

Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение высшего
профессионального образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТОМСКИЙ
ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

УТВЕРЖДАЮ
Зам. директора Института кибернетики
по учебной работе

_____ С.А. Гайворонский
«__» _____ 2014 г.

Институт Кибернетики
Кафедра Оптимизации Систем Управления

Фонд оценочных средств
ТЕКУЩЕГО ОЦЕНИВАНИЯ

по ООП/ дисциплине *Алгоритмы и структуры данных*

Разработан в соответствии с рабочей программой по дисциплине «Алгоритмы и структуры данных», утвержденной на заседании кафедры ОСУ «27» июня 2014г.
направление подготовки ООП 09.03.04 Программная инженерия _

Курс 3, Семестр 5

Распределение учебного времени

Лекции 32 час.

Лабораторные занятия 32 час.

Самостоятельная работа 152 час.

Дата разработки: 25.02.15

	по работе с ними	деки, очереди			
5	Понимание принципов работы различных алгоритмов сортировки	Алгоритмы сортировки	1	Варианты заданий для выполнения лабораторных работ	1
6	Понимание принципов работы различных алгоритмов поиска подстроки в строке	Алгоритмы поиска	1	Варианты заданий для выполнения лабораторных работ	1
7	Понимание устройства структуры организации данных типа хэш-таблица, методов работы с этой структурой, разрешение коллизий	Хэш-таблицы	1	Варианты заданий для выполнения лабораторных работ	1
8	Понимание устройства структуры организации данных типа дерево, и операций по работе с этой структурой.	Деревья	1	Варианты заданий для выполнения лабораторных работ	1

2. Кодификатор

Тема (раздел)	Дидактическая единица	Поведенческие индикаторы	Уровень усвоения	Метод, форма контроля	Уровень значимости задания	Коэфф. трудности
Осенний семестр						
Основные понятия алгоритмов и структур данных	Способы представления алгоритмов	Объяснить способы представления алгоритмов	Понимание	Устный опрос	Базовый	КТ1
	Вычисление трудоемкости алгоритмов	Объяснить методику вычисления трудоемкости алгоритмов	Понимание	Устный опрос	Базовый	КТ2
Данные с динамическо й структурой	Структура списка	Объяснить из каких элементов состоит список и каким образом эти элементы связаны.	Понимание	Устный опрос	Базовый	КТ1
	Операции по работе со списком	Объяснить как выполняются операции с элементами списка: добавление, удаление, вставка элемента.	Понимание	Устный опрос	Базовый	КТ1
Основные понятия алгоритмов и структур данных	Типы данных	Дать определения различных типов данных	Знание	Устный опрос	Базовый	КТ1
	Представление в битовом виде	Объяснить алгоритмы конвертации типов в битовое представление	Понимание	Устный опрос	Базовый	КТ2
	Конвертация битового представления	Объяснить алгоритмы конвертации из битового представления в заданный тип	Понимание	Устный опрос	Базовый	КТ2

Данные с динамической структурой	Виды списков	Дать определения двунаправленного, циклического списков, специализированных видов списков: стека, дека, очереди	Знание	Устный опрос	Средний	КТ1
	Операции по работе с двунаправленными и циклическими списками.	Объяснить как выполняются операции с элементами списка: добавление, удаление, вставка элемента.	Понимание	Устный опрос	Средний	КТ2
	Операции по работе со специализированными списками.	Объяснить как выполняются операции с элементами специализированных списков: добавление, удаление, вставка элемента.	Понимание	Устный опрос	Средний	КТ2
Алгоритмы поиска	Алгоритм «наивного» поиска грубой силы	Объяснить принцип работы алгоритма «наивного» поиска (алгоритм поиск «грубой силы»)	Понимание	Устный опрос	Средний	КТ1
	Алгоритм Кнута-Морриса-Пратта	Объяснить принцип работы алгоритма Кнута-Морриса-Пратта	Понимание	Устный опрос	Средний	КТ3
	Алгоритм Бойера-Мура	Объяснить принцип работы алгоритма Бойера-Мура	Понимание	Устный опрос	Средний	КТ3
	Алгоритм Рабина-Карпа	Объяснить принцип работы алгоритма Рабина-Карпа	Понимание	Устный опрос	Средний	КТ3
Алгоритмы сортировки	Сортировки подсчетом, слиянием и распределением	Объяснить алгоритмы сортировок подсчетом, слиянием и распределением	Понимание	Устный опрос	Средний	КТ3
	Сортировки включением	Объяснить алгоритмы сортировки простым включением и сортировки Шелла	Понимание	Устный опрос	Средний	КТ3

