Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего образования «Новосибирский национальный исследовательский государственный университет» (Новосибирский государственный университет, НГУ)

Высший колледж информатики

Согласовано

Директор ВКИ НГУ

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.Г. Окунев

*подпись*

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_ г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

***Императивное программирование***

направление подготовки: *15.03.06 Мехатроника и робототехника*

направленность (профиль): *Мехатроника и робототехника*

Форма обучения: очная

Разработчики:

*Старший преподаватель* кафедры систем информатики ФИТ

Т.В. Нестеренко,

Старший преподаватель кафедры систем информатики ФИТ

С. Ю .Гатилов

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководитель программы:

д.т. н., Назаров А.Д. . \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Новосибирск, 2020

**Содержание**

[1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы 3](#_Toc58330720)

[2. Место дисциплины в структуре образовательной программы 3](#_Toc58330721)

[3. Трудоемкость дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающегося с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающегося 4](#_Toc58330722)

[4. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий 4](#_Toc58330723)

[5. Перечень учебной литературы 13](#_Toc58330724)

[6. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины 14](#_Toc58330725)

[7. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине 14](#_Toc58330726)

[8. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине 14](#_Toc58330727)

[9. Оценочные средства для проведения текущего контроля и промежуточной аттестации по дисциплине 15](#_Toc58330728)

Приложение 1 Аннотация по дисциплине

Приложение 2 Оценочные средства по дисциплине

# 1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

| Результаты освоения образовательной программы  (компетенции) | В результате изучения дисциплины обучающиеся должны: | | |
| --- | --- | --- | --- |
| знать | уметь | владеть |
| ОПК-3 владение современными информационными технологиями, готовность применять современные средства автоматизированного проектирования и машинной графики при проектировании систем и их отдельных модулей, а также для подготовки конструкторско-технологической документации, соблюдать основные требования информационной безопасности | методы алгоритмизации, языки и технологии программирования, пригодные для практического применения в области моделировании технологических процессов:  *- основные структуры данных и способы их реализации;*  *- алгоритмы решения типовых задач программирования, уметь реализовывать основные структуры данных представленные алгоритмы на языке Си; тестировать программы; использовать стандартные функции языка Си.* | применять методы алгоритмизации, языки и технологии программирования при решении профессиональных задач в области моделировании технологических процессов:  *- формулировать задачу и осуществлять поиск в интернет-ресурсах и библиографических материалах, быстро и эффективно реализовывать классические алгоритмы, использовать систему тестирования NSUts;*  *- владеть навыками работы в текстовых редакторах, интегрированных средах разработки программ, системе NSUts.* | навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач в области моделировании технологических процессов:  *- знать классы сложности типовых алгоритмов; способы оценки объема памяти и быстродействия программ; уметь оценивать объем памяти и быстродействие программ; тестировать программы; использовать автоматизированную систему NSUts.* |

# 2. Место дисциплины в структуре образовательной программы

Дисциплины (практики), изучение которых необходимо для освоения дисциплины *Императивное программирование:* школьный курс Математика, Информатика и ИКТ.

Дисциплина *Императивное программирование* является базовой для освоения дисциплин «Объектно-ориентированное программирование», «Модели вычислений», «Разработка программно-аппаратного комплекса для решения научных и прикладных задач (групповой проект)».

# 3. Трудоемкость дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающегося с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающегося

Трудоемкость дисциплины 11 з.е. (396 ч)

Форма промежуточной аттестации: 1 семестр – экзамен, 2 экзамен – экзамен

|  |  |  |  |
| --- | --- | --- | --- |
| № | Вид деятельности | Семестр | |
| 1 | 2 |
| 1 | Лекции, ч | 32 | 32 |
| 2 | Практические занятия, ч | 32 | 32 |
| 3 | Лабораторные работы, ч | 32 | 32 |
| 4 | Занятия в контактной форме, ч  из них | 100 | 100 |
| 5 | из них аудиторных занятий, ч | 96 | 96 |
| 6 | в электронной форме, ч | - | - |
| 7 | консультаций, час. | 2 | 2 |
| 8 | промежуточная аттестация, ч | 2 | 2 |
| 9 | Самостоятельная работа, час. | 80 | 116 |
| 10 | Всего, ч | 180 | 216 |

