языки и методы программирования," (<https://e-learning.unn.ru/course/view.php?id=1509>).
2. Сысоев А.В. Основы программирования: Учебно-методическое пособие. – Нижний Новгород: Нижегородский госуниверситет, 2024. – 32 с.

5. Фонд оценочных средств для текущего контроля успеваемости и промежуточной аттестации по дисциплине (модулю)

5.1 Типовые задания, необходимые для оценки результатов обучения при проведении текущего контроля успеваемости с указанием критериев их оценивания:

5.1.1 Типовые задания (оценочное средство - Практическое задание) для оценки сформированности компетенции ОПК-3:

По темам 1.1 - 1.26:

1. Дан массив A , содержащий n вещественных чисел. Проверить, является ли массив упорядоченным по возрастанию.
2. Дан массив A , содержащий n целых чисел и вещественное число R . Найти индексы двух элементов массива, сумма которых наиболее близка к числу R .
3. Дана квадратная матрица размера $N \times N$. Найти суммы элементов всех ее четных строк.
4. Дан массив A , содержащий n натуральных чисел. Проверить, образуют ли его элементы арифметическую прогрессию.
5. Дан массив A , содержащий n вещественных чисел. Найти индексы k_1 и k_2 двух элементов массива, расстояние между которыми $(|A[k_1] - A[k_2]|)$ минимально.
6. Дана квадратная матрица размера $N \times N$. Найти суммы элементов всех ее нечетных строк.
7. Дан массив A , содержащий n натуральных чисел. Вычислить сумму S тех элементов массива, значения которых совпадают с их индексами.
8. Задан массив A , содержащий n вещественных чисел. Определить количество разных элементов массива.
9. Дана квадратная матрица размера $N \times N$. Найти суммы элементов всех ее четных столбцов.

По темам 2.1 - 2.13:

1. Определить структуру для представления трехмерного вектора. Обеспечить выполнение операций сложения, вычитания, скалярного произведения векторов, умножения вектора на скаляр, вычисления длины вектора и сравнения длин векторов.
2. Определить структуру для представления треугольника координатами вершин на плоскости. Обеспечить выполнение операций вычисления площади, периметра треугольника, вычисления высот, а также определения вида треугольника.
3. Биржа труда. База вакансий содержит следующую информацию: фирма, должность, условия труда и оплаты, требования к специалисту. Обеспечить поиск фирм, которым требуются люди на должность «менеджера».
4. Разработать класс Рациональное число. Класс должен хранить корректные дроби (знаменатель не равен 0) и выполнять с ними 4 стандартные арифметические операции. Дробь должна храниться в несократимом виде.
- 5.1. 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. Создать производный класс Alcohol (спирт), имеющий крепость. Определить методы переназначения и изменения крепости. Определить необходимые конструкторы, вывод параметров объекта на консоль.

5.3. Из класса Alcohol создать производный класс Beer (пиво), имеющий процент содержания хмеля. Определить методы переназначения и изменения крепости и процента хмеля.

Определить необходимые конструкторы, вывод параметров объекта на консоль

6.1. Создать класс Man (человек), с полями: имя, возраст, пол и вес. Определить методы переназначения имени, изменения возраста и изменения веса. Определить необходимые конструкторы, вывод параметров объекта на консоль.

6.2. Создать производный класс Student, имеющий поля факультет, курс, группа. Определить методы изменения перехода на следующий курс, перевода в другую группу. Определить необходимые конструкторы, вывод параметров объекта на консоль.

Критерии оценивания (оценочное средство - Практическое задание)

| Оценка | Критерии оценивания |
|------------|---|
| зачтено | Выполнены не все лабораторные работы или выполнены не в полном объеме (представлено не полное описание этапов выполнения заданий, программа работает некорректно, результаты работы не представлены преподавателю). |
| не зачтено | Выполнены не все лабораторные работы или выполнены не в полном объеме (представлено не полное описание этапов выполнения заданий, программа работает некорректно, результаты работы не представлены преподавателю). |

5.1.2 Типовые задания (оценочное средство - Проект) для оценки сформированности компетенции ОПК-3:

Проект 1.