	Сортировки выбором	Объяснить алгоритмы сортировки простым извлечением и пирамидальной сортировки	Понимание	Устный опрос	Средний	КТ3
	Сортировки обменами	Объяснить алгоритмы пузырьковой сортировки и сортировки Хоара(быстрой сортировки)	Понимание	Устный опрос	Средний	КТ3
Деревья	Основные понятия	Давать определения терминам «Дерево», «Корень дерева», «Пустое дерево», «Дочерняя вершина», «Родительская вершина», «Предки», «Уровень дерева»	Знание	Устный опрос	Системный	КТ3
	Бинарное дерево	Объяснить отличие от простого дерева, объяснить алгоритмы операций с элементами дерева: добавление, удаление элементов.	Знание	Устный опрос	Средний	КТ2
	Сбалансированные деревья	Объяснить отличие от простого и бинарного дерева, объяснить алгоритмы операций с элементами дерева: добавление, удаление элементов.	Понимание	Устный опрос	Системный	КТ3
Быстрый доступ к данным	Основные понятия	Давать определение терминам «Хэш-таблица», «Коллизия»	Знание	Устный опрос	Средний	КТ1
	Хэш-функция	Объяснить механизм добавления и доступа к элементам хэш-таблицы	Понимание	Устный опрос	Системный	КТ2
	Разрешение коллизий	Объяснить механизм разрешения коллизий с помощью метода открытой адресации с линейным опробыванием и метода цепочек	Понимание	Устный опрос	Средний	КТ3

3. Рейтинг план дисциплины

3.1. Осенний семестр

№ п/п	Наименование темы	Максимальный балл
1	Основные понятия алгоритмов и структур данных (часть 1)	3
2	Данные с динамической структурой (часть 1)	4
3	Алгоритмы поиска	5
4	Алгоритмы сортировки	8
5	Рубежный контроль №1	10
6	Деревья	10
7	Быстрый доступ к данным	5
8	Рубежный контроль №2	15
Итого		60

4. База контрольных учебных заданий

4.1. Основные понятия алгоритмов и структур данных (часть 1)

Задание:

1. Написать алгоритм решения соответствующего варианта задания на псевдокоде
2. На основе описания алгоритма вывести формулу определения числа операций в зависимости от размерности исходных данных
3. Построить графики зависимости времени выполнения от размерности исходных данных
4. Написать программы тестирования алгоритмов и построить графики зависимости времени выполнения от размерности исходных данных на основе вычислительных экспериментов
5. Провести сравнительный анализ графиков по пунктам 3-4

Варианты:

1. В массиве строк найти число одинаковых слов.
2. Дан двумерный массив $A[1..m, 1..n]$. Написать программу построения одномерного массива $B[1..m]$, элементы которого соответственно равны наименьшим средним арифметических элементов строк.