# 4. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

***1 семестр***

**Лекции (32 ч)**

|  |  |
| --- | --- |
| Наименование темы и их содержание | Объем,  час |
| Лекция 1. Основные вопросы по Си  Краткий обзор базовых типов языка Си. Преобразования между базовыми типами. Связь понятия указателя и массива.  Глобальные и локальные переменные. Модификаторы области видимости и времени жизни.  Разбор порядка вычисления выражения, приоритеты операций | 2 |
| Лекция 2. Основные вопросы по Си  Понятие типа/преобразование типов и структуры. Инициализатор структур.  Функции. Порядок передачи параметров через стек.  Понятие указателя в Си.Операции над указателями.  Строки в языке Си. Функции работы со строками: определение длины строки, копирование строк, слияние строк. | 2 |
| Лекция 3. Основные вопросы по Си  Основные стандартные функции языка Си для работы с файлами. Текстовые и бинарные файлы.  Динамическая память. Функции работы с ДП. | 2 |
| Лекция 4. Системы счисления  Рекуррентные функции перевода "число-запись" и "запись-число"; вычисления по схеме Горнера с минимальным числом операций; алгоритмы перевода целых чисел из одной b-с.с. в другую. Модификации этих алгоритмов для перевода действительных чисел. | 2 |
| Лекция 5. Оценка сложности вычислительных алгоритмов, общие методы решения вычислительных задач  Размер задачи как характеристика объема входных данных. Временная и емкостная сложность программы как функции размера задачи. Верхняя, нижняя и средняя оценки. Классы эффективности алгоритмов: примеры задач, допускающих решение за константное, логарифмическое, линейное, квадратичное, полиномиальное, экспоненциальное время.  Основные методы программирования: повторение (цикл), ветвление и рекурсия. Рекурсивный переход, правила выхода, ветвящаяся и хвостовая рекурсия. Алгоритмы с возвратом. | 2 |
| Лекция 6. Перестановки набора, методы поиска всех перестановок  Перестановки набора. Подсчет числа перестановок. Инверсии. Число инверсий, как мера сложности упорядочения набора; таблицы инверсий; алгоритм восстановления перестановки по таблице инверсий.  Таблица инверсий как число в особой с.с.; итерационный алгоритм генерации всех таблиц инверсий. Алгоритмы перебора перестановок: рекурсивный, итерационный с лексикографическим упорядочением (Дейкстры). | 2 |
| Лекция 7. Задача поиска, методы поиска в массиве и подстроки в строке  Постановка задач поиска в произвольном наборе данных. Методы простого поиска в массиве: линейный поиск, бинарный поиск, оценки сложности.  Методы поиска подстроки в строке: алгоритм прямого перебора. | 2 |
| Лекция 8. Задача поиска, методы поиска в массиве и подстроки в строке  Алгоритм Бойера-Мура, Рабина-Карпа. Оценки сложности. | 2 |
| Лекция 9. Задача сортировки массивов, простые и улучшенные методы сортировки  Постановка задач сортировки (С) записей в произвольном наборе данных. Внешняя и внутренняя постановка задачи Си. Методы сортировки массива: метод простых вставок; метод бинарных вставок; метод простого выбора; метод "пузырька"; шейкер-сортировка; метод Шелла. Оценки сложности. | 2 |
| Лекция 10. Задача сортировки массивов, простые и улучшенные методы сортировки  Быстрая сортировка Хоара; сортировка деревом, пирамидальная сортировка; реализация кучи. Оценки сложности. | 2 |
| Лекция 11. Управление памяти в Си, бинарные файлы  Основные ошибки: утечка памяти, висящие указатели. Стратегии выделения памяти. Внешняя и внутренняя фрагментация. Выделение памяти алгоритмом парных меток. Выделение памяти алгоритмом близнецов. Сборка мусора: область применения, основные методы. | 2 |
| Лекция 12. Классические модели динамической памяти: список, стек, очередь; операции, способы реализации, задачи  Список; как универсальная модель линейно упорядоченных структур данных последовательного доступа; разновидности списков: одно/двусвязные; циклические; иерархические Операции над списками, алгоритмы поиска и включения для списков, анализ их эффективности, способы реализации списков (статика, динамика), реализация алгоритма топологической сортировки на иерархических списках. | 2 |
| Лекция 13. Классические модели динамической памяти: список, стек, очередь; операции, способы реализации, задачи  Стек, преобразование инфиксной формы записи выражения в постфиксную и вычисление значения полученного выражения.  Очередь, Основные операции, способы реализации на различных базовых представлениях. | 2 |
| Лекция 14. Бинарные файлы, сортировка файлов  Бинарные файлы | 2 |
| Лекция 15. Бинарные файлы, сортировка файлов  Cортировка файлов простым двухпутевым слиянием | 2 |
| Потоковая контрольная | 2 |
| **Итого:** | **32** |

**Практические занятия (32)**

|  |  |
| --- | --- |
| Содержание практического занятия | Объем, час |
| Тема 1. Основные вопросы по Си. Основные типы данных (целые и вещественные) и операции над ними. Выражения. Операторы присваивания, условный, выбора. Условное выражение. Вид программы на Си. Решение задач. | 2 |
| Тема 2. Основные вопросы по Си. Циклы for, while, do-while. Операторы выхода из цикла break и continue. Решение задач. | 2 |
| Тема 3. Основные вопросы по Си. Линейные массивы. Описание одномерных массивов, инициализация. Операции над одномерными массивами. Задание массива случайным образом. Прямой поиск в массиве. Поиск методом деления пополам | 2 |
| Тема 4. Основные вопросы по Си. Функции. Описание функций. Параметры, области видимости переменных. Виды переменных (локальные, глобальные, статические, автоматические, регистровые, внешние). Передача массива параметром в функцию. Рекурсивные функции (вычисление факториала, Ханойские башни). Алгоритм Евклида (итеративный и рекурсивный). Реализация. | 2 |
| Тема 5. Основные вопросы по Си. Литерный тип. Описание литерного типа, литерные константы. Таблица ASCII. Функции ввода-вывода символов. Решение задач | 2 |
| Тема 6. Основные вопросы по Си. Строки. Описание строк, инициализация. Функции ввода-вывода. Пример реализации некоторых стандартных функций: strlen, strcmp, strcpy. | 2 |
| Тема 7. Основные вопросы по Си. Перевод из одной системы счисления в другую. Кратные системы счисления, особенности реализации на строках. | 2 |
| Тема 8. Основные вопросы по Си. Представление числовых данных в Си, байты и биты, битовые операции, реализация битовых массивов, битовые поля. | 2 |
| Тема 9. Многомерные массивы. Описание многомерного массива, операции над ним, инициализация. Ввод массива из файла. Передача массива в качестве параметра. Решение задач | 2 |
| Тема 10. Файлы, описание, виды файлов, работа с текстовыми файлами, решение задач | 2 |
| Тема 11. Структуры в Си, описание, инициализация, способы работы со структурами, решение задач | 2 |
| Тема 12. Динамическая память, функции работы с динамической памятью, выделение памяти, реализация динамических массивов, решение задач. | 2 |
| Тема 13. Динамические списки, способы создания и обходов. Преобразование списков.. | 2 |
| Тема 14. Линейные списки, стеки, очереди. Способы их реализации на динамических списках и на массивах | 2 |
| Тема 15. Описание аргументов функции main. Создание проекта. | 2 |
| Тема 16. Дифференцированный зачет | 2 |
| **Итого:** | **32** |