Фирма «Умелые ручки» выпускает двухдверные одежные шкафы.

Состав шкафа:

- накладная задняя стенка из ДВП высотой h от 180 до 220 см, шириной w от 80 до 120 см и толщиной 5 мм,
- две боковины из ДСП высотой h , глубиной d от 50 до 90 см и толщиной 15 мм,
- накладные верхняя и нижняя крышки из ДСП шириной w , глубиной d и толщиной 15 мм,
- две накладные двери из дерева высотой h , общей шириной w и толщиной 1 см,
- внутренние полки в шкафу через каждый 40 см из ДСП.

Считая, что известны плотности ДСП, ДВП и дерева, найти массу шкафа.

Проект 2.

Дана шахматная доска. Для двух различных клеток K_1 , K_2 доски и выбранной пользователем фигуры (король, ферзь, ладья, слон, конь) определить:

- Может ли фигура за один ход перейти из K_1 в K_2 ?
- Если нет, могут ли это сделать какие-то из остальных фигур? Если могут, вывести их список.

Проект 3.

Разработать программу «Угадай число».

В программе должно быть предусмотрено два режима.

Режим 1. Программа «загадывает» случайное число из диапазона от 1 до 1000. Пользователь должен вводить отгадки, на которые программа сообщает: «загаданное число больше», «загаданное число меньше», «угадали». Работа завершается, когда пользователь угадает число. Также программа должна подсчитывать число попыток и выводить его в конце работы.

Режим 2. Пользователь загадывает число из диапазона от 1 до 1000 и вводит его. Программа пытается «угадать» число, выводя на экран отгадки, на которые пользователь вводит >, < или =. Работа завершается, когда программа угадает число. Также программа должна подсчитывать число попыток и выводить его в конце работы.

Критерии оценивания (оценочное средство - Проект)

| Оценка | Критерии оценивания |
|------------|---|
| зачтено | Выполнены все или большая часть этапов решения задачи или задача решена с незначительными недочетами. Код и результаты работы представлены преподавателю в срок. |
| не зачтено | Выполнены не все лабораторные работы или выполнены не в полном объеме (представлено не полное описание этапов выполнения заданий, код работает некорректно, результаты работы не представлены преподавателю). |

5.1.3 Типовые задания (оценочное средство - Собеседование) для оценки сформированности компетенции ОПК-3:

По темам 1.1 - 1.26:

- Беззнаковые целые типы данных в C. Представление в памяти. Операции.
- Теорема о структурном программировании. Условный оператор в C, блок-схема. Условная операция. Примеры.
- Понятие типа данных. Примеры.
- Идентификаторы в языке C, правила формирования. Переменные. Константы: явные, именованные.
- Функция в языке C. Объявление (прототип), реализация, использование. Примеры.
- Алгоритм поиска минимального элемента в массиве. Реализация в виде функции.
- Знаковые целые типы данных в C. Представление в памяти в дополнительном коде. Операции.
- Теорема о структурном программировании. Оператор множественного выбора в C, блок-схема. Примеры.
- Одномерные массивы в C. Объявление, инициализация, индексация. Массивы без указания размера. Операция sizeof.
- ВАП процесса. Плоская схема адресации памяти. Сегменты.
- Заголовочный файл в языке C. Назначение, содержимое, использование. Исключение дублирования при компиляции.

- Алгоритм поиска с барьером. Реализация в виде функции.
- Представление вещественных чисел по стандарту IEEE 754.
- Теорема о структурном программировании. Оператор цикла с предусловием, блок-схема. Примеры.
- Многомерные массивы в С. Объявление, инициализация, индексация. Хранение в памяти.
- Указатели в С. Объявление, инициализация, арифметика указателей.
- Многофайловая программа на С. Схема организации. Схема сборки.
- Алгоритм бинарного поиска. Реализация в виде функции.