$$X[n] = \frac{(2^n - 1)}{2}$$

3. Элементы массива $X[1..N]$ вычисляются по формуле $X[n] = \frac{(2^n - 1)}{2}$, $n=1, \dots, N$. Написать программу вычисления элементов массива, не используя операцию возведения в степень.
4. Определить, сколько чисел входят в данный массив только по одному разу и более, чем по одному разу.
5. Элемент двумерного массива называется локальным минимумом, если он строго меньше всех имеющихся у него соседей. Подсчитать количество локальных минимумов заданной матрицы размером $n \times n$
6. Задана последовательность из N чисел. Найти самую длинную подпоследовательность, обладающую следующим свойством:
 $A[i] < A[i + 1] > A[i + 2] < A[i + 3] > A[i + 4] < \dots$
7. В данной последовательности целых чисел переставить члены так, чтобы положительные числа шли в порядке возрастания в начале массива.
8. В данной последовательности целых чисел найти количество различных нечетных положительных чисел.
9. В заданном массиве строк найти номера строк с наибольшим и наименьшим числом символов
10. В матрице найти число столбцов упорядоченных по возрастанию

4.2. Данные с динамической структурой (часть 1)

Варианты заданий:

1. Описать, используя структуру данных запись, каталог книг в библиотеке. Составить программу, выдающую список книг В.Пикуля, хранящихся в библиотеке.
2. Описать, используя структуру данных запись, телефонную книгу. Составить программу, выдающую список абонентов, имеющих телефонный номер, начинающийся на 331.
3. Описать, используя структуру данных запись, таблицу дат и событий русской истории. Составить программу, выдающую список событий 19 века.
4. Описать, используя структуру данных запись, школьный класс (Фамилия и инициалы, дата рождения, месяц рождения, год рождения). Составить программу, выдающую список учеников, рожденных в мае месяце.
5. Описать, используя структуру данных запись, записную книжку (Фамилия и инициалы, год рождения, дата рождения, месяц рождения). Составить программу, выдающую список друзей, кому в этом году исполняется 25 лет.
6. Описать, используя структуру данных запись, школьный класс (Фамилия и инициалы, дата рождения, месяц рождения, год рождения). Составить программу, выдающую день рождения класса (среднее арифметическое дат и месяцев).
7. Описать, используя структуру данных запись, школьную нагрузку (фамилия преподавателя, класс, часы). Составить программу, определяющую нагрузку каждого преподавателя. Определить у какого преподавателя самая большая нагрузка и кого самая низкая.
8. Описать, используя структуру данных запись, таблицу соревнований (название команды, количество набранных очков). Выбрать команду, занявшую первое место. Упорядочить список команд, в зависимости от занятого места.
9. При сдаче норм ГТО, были получены результаты забега на 100 метров и прыжков в длину, задайте нормы ГТО по этим видам, определите списки учеников, невыполнивших нормативы, количество учеников сдавших нормативы, а также списки 3 лучших.
10. Описать, используя структуру данных запись, оценки за год. Посчитать процент и качество успеваемости в классе за год, составьте списки неуспевающих и отличников.

4.3. Основные понятия алгоритмов и структур данных (часть 2)

Задание:

1. Для заданных типов данных (в соответствии с вариантом) определить набор значений, необходимый для изучения внутреннего представления.
2. Преобразовать значения в двоичный код.
3. Преобразовать двоичный код в значение.
4. Разработать и отладить программу, выдающую двоичное представление значений заданных типов. В программе реализовать процедуры PrintByte (печатает двоичное значение переменной заданного типа) и PrintVar(печатает преобразованное из двоичного значение переменной заданного типа).

Варианты:

1. Int, String, {red, yellow, green}
2. Byte, Short, 1000...2000
3. Float, Long, 'A'...'Z'
4. Char, Int, массив [1...8] Bool
5. Bool, Double, {winter, spring, summer, autumn}
6. String, Short, -256...-1
7. UInt, Char, 1...50
8. Double, Char, {a, b, c, d, e, f, g, h}
9. Bool, Char, массив [1...5] Float
10. Float, String, {day,night}

4.4. Данные с динамической структурой (часть 2).

Задание:

Взять тип данных, использованный при выполнении 4.2 (в соответствии с вариантом) и реализовать указанные структуры и операторы. При реализации учесть способ задания списка.