**Лабораторные работы (32 ч)**

|  |  |
| --- | --- |
| Содержание лабораторных работ | Объем, час |
| Тема 1. Основные вопросы по Си, управление памятью. Задачи без циклов, числовые типы данных. Реализация задач:  1. Треугольник  2. Скорая помощь  3. Корни  4. Цифра | 2 |
| Тема 2. Основные вопросы по Си, управление памятью. Задачи с циклами, числовые типы данных. Реализация задач:  1. Сумма ряда  2. Обращение числа  3. Простые числа  4. Близкий к целому  5. Цифра | 2 |
| Тема 3 Задача поиска, методы поиска в массиве. Одномерные массивы. Реализация задач:  1. Границы массива  2. Решето Эратосфена  3. Даты  4. Бинарный поиск  5. Подпоследовательности  6. Различные числа  7. Циклический сдвиг  8. Бинарный поиск 2 | 2 |
| Тема 4. Задача сортировки массивов, простые методы сортировки. Функции, рекурсия, простые сортировки. Реализация задач:  1. Сортировка выбором  2. Сортировка вставками  3. Шейкер-сортировка  4. НОД  5. НОК | 2 |
| Тема 5. Основные вопросы по Си, управление памятью. Функции, рекурсия, простые сортировки. Реализация задач:  1. Сортировка выбором  2. Сортировка вставками  3. Шейкер-сортировка  4. НОД  5. НОК | 2 |
| Тема 6. Основные вопросы по Си, управление памятью. Строки. Реализация задач:  1. Игра в алфавит  2. Смешивание цветов  3. Топики  4. Разность множеств  5. Слова | 2 |
| Тема 7. Системы счисления (с.с.). Перевод из одной системы счисления в другую. Реализация задач:  1. Целое: значение -> запись  2. Целое: запись -> значение  3. Дробное: значение -> запись  4. Дробное: запись -> значение  5. Запись -> запись | 2 |
| Тема 8. Модели машинной арифметики с конечной разрядностью. Биты и байты. Реализация задач:  1. Количество единиц  2. Обращение числа  3. Битовый массив | 2 |
| Тема 9. Многомерные массивы. Многомерные массивы. Реализация задач:  1. Японский кроссворд  2. Шифр Шерлока Холмса. | 2 |
| Тема 10. Перестановки набора, методы поиска всех перестановок. Перестановки. Реализация задач:  1. Следующая по алфавиту перестановка  2. Таблица инверсий  3. Восстановление перестановки. | 2 |
| Тема 11. Задача сортировки массивов, улучшенные методы сортировки. Улучшенные сортировки. Реализация:  1. Быстрая сортировка  2. Сортировка Шелла  3. Пирамидальная сортировка | 2 |
| Тема 12. Основные вопросы по Си, управление памятью. Текстовые файлы. Реализация задач:  1. Арифметическая прогрессия  2. Количество символов  3. Количество слов  4. Количество строк. | 2 |
| Тема 13. Классические модели динамической памяти: список, стек, очередь; операции, способы реализации, задачи. Односвязные списки. Реализация задач:  1. Построение списка добавлением в хвост  2. Построение списка добавлением в голову  3. Построение упорядоченного списка. | 2 |
| Тема 14. Классические модели динамической памяти: список, стек, очередь; операции, способы реализации, задачи. Преобразование списков. Реализация задач:  1. Удаление из списка  2. Задача Иосифа  3. Слияние списков. | 2 |
| Тема 15. Классические модели динамической памяти: список, стек, очередь; операции, способы реализации, задачи. Линейные списки: стеки, очереди. Задачи:  1. Перевод из инфиксной формы в постфиксную  2. Вычисление на стеке  3. Циклический буфер. | 2 |
| Тема 16. Классические модели динамической памяти: список, стек, очередь; операции, способы реализации, задачи. Реализация проекта Калькулятор. | 2 |
| **Итого:** | **32** |