Критерии оценивания (оценочное средство - Собеседование)

| Оценка | Критерии оценивания |
|------------|---|
| зачтено | Студент дал развернутый ответ на все вопросы без существенных ошибок. |
| не зачтено | При ответе студент допускает грубые ошибки в основном материале. |

5.1.4 Типовые задания (оценочное средство - Тест) для оценки сформированности компетенции ОПК-3:

По темам 2.1 - 2.13:

- Что может быть полем (членом) класса в языке C++:
 1. a. массив +
 2. b. другой класс +
 3. c. структура +
 4. d. указатель на сам этот класс +
 5. e. константа
 6. f. выражение
 7. g. сам этот класс
- Конструктор в языке C++ - специальный метод класса, имеющий следующие особенности (отметьте правильные варианты):
 1. a. Имя метода-конструктора совпадает с именем класса +
 2. b. Конструктор имеет имя Constructor.
 3. c. Тип возвращаемого конструктором значения определяется типом объекта
 4. d. Конструктор ничего не возвращает, даже типа void. +
 5. e. Можно задавать значения параметров по умолчанию +
 6. f. Параметры конструктора не могут задаваться по умолчанию.
- Деструктор в языке C++ - специальный метод класса, имеющий следующие особенности (отметьте правильные варианты):
 1. a. Имя метода-деструктора имеет вид: ~<имя_класса> +
 2. b. Конструктор имеет имя Destructor.
 3. c. Деструктор не имеет параметров +
 4. d. Параметры деструктора могут иметь любой тип. Можно задавать значения параметров по умолчанию.
 5. e. Деструктор ничего не возвращает +
 6. f. Тип возвращаемого деструктором значения определяется типом объекта

Критерии оценивания (оценочное средство - Тест)

| Оценка | Критерии оценивания |
|------------|--|
| зачтено | как минимум 80% правильных ответов в тесте |
| не зачтено | менее 80% правильных ответов в тесте |

5.2. Описание шкал оценивания результатов обучения по дисциплине при промежуточной аттестации

Шкала оценивания сформированности компетенций

| Уровень сформированности компетенций (индикатора достижения 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 несущественных ошибок | Уровень знаний в объеме, соответствующем программе подготовки. Ошибок нет. | 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 | Продемонстрированы основные умения. Решены типовые задачи с негрубыми ошибками. Выполнены все задания, но не в полном объеме | Продемонстрированы все основные умения. Решены все основные задачи с негрубыми ошибками. Выполнены все задания в полном объеме, но некоторые с недочетами | Продемонстрированы все основные умения. Решены все основные задачи. Выполнены все задания в полном объеме, но некоторые с недочетами | Продемонстрированы все основные умения. Решены все основные задачи с отдельными и несущественными недочетами, выполнены все задания в полном объеме | Продемонстрированы все основные умения. Решены все основные задачи. Выполнены все задания, в полном объеме без недочетов |
| <u>Навыки</u> | Отсутствие базовых навыков. Невозможность оценить наличие навыков вследствие отказа обучающегося от | При решении стандартных задач не продемонстрированы базовые навыки. Имели место грубые ошибки | Имеется минимальный набор навыков для решения стандартных задач с некоторым | Продемонстрированы базовые навыки при решении стандартных задач с некоторым и | Продемонстрированы базовые навыки при решении стандартных задач без ошибок и недочетов | Продемонстрированы навыки при решении нестандартных задач без ошибок и | Продемонстрирован творческий подход к решению нестандартных задач |

| | | | | | | | |
|--|--------|--|-----------------|------------|--|-----------|--|
| | ответа | | и недочетами | недочетами | | недочетов | |
|--|--------|--|-----------------|------------|--|-----------|--|

Шкала оценивания при промежуточной аттестации

| Оценка | | Уровень подготовки |
|-------------------|----------------------------|--|
| зачтено | превосходно | Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «превосходно», продемонстрированы знания, умения, владения по соответствующим компетенциям на уровне выше предусмотренного программой |
| | отлично | Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «отлично». |
| | очень хорошо | Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «очень хорошо» |
| | хорошо | Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «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». |
| | плохо | Хотя бы одна компетенция сформирована на уровне «плохо» |

5.3 Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения на промежуточной аттестации с указанием критериев их оценивания:

5.3.1 Типовые задания (оценочное средство - Контрольные вопросы) для оценки сформированности компетенции ОПК-3