Варианты:

1. Реализуйте односвязный список для соответствующего типа данных и его операторы INSERT, LOCATE, RETRIEVE, DELETE, NEXT, PREVIOUS, MAKENULL, PRINTLIST. Список задан в виде массива.
2. Реализуйте двусвязный список для соответствующего типа данных и его операторы INSERT, LOCATE, RETRIEVE, DELETE, NEXT, PREVIOUS, MAKENULL, PRINTLIST. Список представляется с использованием указателей.
3. Реализуйте стек для соответствующего типа данных и его операторы MAKENULL, TOP, POP, PUSH, EMPTY. Стек задан в виде массива.
4. Реализуйте стек для соответствующего типа данных и его операторы MAKENULL, TOP, POP, PUSH, EMPTY. Стек задан в виде односвязного списка.
5. Реализуйте стек для соответствующего типа данных и его операторы MAKENULL, TOP, POP, PUSH, EMPTY. Стек представляется с использованием указателей.
6. Реализуйте очередь для соответствующего типа данных и его операторы MAKENULL, FRONT, ENQUEUE, DEQUEUE, EMPTY. Очередь задана в виде циклического однонаправленного списка.
7. Реализуйте очередь для соответствующего типа данных и его операторы MAKENULL, FRONT, ENQUEUE, DEQUEUE, EMPTY. Очередь задана в виде двусвязного списка.
8. Реализовать процедуры перемещения элементов одно-связанного списка в стек и наоборот. Список реализуется с помощью указателей, стек – массивом.
9. Реализовать процедуры перемещения элементов очереди в двух связанный список и наоборот. Список и очередь реализуется с помощью указателей.
10. Реализовать процедуры перемещения элементов дека в очередь и наоборот. Дек реализуется с помощью указателей, очередь – массивом.

4.5. Алгоритмы поиска

Задание:

- 1) Необходимо взять текст, объемом 10 страниц и сравнить по трудоемкости алгоритмы поиска подстроки в строке в соответствии с вариантом.
- 2) Составить сравнительные графики работы этих алгоритмов (время работы – размер текста)

Варианты:

- 1) Алгоритм «наивного» грубого поиска - Алгоритм Кнута-Морриса-Пратта
- 2) Алгоритм «наивного» грубого поиска - Алгоритм Бойера-Мура
- 3) Алгоритм «наивного» грубого поиска - Алгоритм Рабина-Карпа
- 4) Алгоритм Кнута-Морриса-Пратта - Алгоритм Бойера-Мура
- 5) Алгоритм Кнута-Морриса-Пратта - Алгоритм Рабина-Карпа
- 6) Алгоритм Бойера-Мура - Алгоритм Рабина-Карпа
- 7) Алгоритм Кнута-Морриса-Пратта - Алгоритм Бойера-Мура
- 8) Алгоритм «наивного» грубого поиска - Алгоритм Бойера-Мура
- 9) Алгоритм Бойера-Мура - Алгоритм Рабина-Карпа
- 10) Алгоритм «наивного» грубого поиска - Алгоритм Рабина-Карпа

4.6. Алгоритмы сортировки

Задание:

- 1) Реализовать алгоритмы поиска в соответствии с вариантом.
- 2) Провести сравнительный анализ этих методов на массивах данных из 100, 1000, 10000, 100000 элементов и построить соответствующий график.

Варианты:

- 1) Сортировка подсчетом и сортировка Шелла
- 2) Сортировка Хоара и сортировка включением
- 3) Сортировка простым извлечением и пузырьковая сортировка
- 4) Сортировка Шелла и пирамидальная сортировка
- 5) Сортировка подсчетом и сортировка простым извлечением
- 6) Сортировка Шелла и сортировка Хоара
- 7) Пузырьковая сортировка и пирамидальная сортировка
- 8) Пузырьковая сортировка и сортировка подсчетом
- 9) Сортировка Хоара и пирамидальная сортировка
- 10) Сортировка Шелла и пузырьковая сортировка

4.7. Деревья

Задание:

- 1) Придумать дерево из 15 вершин.
- 2) Написать программу в соответствии с вариантом.