**Самостоятельная работа студентов (80 ч)**

|  |  |
| --- | --- |
| Перечень занятий на СРС | Объем, час |
| Изучение темы дисциплины по учебной литературе, учебным пособиям, поиск в интернете. Изучение предлагаемых теоретических разделов в соответствии с настоящей Программой. Учебно-методические материалы по дисциплине «Императивное программирование» выложены на странице курса в сети Интернет . | 14 |
| Подготовка к практическим занятиям, к лабораторным работам, к текущему контролю знаний и промежуточной аттестации. Разбор решенных задач, самостоятельное решение задач, реализация проекта | 30 |
| Подготовка к экзамену. Повторение теоретического материала по вопросам, совпадающим с темами лекций. | 36 |
| **Итого:** | **80** |

***2 семестр***

**Лекции (32 ч)**

|  |  |
| --- | --- |
| Наименование темы и их содержание | Объем,  час |
| Лекция 1. Динамическое программирование  Динамическое программирование как решение задач с помощью табличной техники. Примеры задач: о делимости, о гангстерах. Алгоритм Ахо, задача о расстановке скобок в выражении. | 2 |
| Лекция 2. Динамическое программирование  Задачи о рюкзаке, метод ветвей и границ. Задача об умножении матриц. | 2 |
| Лекция 3. Абстрактные структуры данных: графы, деревья, способы реализации  Графы, пути и маршруты, модели представления в ЭВМ: матрицы смежности и инцидентности, динамическая структура со списками дуг, табличное представление. Дерево, как частный вид графа. Дерево двоичного поиска, создание орфографического словаря по заданному набору слов. | 2 |
| Лекция 4. Абстрактные структуры данных: графы, деревья, способы реализации  Уравновешенное дерево. Алгоритмы включения и удаления, сравнение их эффективности с известными алгоритмами на массивах. Левое/правое скобочное представление деревьев. Дерево отрезков, реализация, оценки. | 2 |
| Лекция 5. Алгоритмы перебора на абстрактных динамических структурах: обходы, минимальные пути, остовные деревья, двусвязные компоненты графа, реализация  Обход вершин графа методом поиска в глубину, реализация. Свойства поиска в глубину, использование стека. Двусвязные компоненты графа, реализация алгоритма поиска двусвязных компонент и точек сочленения графа. Обходы деревьев в глубину: в ре/пост/инфиксном порядке, связь с выражениями. | 2 |
| Лекция 6. Алгоритмы перебора на абстрактных динамических структурах: обходы, минимальные пути, остовные деревья, двусвязные компоненты графа, реализация  Обход графа методом в ширину. Реализация с помощью очереди, примеры задач. Кратчайшие ипути в лабиринте. Обходы деревьев в ширину слева направо. | 2 |
| Лекция 7. Алгоритмы перебора на абстрактных динамических структурах: обходы, минимальные пути, остовные деревья, двусвязные компоненты графа, реализация  Топологическая сортировка, реализация на матрице смежности и на еирархических списках. Поиск кратчайших путей в графе, алгоритмы Дейкстры, Беллмана-Форда, Флойда-Уоршелла. Транзитивное замыкание графа. Алгоритм Флойда. Реализация алгоритмов в том числе на куче, оценки сложности. | 2 |
| Лекция 8. Алгоритмы перебора на абстрактных динамических структурах: обходы, минимальные пути, остовные деревья, двусвязные компоненты графа, реализация  Каркас графа минимальной стоимости, алгоритмы Краскала, Прима, Прима-Краскала. Система непересекающихся множеств, реализация, оценки сложности. | **2** |
| Лекция 9. Элементы теорий вероятностей, информации и кодирования, реализация проекта  Модель информационной системы связи Шеннона. Кодирование как преобразование сообщения при переходе в среду с другим алфавитом. Префиксный код. Однозначная декодируемость префиксного кода. Примеры непрефиксного однозначно декодируемого кода. | **2** |
| Лекция 10. Элементы теорий вероятностей, информации и кодирования, реализация проекта  Формулы Шеннона, Хартли. Избыточность кодирования. Сжатие по Хаффману. Реализация проекта проекта по созданию архиватора. | 2 |
| Лекция 11. В-деревья, красно-черные деревья, реализация проекта  В-деревья, основные определения. Примеры. Реализация на файлах. Алгоритм поиска вершины. | 2 |
| Лекция 12. В-деревья, красно-черные деревья, реализация проекта  Алгоритмы разбиения вершины, добавления и удаления элемента, реализация, оценки. Реализация проекта. | 2 |
| Лекция 13. Раскраски графа, гамильтоновы и Эйлеровы циклы, алгоритмы и реализация.  Раскраска графа. Задача о составлении расписаний, о распределении ресурсов. Алгоритм последовательной раскраски. Эйлеров граф. Алгоритм Флери, реализация, оценки. Поиск Эйлерова цикла за время O(n+m), реализация. Гамильтонов цикл, реализация перебором. | 2 |
| Лекция 14. Потоки в сетях, алгоритмы поиска максимального потока, реализация  Максимальный поток в сети, примеры, сокращение потока, несколько истоков, остаточные сети, дополняющие пути, разрезы. Метод Форда-Фалкерсона, реализация, оценки. | 2 |
| Лекция 15. Потоки в сетях, алгоритмы поиска максимального потока, реализация  Задача о максимальном паросочетании, реализация. Алгоритм проталкивания предпотока, реализация, оценки. | 2 |
| Лекция 16. Элементы теории языков  Символьные формализмы для описания синтаксиса языков: БНФ, РБНФ. Формальные грамматики. Синтаксический анализатор. Нисходящий и восходящий разбор. Полный перебор правил подстановки. | 2 |
| **Итого:** | **32** |