По темам 1.1 - 1.26:

1. Беззнаковые целые типы данных в С. Представление в памяти. Операции.
2. Теорема о структурном программировании. Условный оператор в С, блок-схема. Условная операция. Примеры.
3. Понятие типа данных. Примеры.
4. Идентификаторы в языке С, правила формирования. Переменные. Константы: явные, именованные.
5. Функция в языке С. Объявление (прототип), реализация, использование. Примеры.
6. Алгоритм поиска минимального элемента в массиве. Реализация в виде функции.
7. Знаковые целые типы данных в С. Представление в памяти в дополнительном коде. Операции.

8. Теорема о структурном программировании. Оператор множественного выбора в C, блок-схема. Примеры.
9. Одномерные массивы в C. Объявление, инициализация, индексация. Массивы без указания размера. Операция sizeof.
10. ВАП процесса. Плоская схема адресации памяти. Сегменты.
11. Заголовочный файл в языке C. Назначение, содержимое, использование. Исключение дублирования при компиляции.
12. Алгоритм поиска с барьером. Реализация в виде функции.
13. Представление вещественных чисел по стандарту IEEE 754.
14. Теорема о структурном программировании. Оператор цикла с предусловием, блок-схема. Примеры.
15. Многомерные массивы в C. Объявление, инициализация, индексация. Хранение в памяти.
16. Указатели в C. Объявление, инициализация, арифметика указателей.
17. Многофайловая программа на C. Схема организации. Схема сборки.
18. Алгоритм бинарного поиска. Реализация в виде функции.

По темам 2.1 - 2.13:

1. Парадигмы ООП. Примеры применения парадигм ООП.
2. 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”.
3. Объявление объектов класса. Указатели на объекты. Взаимодействие объектов и методов класса. Пример разработки класса “Строка”.
4. Конструкторы. Типы конструкторов. Конструкторов классов «Комплексное число» и «Строка». Когда и как вызываются конструкторы.
5. Конструкторы копирования и преобразования типов. Примеры разработки конструкторов копирования и преобразования типов.
6. Деструкторы. Когда и как вызываются деструкторы.
7. Наследование. Перекрытие методов. Спецификаторы доступа при наследовании.
8. Конструирование и деструктурирование объектов при наследовании.
9. Обзор операций языка C++. Приоритеты операций. Операции низкого и высокого уровней. Примеры операций.

10. Выражения в языке C++. Правила вычисления выражений. Приоритеты операций и правило ассоциативности. Приведение типов в C++.
11. Ссылки. Константные ссылки. Примеры использования ссылок.
12. Константные поля. Инициализация константных полей. Константные методы. Примеры константных полей и методов. Константные объекты.
13. Статические поля и методы. Примеры статических полей и методов.
14. Перегрузка операций. Общие правила и ограничения. Два способа перегрузки операций. Дружественные функции класса.
15. Правила перегрузки операции присваивания.
16. Правила перегрузки операций +, -, *, /.
17. Правила перегрузки операций типа +=.
18. Перегрузка операции индексации и унарных операций.
19. Перегрузка операции ввода/вывода в поток.
20. Перегрузка операций и конструкторы преобразования типов. Класс CData
21. Множественное наследование и иерархия «родитель-ребенок»
22. Наследование и перекрытие методов. Классы Container и Dictionary.
23. Обработка ошибок. Возбуждение и обработка исключений.
24. Шаблоны функций. Механизм конкретизации (инстанцирования) шаблона функций. Вызов шаблонов функций. Перегрузка шаблонов функций. Специализация шаблонов функций. Модели компиляции шаблонов. Примеры шаблонов функций.
25. Шаблоны классов. Объявление и описание классов с шаблонами. Специализация класса с шаблоном. Пример класса с шаблоном.
26. Виртуальные методы. Правила виртуального перекрытия методов. Приведение типов при виртуальном перекрытии методов.
27. Механизм работы виртуальных методов. Ранее и позднее связывание. Примеры применения виртуальных методов.
28. Абстрактные методы и классы. Понятия, правила и ограничения. Примеры абстрактных классов.

Критерии оценивания (оценочное средство - Контрольные вопросы)

| Оценка | Критерии оценивания |
|---------------------|---|
| превосходно | Все практические задания (лабораторные работы) выполнены в полном объеме и в срок, при этом применен творческий подход к решению нестандартных задач. Описаны все этапы выполнения заданий, код и результаты работы представлены преподавателю. |
| отлично | Все практические задания (лабораторные работы) выполнены в полном объеме и в срок. Описаны все этапы выполнения заданий, код и результаты работы представлены преподавателю. |
| очень хорошо | Выполнены основные этапы решения задачи или задача решена с незначительными недочетами. Код и результаты работы представлены преподавателю в срок. |
| хорошо | Выполнены часть этапов решения задачи или задача решена с недочетами. Код и результаты работы представлены преподавателю в срок. |
| удовлетворительно | Выполнены часть этапов решения задачи или задача решена с существенными недочетами. Код и результаты работы представлены преподавателю, но с отклонениями от сроков. |
| неудовлетворительно | Выполнены не все практические задания (лабораторные работы) или выполнены не в полном объеме (представлено не полное описание этапов выполнения заданий, код работает некорректно, результаты работы не представлены преподавателю). |
| плохо | Студент не приступал к выполнению практических заданий. |

6. Учебно-методическое и информационное обеспечение дисциплины (модуля)

Основная литература:

1. Грудзинский Александр Олегович. Методы программирования : Курс на основе языка Object Pascal : учеб. пособие для студентов, обучающихся по направлению 510200 - Прикладная математика и информатика и по специальности 010200 - Прикладная математика и информатика / ННГУ. - Н. Новгород : Изд-во Нижегород. гос. ун-та, 2006. - 392 с. + 1 электрон. опт. диск (CD-ROM). - ISBN 5-85746-903-1 : 150.00., 52 экз.
2. Язык программирования C / Керниган Б.В., Ричи Д.М. - Москва : ИНТУИТ, 2016., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=663682&idb=0>.
3. Язык программирования C++ для профессионалов / Страуструп Б. - Москва : ИНТУИТ, 2016., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=663690&idb=0>.

Дополнительная литература:

1. Язык программирования C++ / Фридман А.Л. - Москва : ИНТУИТ, 2016.,

<https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=663697&idb=0>.

2. Кетков Юлий Лазаревич. Диалог на языке бейсик для мини- и микроЭВМ. - М. : Наука, 1988. - 367 с. - ISBN 5-02-013779-0 (в пер.) : 1.80., 2 экз.

3. Подбельский В.В. Стандартный Си++ : учебное пособие / Подбельский В.В. - Москва : Финансы и статистика, 2022. - 688 с. - ISBN 978-5-00184-081-7., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=869519&idb=0>.

4. Кетков Юлий Лазаревич. Введение в системное программирование на языке ассемблера ЕС ЭВМ : [учеб. пособие для вузов]. - М. : Наука , 1982. - 263 с. : ил. - 0.70., 2 экз.

Программное обеспечение и Интернет-ресурсы (в соответствии с содержанием дисциплины):

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

Используемое лицензионное и свободно распространяемое программное обеспечение:

1. SCILAB - <https://www.scilab.org>
2. Microsoft Visual Studio Community - <https://visualstudio.microsoft.com/ru/>

7. Материально-техническое обеспечение дисциплины (модуля)

Учебные аудитории для проведения учебных занятий, предусмотренных образовательной программой, оснащены мультимедийным оборудованием (проектор, экран), техническими средствами обучения, компьютерами.

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 по направлению подготовки/специальности 02.03.02 - Фундаментальная информатика и информационные технологии.

Автор(ы): Мееров Иосиф Борисович, кандидат технических наук, доцент

Сысоев Александр Владимирович, кандидат технических наук

Карпенко Сергей Николаевич, кандидат технических наук, доцент.

Заведующий кафедрой: Мееров Иосиф Борисович, кандидат технических наук.

Программа одобрена на заседании методической комиссии от 25.06.2025, протокол № Протокол №11.