Варианты:

- 1) Дан указатель P1 на корень непустого дерева. Вывести указатель P2 на последнюю вершину дерева с максимальным нечетным значением (вершины перебирать в инфиксном порядке). Если дерево не содержит вершин с нечетными значениями, то вывести null.
- 2) Дан указатель P1 на корень непустого дерева. Вывести значения всех вершин дерева в префиксном порядке (вначале выводится значение корня, затем — содержимое левого поддерева в префиксном порядке, затем — содержимое правого поддерева в префиксном порядке).
- 3) Дан указатель P1 на корень непустого дерева. Вывести значения всех вершин дерева в постфиксном порядке (вначале выводится содержимое левого поддерева в постфиксном порядке, затем — содержимое правого поддерева в постфиксном порядке, затем — значение корня).
- 4) вывести значения всех вершин с порядковыми номерами от N1 до N2.
- 5) Дан указатель P1 на корень непустого дерева. Вывести максимальное из значений его вершин и количество вершин, имеющих это максимальное значение.
- 6) Дан указатель P1 на корень непустого дерева. Вывести минимальное из значений всех его вершин и количество листьев, имеющих это минимальное значение (данное количество может быть равно 0).
- 7) Дан указатель P1 на корень непустого дерева. Вывести минимальное из значений его вершин, являющихся листьями.
- 8) Дан указатель P1 на корень дерева. Вывести максимальное из значений его внутренних вершин (то есть вершин, не являющихся листьями).
- 9) Дан указатель P1 на корень непустого дерева. Вывести указатель P2 на первую вершину дерева с минимальным значением (вершины перебирать в префиксном порядке).
- 10) Дан указатель P1 на корень непустого дерева. Вывести сумму значений всех листьев данного дерева.

4.8. Быстрый доступ к данным

Задание:

Реализовать хеш-таблицу со следующими возможностями:

1. Емкость таблицы – порядка 2000 элементов, ключи – натуральные числа;
2. Добавление, извлечение элементов, проверка на входжение;
3. Вывод содержимого структуры данных на экран с указанием индекса в массиве, хеш-значения, значения элемента;
4. Для каждой операции добавления и поиска реализовать подсчет количества коллизий (если такое требование указано явно);

Варианты:

- 1) Хеш-функция – вариант деления с остатком, метод разрешения коллизий – линейное опробывание.
- 2) Хеш-функция – вариант с умножением, метод разрешения коллизий – линейное опробывание.
- 3) Хеш-функция – вариант деления с остатком, метод разрешения коллизий – линейное опробывание.
- 4) Хеш-функция – вариант с умножением, метод разрешения коллизий – линейное опробывание.
- 5) Хеш-функция – вариант деления с остатком, метод разрешения коллизий – метод цепочек.
- 6) Хеш-функция – вариант с умножением, метод разрешения коллизий – метод цепочек.
- 7) Хеш-функция – вариант деления с остатком, метод разрешения коллизий – метод цепочек.
- 8) Хеш-функция – вариант с умножением, метод разрешения коллизий – метод цепочек.
- 9) Хеш-функция – вариант с умножением, метод разрешения коллизий – линейное опробывание.
- 10) Хеш-функция – вариант деления с остатком, метод разрешения коллизий – метод цепочек.

5. Спецификация контролирующих мероприятий

5.1. Лабораторные работы

Цель выполнения – проверка текущего уровня понимания материалов по основным темам дисциплины «Алгоритмы и структуры данных».

Содержание и структура лабораторных работ. Определяется банком заданий в зависимости от темы, в п. 4 настоящего документа.

Время выполнения лабораторных работ. Нормы времени для выполнения заданий лабораторных работ приведены в таблице 5.1.