**Практические занятия (32)**

|  |  |
| --- | --- |
| Содержание практического занятия | Объем, час |
| Тема 1. Динамическое программирование.  Функции с переменным числом параметров (доклад студента)..  Решение задач на списки. | 2 |
| Тема 2. Динамическое программирование.  Динамическое программирование. Решение простых задач, обсуждение задач для самостоятельного решения. | 2 |
| Тема 3 Абстрактные структуры данных: графы, деревья, способы реализации.  Деревья, представление деревьев, обходы, способы реализации.  Задачи на создание и обходы деревьев. | 2 |
| Тема 4 Абстрактные структуры данных: графы, деревья, способы реализации.  Способы представления графов. Перевод из одного представления в другое. Способы построения сбалансированного дерева. | 2 |
| Тема 5. Алгоритмы перебора на абстрактных динамических структурах: обходы, минимальные пути, остовные деревья, реализация Метод поиска в глубину. Задача – проверить, является ли граф, заданный матрицей смежности, деревом. | 2 |
| Тема 6. Алгоритмы перебора на абстрактных динамических структурах: обходы, минимальные пути, остовные деревья, реализация Обход дерева в ширину с использованием очереди. Обход графа в ширину | 2 |
| Тема 7. Алгоритмы перебора на абстрактных динамических структурах: обходы, минимальные пути, остовные деревья, реализация Топологическая сортировка, три метод реализации: на матрице смежности, на иерархических списках, методом поиска в глубину. | 2 |
| Тема 8. Алгоритмы перебора на абстрактных динамических структурах: обходы, минимальные пути, остовные деревья, реализация Поиск кратчайших путей в графе. Алгоритм Дейкстры. Реализация кучи. | 2 |
| Тема 9. Алгоритмы перебора на абстрактных динамических структурах: обходы, минимальные пути, остовные деревья, реализация Транзитивное замыкание графа и алгоритм Флойда. Методы реализации. | 2 |
| Тема 10. Алгоритмы перебора на абстрактных динамических структурах: обходы, минимальные пути, остовные деревья, реализация Система непересекающихся множеств. Использование СНМ в алгоритме Краскала построения минимального остовного дерева. Использование кучи в алгоритме Прима. | 2 |
| Тема 11. Элементы теорий вероятностей, информации и кодирования, реализация проекта  Обсуждение реализации проекта Архиватор. Построение кодирующего дерева. Работа с бинарными файлами (доклад студента). Способы записи и воспроизведения закодированной информации. | 2 |
| Тема 12. В-деревья, красно-черные деревья, реализация проекта.  Обсуждение проекта Каталог на B-деревьях.  Директивы препроцессора (доклад студента). | 2 |
| Тема 13. В-деревья, красно-черные деревья, реализация проекта  Красно-черные деревья, способы реализации.  Тест на знание языка Си. | 2 |
| Тема 14. Раскраски графа, гамильтоновы и Эйлеровы циклы, алгоритмы и реализация. Способы поиска Эйлеровых циклов в графе, реализация на стеке.  Задачи полного перебора на примере поиска гамильтонова цикла в графе. | 2 |
| Тема 15. Двусвязные компоненты графа. Реализация алгоритма последовательной раскраски графа. Использование метода поиска в глубину для поиска двусвязных компонент и точек сочленения | 2 |
| Контрольная работа. | 2 |
| **Итого:** | **32** |

**Лабораторные работы (32 ч)**

|  |  |
| --- | --- |
| Содержание лабораторных работ | Объем, час |
| Тема 1. Динамическое программирование. Реализация функции с переменным числом параметров my\_printf. | 2 |
| Тема 2-3. Динамическое программирование .  Задачи: 1. Рюкзак 2. Делимость 3. Гангстеры 4. Телефонный номер | 4 |
| Тема 4. Абстрактные структуры данных: графы, деревья, способы реализации . Деревья.  Задачи: 1. Дерево двоичного поиска 2. Обходы дерева 3. Дерево-формула | 2 |
| Тема 5. Алгоритмы перебора на абстрактных динамических структурах: обходы, минимальные пути, остовные деревья, реализация.  1. Компоненты связности  2. Лабиринт  3. Знакомства. | 2 |
| Тема 6. Алгоритмы перебора на абстрактных динамических структурах: обходы, минимальные пути, остовные деревья, реализация Обход графа в ширину. Задачи:  1. Лабиринт  2. Встреча шаманов | 2 |
| Тема 7. Алгоритмы перебора на абстрактных динамических структурах: обходы, минимальные пути, остовные деревья, реализация Топологическая сортировка. Задачи:  1. Наложение рамок  2. Топологическая сортировка | 2 |
| Тема 8-9. Алгоритмы перебора на абстрактных динамических структурах: обходы, минимальные пути, остовные деревья, реализация Кратчайшие пути в графе. Задачи:  1. Шлю я за пакетом пакет…  2. Поиск пути  3. Сплетницы  4. Авиаперелеты | 4 |
| Тема 10. Алгоритмы перебора на абстрактных динамических структурах: обходы, минимальные пути, остовные деревья, реализация Топологическая сортировка. Задачи:  Транзитивное замыкание графа. Задачи:  1. Спички  2. Фиолетовое такси | 2 |
| Тема 11-12. Алгоритмы перебора на абстрактных динамических структурах: обходы, минимальные пути, остовные деревья, реализация Топологическая сортировка. Задачи:  Каркасы в графе, СНМ. Задачи:  1. Highways  2. Дорожки | 4 |
| Тема 13-15. Элементы теории вероятностей, информации и кодирования, реализация проекта  В-деревья, красно-черные деревья, реализация проекта Работа над одним из проектов:  − Архиватор Хаффмана,  − Каталог файлов (B-деревья) | 6 |
| Сдача проекта | 2 |
| **Итого:** | **32** |

**Самостоятельная работа студентов (116 ч)**

|  |  |
| --- | --- |
| Перечень занятий на СРС | Объем, час |
| Изучение темы дисциплины по учебной литературе, учебным пособиям, поиск в интернете. Изучение предлагаемых теоретических разделов в соответствии с настоящей Программой. Учебно-методические материалы по дисциплине «Императивное программирование» выложены на странице курса в сети Интернет . | 20 |
| Подготовка к практическим занятиям, к лабораторным занятиям, к текущему контролю знаний и промежуточной аттестации. Разбор решенных задач, самостоятельное решение задач, реализация проекта. | 60 |
| Подготовка к экзамену. Повторение теоретического материала по вопросам, совпадающим с темами лекций. | 36 |
| **Итого:** | **116** |

# 5. Перечень учебной литературы

***5.1 Основная литература***

1. Стандарт языка Си ISO/IEC 9899 – URL: http://www.open-std.org/jtc1/sc22/wg14/www/standards.html#9899
2. Стандарт машинной арифметики IEEE 754 – URL: <http://www.eecs.berkeley.edu/~wkahan/ieee754status/IEEE754.PDF>

***5.2 Дополнительная литература***

1. *Керниган, Б.В.*, *Д.М. Ричи* Язык программирования C / М. : Интернет-Университет Информационных Технологий, 2006. - 272 с. ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=234039>.
2. *Ахо, Альфред В.* Построение и анализ вычислительных алгоритмов / А. Ахо, Дж. Хопкрофт, Дж. Ульман ; пер. с англ. А.О. Слисенко ; под ред. Ю.В. Матиясевича. Москва : Мир, 1979. 536 с. : ил. ; 22 см.
3. Ахо, Альфред В. Теория синтаксического анализа, перевода и компиляции : [в 2 т.] / А. Ахо, Дж. Ульман ; пер. с англ. под ред. В.М. Курочкина. Москва : Мир, 1978. ; 22 см. Т.1: Синтаксический анализ / пер. В.Н. Агафонова. 1978. 612 с. : схемы.
4. Подбельский, В.В. Курс программирования на языке Си : учебник / В.В. Подбельский, С.С. Фомин. - Москва : ДМК Пресс, 2012. - 384 с. - ISBN 978-5-94074-449-8 ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=232130>
5. Вирт, Н. Алгоритмы и структуры данных. Новая версия для Оберона / Н. Вирт ; пер. с англ. под ред. Ф.В. Ткачева. - Москва : ДМК Пресс, 2010. - 272 с. : ил. - ISBN 978-5-94074-584-6 ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=86483>

# 6. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

Для освоение дисциплины используются следующие ресурсы:

- электронная информационно-образовательная среда НГУ (ЭИОС);

- образовательные интернет-порталы;

- информационно-телекоммуникационная сеть Интернет.

Взаимодействие обучающегося с преподавателем (синхронное и (или) асинхронное) осуществляется через личный кабинет студента в ЭИОС.

***6.1 Современные профессиональные базы данных:***

- Полнотекстовые журналы Springer Journals за 1997-2015 г., электронные книги (2005-2016 гг.), коллекция научных биомедицинских и биологических протоколов SpringerProtocols, коллекция научных материалов в области физических наук и инжиниринга SpringerMaterials, реферативная БД по чистой и прикладной математике zbMATH.

- Электронные ресурсы Web of Science Core Collection (Thomson Reuters Scientific LLC.), Journal Citation Reports + ESI

- БД Scopus (Elsevier)

***6.2. Информационные справочные системы***

***-*** <http://www.spsl.nsc.ru> - портал ГПНТБ СОРАН

- <https://olympic.nsu.ru/nsuts-new/login.cgi> - Система тестирования NSUts

- электронная библиотека диссертаций Российской государственной библиотеки (ЭБД РГБ)

# 7. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине

***7.1 Перечень программного обеспечения***

Для обеспечения реализации дисциплины *Императивное программирование*» используется стандартный комплект программного обеспечения (ПО), включающий регулярно обновляемое лицензионное ПО Windows и MS Office.