Таблица 5.1 – Время выполнения заданий лабораторных работ

№ п/п	Наименование темы	Количество заданий	Время выполнения, академ. час
1	Основные понятия алгоритмов и структур данных	1	1
2	Данные с динамической структурой	1	1
3	Алгоритмы поиска	1	3
4	Алгоритмы сортировки	1	3
5	Деревья	1	3
6	Быстрый доступ к данным	1	3

5.2. Рубежный контроль – контрольные работы

Цель выполнения – проверка текущего уровня усвоения учащимися материалов (их понимания) по дисциплине «Алгоритмы и структуры данных». Анализ результатов контрольной работы для выявления наиболее проблемных тем и соответствующая корректировка учебной программы.

Содержание работы. Контрольная работа содержит от 1 до 3 задач, тематика которых соответствует пройденным до даты проведения рубежного контроля темам.

Структура контрольной работы. Рубежный контроль №1 содержит 3 задания для проверки понимания тем «Представление алгоритмов и их сложность», «Однонаправленные списки», «Типы данных и их представление», «Двунаправленные, циклически и специализированные виды списков».

Рубежный контроль №2 включает в себя одно комплексное задание для проверки понимания тем «Алгоритмы сортировки», «Алгоритмы поиска», «Хэш-таблицы», «Деревья».

Время и способ выполнения заданий. На выполнение заданий рубежного контроля отводится 90 минут. Работы выполняются на ПК. Среда разработки выбирается студентом. По истечению срока выполнения написанные программы сохраняются для последующей проверки преподавателем.

6. Методические материалы, определяющие процедуру контроля и критерии оценивания

6.1. Оценка выполнения лабораторных работ

Процедура контроля. Лабораторные работы выполняются учащимися согласно расписанию занятий. При сдаче лабораторной работы преподавателем оценивается верность полученного результата, а также теоретические знания по теме.

Критерии оценки. Критерии оценки выполнения лабораторных работ по темам дисциплины приведены в таблице 6.1.

Таблица 6.1 – Критерии оценки выполнения лабораторных работ

№ п/п	Наименование темы	Критерий оценки	Балл
Осенний семестр			
1	Основные понятия алгоритмов и структур данных (часть 1)	Выполнено задание №1	1
		Учащийся верно обосновал подсчет трудоемкости алгоритма	1
		Учащийся верно проиллюстрировал свой алгоритм с помощью способа представления «блок-схема».	1
		Максимальный балл по теме	3
2	Данные с динамической структурой (часть 1)	Выполнено задание №1	2
		Учащийся верно объяснил механизм работы методов удаления и добавления элементов.	
		Максимальный балл по теме	3
3	Основные понятия алгоритмов и структур данных (часть 2)	Выполнено задание №1	2
		Учащийся верно объяснил механизм конвертации типов данных в битовое представление и обратно	1
		Максимальный балл по теме	3
4	Данные с динамической структурой (часть 2)	Выполнено задание №1	3
		Учащийся верно объяснил различия между различными типами списков.	1
		Учащийся верно объяснил механизм удаления и добавления элементов в список.	1
		Максимальный балл по теме	5
5	Алгоритмы поиска	Реализован алгоритм №1	2
		Реализован алгоритм №2	2
		Правильно произведено сравнение эффективности этих алгоритмов.	1
		Учащийся верно объяснил и проиллюстрировал принцип работы алгоритмов.	2
		Максимальный балл по теме	7
6	Алгоритмы сортировки	Реализован алгоритм №1	2
		Реализован алгоритм №2	2
		Правильно произведено сравнение эффективности этих алгоритмов.	1
		Учащийся верно объяснил и проиллюстрировал принцип работы алгоритмов.	2
		Максимальный балл по теме	7

9	Деревья	Выполнено задание №1	3
		Учащийся успешно дал определения терминам, соответствующим теме «Деревья»	1
		Учащийся проиллюстрировал механизм добавления и удаления элементов из дерева.	2
		Максимальный балл по теме	6
7	Быстрый доступ к данным	Выполнено задание №1	2
		Учащийся верно объяснил принцип добавления и доступа к элементам хэш-таблицы	2
		Учащийся верно объяснил принцип использованного им метода разрешения коллизий	2
		Максимальный балл по теме	6