# 8. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине

Для реализации дисциплины *Императивное программирование* используются специальные помещения:

1. Учебные аудитории для проведения занятий лекционного типа, занятий семинарского типа, лабораторных работ, групповых и индивидуальных консультаций, текущего контроля, промежуточной аттестации;

2. Помещения для самостоятельной работы обучающихся.

Учебные аудитории укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду НГУ.

Материально-техническое обеспечение образовательного процесса по дисциплине *Императивное программирование* для обучающихся из числа лиц с ограниченными возможностями здоровья осуществляется согласно «Порядку организации и осуществления образовательной деятельности по образовательным программам для инвалидов и лиц с ограниченными возможностями здоровья в Новосибирском государственном университете».

# 9. Оценочные средства для проведения текущего контроля и промежуточной аттестации по дисциплине

Перечень результатов обучения по дисциплине *Императивное программирование* и индикаторов их достижения представлен в виде знаний, умений и владений в разделе 1.

***9.1 Порядок проведения текущего контроля и промежуточной аттестации по дисциплине***

***Текущий контроль успеваемости:***

по дисциплине *Императивное программирование* осуществляется на практических занятиях и лабораторных работах и заключается в выполнении различного рода заданий (прием задач, тестирование в системе NSUts); контрольная работа, выполнение проекта.. По результатам текущей аттестации выставляется оценка «зачтено» или «не зачтено». Для получения оценки «зачтено» задание должно быть выполнено и защищено в полном соответствии с предъявляемыми требованиями.

***Промежуточная аттестация:***

(итоговая по дисциплине) проводится по завершению каждого периода ее освоения (семестра) в виде проведения экзамена. Экзамен проводятся в аудитории, студентам разрешено пользоваться бумагой для записей и авторучкой. Справочной, учебной и другой литературой пользоваться не разрешается. Использование электронных устройств (телефоны, любые виды компьютеров, т.д.) запрещено.

Оценка промежуточной аттестации для экзамена выставляется по результатам заданий, входящих в портфолио:

− на основании выполнения заданий по лабораторным работам;

− по результату защиты проекта.

и по результату устного ответа на экзаменационный билет;

Результаты промежуточной аттестации дисциплины «*Императивное программирование*» определяются оценками «отлично», «хорошо», «удовлетворительно», «неудовлетворительно». Оценки «отлично», «хорошо», «удовлетворительно» означают успешное прохождение промежуточной аттестации.

***Описание критериев и шкал оценивания индикаторов достижения результатов обучения по дисциплине*** «*Императивное программирование*»

Таблица 9.1

|  |  |  |
| --- | --- | --- |
| **Код компетенции** | **Результат обучения по дисциплине** | **Оценочное средство** |
| ОПК-3 | Знать основные структуры данных и способы их реализации.  Знать алгоритмы решения типовых задач программирования, уметь реализовывать основные структуры данных представленные алгоритмы на языке Си; тестировать программы; использовать стандартные функции языка Си. | Портфолио  Экзамен |
| Уметь формулировать задачу и осуществлять поиск в интернет-ресурсах и библиографических материалах, быстро и эффективно реализовывать классические алгоритмы, использовать систему тестирования NSUts.  Уметь владеть навыками работы в текстовых редакторах, интегрированных средах разработки программ, системе NSUts.. | Портфолио  Экзамен |
| Знать классы сложности типовых алгоритмов; способы оценки объема памяти и быстродействия программ; уметь оценивать объем памяти и быстродействие программ; тестировать программы; использовать автоматизированную систему NSUts. | Портфолио  Экзамен |

Таблица 9.2

|  |  |
| --- | --- |
| **Критерии оценивания результатов обучения** | **Шкала**  **оценивания** |
| **Портфолио**  Необходимым условием для прохождения промежуточной аттестации является оценка «зачтено» по результатам выполненного портфолио.  Для оценивания портфолио студенту необходимо сдать все лабораторные работы и выполнить проект.  **Экзамен:**  Ответы на вопросы билета.  Студент демонстрирует полное понимание и обоснование теоретических результатов изучаемой темы, приводит примеры из изучаемой темы. Показывает умение применять теоретические результаты в различных ситуациях, эффективно тестирует программы. Может реализовать основные и усложненные структуры данных и алгоритмы на языке Си; имеет навыки работы языка Си и со стандартными библиотеками. Знает классические алгоритмы, способы их решения, авторов, литературу; основы устройства и функционирования операционных систем. | *Отлично* |
| **Портфолио**  Необходимым условием для прохождения промежуточной аттестации является оценка «зачтено» по результатам выполненного портфолио.  Для оценивания портфолио студенту необходимо сдать все лабораторные работы и выполнить проект.  **Экзамен:**  Ответы на вопросы билета.  Студент демонстрирует понимание и обоснование теоретических результатов изучаемой темы, частично приводит примеры из изучаемой темы. Показывает умение применять теоретические результаты в различных ситуациях, эффективно тестирует программы. Может реализовать основные и усложненные структуры данных и алгоритмы на языке Си; имеет навыки работы языка Си и со стандартными библиотеками. Знает классические алгоритмы, способы их решения, авторов, литературу; основы устройства и функционирования операционных систем. | *Хорошо* |
| **Портфолио**  Необходимым условием для прохождения промежуточной аттестации является оценка «зачтено» по результатам выполненного портфолио.  Для оценивания портфолио студенту необходимо сдать более 70% лабораторные работы и выполнить проект.  **Экзамен:**  Ответы на вопросы билета.  Студент демонстрирует частичное понимание изучаемой темы, частично приводит примеры из изучаемой темы. С трудом показывает умение применять теоретические результаты в различных ситуациях. Не может формализовать задачу. Может реализовать основные структуры данных и алгоритмы на языке Си. Знает классические алгоритмы, способы их решения, авторов, литературую | *Удовлетворительно* |
| **Портфолио**  Необходимым условием для прохождения промежуточной аттестации является оценка «зачтено» по результатам выполненного портфолио.  Для оценивания портфолио студенту необходимо сдать все лабораторные работы и выполнить проект. Сдано менее 50% лабораторных задач, проект не выполнен.  **Экзамен:**  Отсутствие ответов на вопросы билета и на дополнительные вопросы. Студент не может дать определение и привести пример для понятия из списка базовых понятий. Демонстрирует слабые знания структур данных, не умеет использовать стандартные библиотеки языка Си. Не знает классические алгоритмы и способы их применения. | *Неудовлетво-рительно* |

***Типовые контрольные задания и иные материалы, необходимые для оценки результатов обучения***

.Примеры оценочных средств

*Пример экзаменационного билета*

1. Структуры в Си. Синтаксис описания структур. Обращение к полям структур для объектов и к полям по указателю на объект типа структура. Инициализатор структур. Примеры использования структур в алгоритмах.

2. Система непересекающихся множеств (СНМ). Реализация алгоритма Прима с использованием СНМ.

3. В компании из *N* человек составили список пар знакомых. Пусть *N* = 13. После того, как всех пронумеровали, получились следующие пары знакомых друг с другом:

(1, 2), (2, 7), (3, 11), (5, 3), (7, 10), (8, 4), (2, 9), (6, 5), (8, 12)

Доопределим понятие «знакомы» следующим образом. Будем считать двух человек знакомыми, если они знакомы лично, либо у них есть общий знакомый.

Определить минимальное возможное число людей *K*, такое, чтобы при любом выборе *К* человек из *N* имеющихся среди них гарантированно встретились хотя бы двое знакомых (лично или опосредованно).

Требуется:

* найти *K* для данного примера,
* описать алгоритм решения задачи для любого *N* и произвольного непустого набора пар знакомых.

Примеры задач

*Пример задачи, которую требуется сдать в автоматизированную систему тестирования:*

|  |  |
| --- | --- |
| *Имя входного файла:* | *input.txt* |
| *Имя выходного файла:* | *output.txt* |
| *Ограничение по времени:* | *1 секунда на тест* |
| *Ограничение по памяти:* | *64 Мб* |

Задача «Спички» (Транзитивное замыкание графа)

Во время игры какое-то количество спичек рассыпается по столу, и игроки должны убрать их, выбирая по одной, не сдвинув с места остальные спички. Вам нужно определить, касаются ли данные спички друг друга. Вам будет дан список координат концов спичек, и нужно будет определить, соединены ли две данные спички или нет. Заметим, что касание — это соединение, и что две спички могут быть соединены, если они соединены не прямо, а посредством других спичек.

Входные данные: Входной файл сдержит несколько тестов. В первой строке каждого теста находится целое число *n* (1 < *n* < 13), задающее количество спичек на столе. Каждая из следующих n строк содержит 4 положительных целых x1, y1, x2, y2, обозначающих координаты (x1, y1), (x2, y2)концов одной спички. Все координаты меньше 100. (Заметим, что спички могут быть разной длины). Первая введенная спичка будет спичкой номер 1, вторая, соответственно, номер 2 и т.д. остальные строки теста содержат по два целых числа a и b (от 1 до n, включительно). Вам нужно определить, связана ли спичка a со спичкой b, или нет. Конец теста: a = b = 0. Все данные корректны и нет спичек нулевой длины. После каждого теста во входном файле пустая строка. Последняя строка в файле содержит один символ — 0.

Выходные данные: Вам нужно для каждой пары спичек сгенерировать строчку CONNECTED, если данные спички связаны, и NOT CONNECTED, наоборот. Будем считать, что спичка связана сама с собой. Пустые строки между тестами не вставлять.

Пример входных данных:

|  |  |
| --- | --- |
| ***input.txt*** | ***output.txt*** |
| **7**  **1 6 3 3**  **4 6 4 9**  **4 5 6 7**  **1 4 3 5**  **3 5 5 5**  **5 2 6 3**  **5 4 7 2**  **1 4**  **1 6**  **3 3**  **6 7**  **2 3**  **1 3**  **0 0**  **0** | **CONNECTED**  **NOT CONNECTED**  **CONNECTED**  **CONNECTED**  **NOT CONNECTED**  **CONNECTED** |

Оценочные материалы по промежуточной аттестации (приложение 2), предназначенные для проверки соответствия уровня подготовки по дисциплине требованиям ФГОС, хранятся на кафедре-разработчике РПД в печатном и электронном виде.

**Лист актуализации рабочей программы дисциплины**

**«*Императивное программирование*»**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Характеристика внесенных изменений (с указанием пунктов документа) | Дата и №  протокола Ученого совета ВКИ | Подпись  ответственного |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |