

Документ подписан простой электронной подписью
 Информация о владельце:
 ФИО: ЧУМАЧЕНКО ТАТЬЯНА АЛЕКСАНДРОВНА
 Должность: РЕКТОР
 Дата подписания: 31.08.2022 11:48:08
 Уникальный программный ключ:
 9c9f7aaffa4840d284abe156657b8f85432bdb16



МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ
ГУМАНИТАРНО-ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ЮУнГГПУ»)

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ
(ОЦЕНОЧНЫЕ СРЕДСТВА)

Шифр	Наименование дисциплины (модуля)
Б1.О	Алгоритмы и структуры данных

Код направления подготовки	09.03.02
Направление подготовки	Информационные системы и технологии
Наименование (я) ОПОП (направленность / профиль)	Информационные технологии в образовании
Уровень образования	бакалавр
Форма обучения	очная

Разработчики:

Должность	Учёная степень, звание	Подпись	ФИО
Старший преподаватель			Боровская Елена Владимировна

Рабочая программа рассмотрена и одобрена (обновлена) на заседании кафедры (структурного подразделения)

Кафедра	Заведующий кафедрой	Номер протокола	Дата протокола	Подпись
кафедра информатики, информационных технологий и методики обучения информатике	Рузаков Андрей Александрович	10	13.06.2019	
кафедра информатики, информационных технологий и методики обучения информатике	Рузаков Андрей Александрович	1	10.09.2020	

Раздел 1. Компетенции обучающегося, формируемые в результате освоения образовательной программы с указанием этапов их формирования

Таблица 1 - Перечень компетенций, с указанием образовательных результатов в процессе освоения дисциплины (в соответствии с РПД)

Формируемые компетенции			
Индикаторы ее достижения	Планируемые образовательные результаты по дисциплине		
	знать	уметь	владеть
ОПК-6 способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий			
ОПК.6.1 Знать методы алгоритмизации, языки и технологии программирования, пригодные для практического применения в области информационных систем и технологий	3.2 Знать основные алгоритмы сортировки и поиска, алгоритмы на строках, жадные алгоритмы и т.д.		
ОПК.6.2 Уметь применять методы алгоритмизации, языки и технологии программирования при решении профессиональных задач в области информационных систем и технологий		У.2 Уметь: программировать статические и динамические структуры со всеми их операторами; анализировать эффективность алгоритмов, моделей и структур данных, объектов и интерфейсов; программировать методы сортировки, поиска на различных структурах.	
ОПК.6.3 Иметь навыки программирования, отладки и тестирования прототипов программно-технических комплексов задач			В.2 Владеть: навыками анализа предметной области, моделирования информационной системы и применения соответствующих структур и алгоритмов при проектировании программ.
ОПК-1 способен применять естественнонаучные и инженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности			
ОПК.1.1 Знать основы математики, физики, вычислительной техники и программирования.	3.1 Знать основные структуры данных: стек, дек, список, дерево и т.д.		
ОПК.1.2 Уметь решать стандартные профессиональные задачи с применением естественнонаучных и инженерных знаний, методов математического анализа и моделирования.		У.1 умение проводить анализ корректности и временной сложности алгоритмов	
ОПК.1.3 Иметь навыки теоретического и экспериментального исследования объектов профессиональной деятельности.			В.1 Способность программировать, отлаживать и тестировать алгоритмы и структуры данных

Компетенции связаны с дисциплинами и практиками через матрицу компетенций согласно таблице 2.

Таблица 2 - Компетенции, формируемые в результате обучения

Код и наименование компетенции	
Составляющая учебного плана (дисциплины, практики, участвующие в формировании компетенции)	Вес дисциплины в формировании компетенции (100 / количество дисциплин, практик)
ОПК-6 способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий	
Исследование операций и методы оптимизации	20,00
Технологии программирования	20,00
учебная практика (ознакомительная)	20,00
производственная практика (технологическая (проектно-технологическая))	20,00
Алгоритмы и структуры данных	20,00
ОПК-1 способен применять естественнонаучные и общетехнические знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности	
Исследование операций и методы оптимизации	7,69
Математическая логика	7,69
Архитектура информационных систем	7,69
Вычислительная математика	7,69
Физика	7,69
Информационные технологии	7,69
Моделирование систем	7,69
учебная практика (ознакомительная)	7,69
производственная практика (технологическая (проектно-технологическая))	7,69
Алгоритмы дискретной математики	7,69
Алгоритмы и структуры данных	7,69
Теория информации, данные, знания	7,69
Математика	7,69

Таблица 3 - Этапы формирования компетенций в процессе освоения ОПОП

Код компетенции	Этап базовой подготовки	Этап расширения и углубления подготовки	Этап профессионально-практической подготовки
ОПК-6	Исследование операций и методы оптимизации, Технологии программирования, учебная практика (ознакомительная), производственная практика (технологическая (проектно-технологическая)), Алгоритмы и структуры данных		учебная практика (ознакомительная), производственная практика (технологическая (проектно-технологическая))

ОПК-1	Исследование операций и методы оптимизации, Математическая логика, Архитектура информационных систем, Вычислительная математика, Физика, Информационные технологии, Моделирование систем, учебная практика (ознакомительная), производственная практика (технологическая (проектно-технологическая)), Алгоритмы дискретной математики, Алгоритмы и структуры данных, Теория информации, данные, знания, Математика		учебная практика (ознакомительная), производственная практика (технологическая (проектно-технологическая))
-------	---	--	---

Раздел 2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Таблица 4 - Показатели оценивания компетенций на различных этапах их формирования в процессе освоения учебной дисциплины (в соответствии с РПД)

№	Раздел		
Формируемые компетенции			
	<table> <tr> <th>Показатели сформированности (в терминах «знать», «уметь», «владеть»)</th><th>Виды оценочных средств</th></tr> </table>	Показатели сформированности (в терминах «знать», «уметь», «владеть»)	Виды оценочных средств
Показатели сформированности (в терминах «знать», «уметь», «владеть»)	Виды оценочных средств		
1	Алгоритмы		
ОПК-1 ОПК-6			
Знать знать основные алгоритмыЖ сортировки и поиска, алгоритмы на строках, жадные алгоритмы и т.д.	Задача		
<p>Уметь умение проводить анализ корректности и временной сложности алгоритмов</p> <p>Уметь уметь:программировать статические и динамические структуры со всеми их операторами; анализировать эффективность алгоритмов, моделей и структур данных, объектов и интерфейсов; программировать методы сортировки, поиска на различных структурах.</p>	Задача		
<p>Владеть способность программировать, отлаживать и тестировать алгоритмы и структуры данных</p> <p>Владеть владеть:навыками анализа предметной области,моделирования информационной системы и применения соответствующих структур и алгоритмов при проектировании программ.</p>	Задача		
2	Структурв данных		
ОПК-1 ОПК-6			
Знать знать основные структуры данных: стек, дек, список, дерево и т.д. Знать знать основные алгоритмыЖ сортировки и поиска, алгоритмы на строках, жадные алгоритмы и т.д.	Задача		
<p>Уметь умение проводить анализ корректности и временной сложности алгоритмов</p> <p>Уметь уметь:программировать статические и динамические структуры со всеми их операторами; анализировать эффективность алгоритмов, моделей и структур данных, объектов и интерфейсов; программировать методы сортировки, поиска на различных структурах.</p>	Задача		
<p>Владеть способность программировать, отлаживать и тестировать алгоритмы и структуры данных</p> <p>Владеть владеть:навыками анализа предметной области,моделирования информационной системы и применения соответствующих структур и алгоритмов при проектировании программ.</p>	Задача		

Таблица 5 - Описание уровней и критериев оценивания компетенций, описание шкал оценивания

Код	Содержание компетенции			
Уровни освоения компетенции	Содержательное описание уровня	Основные признаки выделения уровня (критерии оценки сформированности)	Пятибалльная шкала (академическая оценка)	% освоения (рейтинговая оценка)
ОПК-6	ОПК-6 способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий			
ОПК-1	ОПК-1 способен применять естественнонаучные и общетехнические знания, методы математического анализа и моделирования, теоретического и экспериментал...			

Раздел 3. Типовые контрольные задания и (или) иные материалы, необходимые для оценки планируемых результатов обучения по дисциплине (модулю)

1. Оценочные средства для текущего контроля

Раздел: Алгоритмы

Задания для оценки знаний

1. Задача:

1. Представьте, что вы являетесь разработчиком стратегии реального времени. Допустим, игрок может создавать на карте здания-приемники (амбары, лесопилки, кузницы и т. д.). В них поступают ресурсы с прилегающих территорий (зерно, лес, руда и т. д.). При имеющемся расположении приемников необходимо поделить всю карту на непересекающиеся области таким образом, чтобы в каждой области был ровно один приемник и он был ближайшим для всех ресурсов в этой области. Можно считать, что препятствий на карте нет.
2. У преподавателя есть список заданий, каждое из которых имеет тип (теория/практика/блиц/прочие), тематику и уровень сложности. Постройте алгоритм для генерации списка билетов, такой, чтобы одновременно выполнялись условия:
 - 1) в зависимости от внешних условий состав билетов меняется;
 - 2) билет содержит одинаковое количество вопросов из разных тем, и одинаковое соотношение заданий разных типов (например, 2 теории и одна практика);
 - 3) уровень сложности всех билетов приблизительно одинаковый.
3. Преподавателю надо печатать билеты и при этом потратить как можно меньше бумаги. Упорядочите билеты так, чтобы они занимали наименьшее число листов А4, между ними было место для разреза, у листа были поля сверху и снизу и ни один билет не оказался «разорван» между листами.
4. Предположим, что вы устроились погромистом в некую успешную компанию. Вам поручили автоматизировать составление сменного графика работы сотрудников. В зависимости от должности, каждый сотрудник должен отработать определенное число дней в месяц (например, 22 из 30) по сменному графику. Каждый из сменных работников отправляет желаемое расписание на месяц, а ваша задача — составить график так, чтобы максимально удовлетворить все заинтересованные стороны (сделать меньше всего изменений в графиках работников и избежать случая, когда в один день на работе 10 сотрудников, а в другой — ни одного). Помните, что число дней в месяце может быть разным.

Задания для оценки умений

1. Задача:

1. Представьте, что вы являетесь разработчиком стратегии реального времени. Допустим, игрок может создавать на карте здания-приемники (амбары, лесопилки, кузницы и т. д.). В них поступают ресурсы с прилегающих территорий (зерно, лес, руда и т. д.). При имеющемся расположении приемников необходимо поделить всю карту на непересекающиеся области таким образом, чтобы в каждой области был ровно один приемник и он был ближайшим для всех ресурсов в этой области. Можно считать, что препятствий на карте нет.
2. У преподавателя есть список заданий, каждое из которых имеет тип (теория/практика/блиц/прочие), тематику и уровень сложности. Постройте алгоритм для генерации списка билетов, такой, чтобы одновременно выполнялись условия:
 - 1) в зависимости от внешних условий состав билетов меняется;
 - 2) билет содержит одинаковое количество вопросов из разных тем, и одинаковое соотношение заданий разных типов (например, 2 теории и одна практика);
 - 3) уровень сложности всех билетов приблизительно одинаковый.
3. Преподавателю надо печатать билеты и при этом потратить как можно меньше бумаги. Упорядочите билеты так, чтобы они занимали наименьшее число листов А4, между ними было место для разреза, у листа были поля сверху и снизу и ни один билет не оказался «разорван» между листами.
4. Предположим, что вы устроились погромистом в некую успешную компанию. Вам поручили автоматизировать составление сменного графика работы сотрудников. В зависимости от должности, каждый сотрудник должен отработать определенное число дней в месяц (например, 22 из 30) по сменному графику. Каждый из сменных работников отправляет желаемое расписание на месяц, а ваша задача — составить график так, чтобы максимально удовлетворить все заинтересованные стороны (сделать меньше всего изменений в графиках работников и избежать случая, когда в один день на работе 10 сотрудников, а в другой — ни одного). Помните, что число дней в месяце может быть разным.

Задания для оценки владений

1. Задача:

1. Представьте, что вы являетесь разработчиком стратегии реального времени. Допустим, игрок может создавать на карте здания-приемники (амбары, лесопилки, кузницы и т. д.). В них поступают ресурсы с прилегающих территорий (зерно, лес, руда и т. д.). При имеющемся расположении приемников необходимо поделить всю карту на непересекающиеся области таким образом, чтобы в каждой области был ровно один приемник и он был ближайшим для всех ресурсов в этой области. Можно считать, что препятствий на карте нет.

2. У преподавателя есть список заданий, каждое из которых имеет тип (теория/практика/блиц/прочие), тематику и уровень сложности. Постройте алгоритм для генерации списка билетов, такой, чтобы одновременно выполнялись условия:

1) в зависимости от внешних условий состав билетов меняется;

2) билет содержит одинаковое количество вопросов из разных тем, и одинаковое соотношение заданий разных типов (например, 2 теории и одна практика);

3) уровень сложности всех билетов приблизительно одинаковый.

3. Преподавателю надо печатать билеты и при этом потратить как можно меньше бумаги. Упорядочите билеты так, чтобы они занимали наименьшее число листов А4, между ними было место для разреза, у листа были поля сверху и снизу и ни один билет не оказался «разорван» между листами.

4. Предположим, что вы устроились погромистом в некую успешную компанию. Вам поручили автоматизировать составление сменного графика работы сотрудников. В зависимости от должности, каждый сотрудник должен отработать определенное число дней в месяц (например, 22 из 30) по сменному графику. Каждый из сменных работников отправляет желаемое расписание на месяц, а ваша задача — составить график так, чтобы максимально удовлетворить все заинтересованные стороны (сделать меньше всего изменений в графиках работников и избежать случая, когда в один день на работе 10 сотрудников, а в другой — ни одного). Помните, что число дней в месяце может быть разным.

Раздел: Структур данных

Задания для оценки знаний

1. Задача:

На междугородной телефонной станции картотека абонентов, содержит сведения о телефонах и их владельцах.

Программа должна обеспечивать:

- ☐ начальное формирование картотеки в виде линейного списка;
- ☐ вывод всей картотеки в консоль;
- ☐ ввод номера телефона и вывод времени разговора;
- ☐ ввод данных абонента и вывод извещения на оплату телефонного разговора.

Анкета для опроса населения содержит две группы вопросов.

Первая группа содержит сведения о респонденте:

- ☐ возраст;
- ☐ пол;
- ☐ образование (начальное, среднее, высшее).

Вторая группа содержит собственно вопрос анкеты, ответ на который может быть ДА или НЕТ.

Программа должна обеспечивать:

- ☐ начальный ввод анкет и формирование из них линейного списка;
- ☐ ответы на следующие вопросы на основе анализа анкет:
 - а). сколько мужчин старше 40 лет, имеющих высшее образование, ответили ДА на вопрос анкеты;
 - б). сколько женщин моложе 30 лет, имеющих среднее образование, ответили НЕТ на вопрос анкеты;
 - в). сколько мужчин моложе 25 лет, имеющих начальное образование, ответили ДА на вопрос анкеты;
 - г). вывод всех анкет и ответов на вопросы в консоль.

Разработать консольное приложение, которое с помощью абстрактных структур данных СТЭК и ОЧЕРЕДЬ проверяет введенную с клавиатуры строку текста на палиндром. Приложение должно:

1. делать запрос на ввод строки текста;
2. выводить в консоль:

- ☐ в случае палиндрома – сообщение об этом;
 - ☐ в противном случае – сообщение об этом и символ, для которого не оказалось равного ему симметричного;
3. делать запрос на выход из приложения

Используя абстрактную структуру данных БИНАРНОЕ ДЕРЕВО, разработать калькулятор, вычисляющий арифметические выражения из 4-х основных действий и скобок, записанные в постфиксной форме (postfix notation), используя подходящий порядок обхода и следующие правила:

1. дерево состоит только из узлов, у которых ровно 2 ребёнка и листьев;
2. листьям дерева соответствуют операнды – числа;
3. остальным узлам соответствуют бинарные операции – действия над 2-мя числами.

T1 и T2 - два очень больших бинарных дерева, причем T1 значительно больше T2. Создайте алгоритм, проверяющий, является ли T2 поддеревом T1. Дерево T2 считается поддеревом T1, если существует такой узел n в T1, что поддерево, «растущее» из n, идентично дереву T2. (Иначе говоря, если вырезать дерево в узле n, оно будет идентично T2.)

Как известно, слишком высокая стопка тарелок может развалиться. Следовательно, в реальной жизни, когда высота стопки превысила бы некоторое пороговое значение, мы начали бы складывать тарелки в новую стопку. Реализуйте структуру данных SetOfStacks, имитирующую реальную ситуацию. Структура SetOfStack должна состоять из нескольких стеков, новый стек создается, как только предыдущий достигнет порогового значения. Методы SetOfStacks.push () и SetOfStacks.pop () должны вести себя так же, как при работе с одним стеком (то есть метод pop () должен возвращать те же значения, которые бы он возвращал при использовании одного большого стека).

Задания для оценки умений

1. Задача:

На междугородной телефонной станции картотека абонентов, содержит сведения о телефонах и их владельцах.

Программа должна обеспечивать:

- ☐ начальное формирование картотеки в виде линейного списка;
- ☐ вывод всей картотеки в консоль;
- ☐ ввод номера телефона и вывод времени разговора;
- ☐ ввод данных абонента и вывод извещения на оплату телефонного разговора.

Анкета для опроса населения содержит две группы вопросов.

Первая группа содержит сведения о респонденте:

- ☐ возраст;
- ☐ пол;
- ☐ образование (начальное, среднее, высшее).

Вторая группа содержит собственно вопрос анкеты, ответ на который может быть ДА или НЕТ.

Программа должна обеспечивать:

- ☐ начальный ввод анкет и формирование из них линейного списка;
- ☐ ответы на следующие вопросы на основе анализа анкет:
 - а). сколько мужчин старше 40 лет, имеющих высшее образование, ответили ДА на вопрос анкеты;
 - б). сколько женщин моложе 30 лет, имеющих среднее образование, ответили НЕТ на вопрос анкеты;
 - в). сколько мужчин моложе 25 лет, имеющих начальное образование, ответили ДА на вопрос анкеты;
 - г). вывод всех анкет и ответов на вопросы в консоль.

Разработать консольное приложение, которое с помощью абстрактных структур данных СТЭК и ОЧЕРЕДЬ проверяет введенную с клавиатуры строку текста на палиндром. Приложение должно:

1. делать запрос на ввод строки текста;
2. выводить в консоль:

- ☐ в случае палиндрома – сообщение об этом;
 - ☐ в противном случае – сообщение об этом и символ, для которого не оказалось равного ему симметричного;
3. делать запрос на выход из приложения

Используя абстрактную структуру данных БИНАРНОЕ ДЕРЕВО, разработать калькулятор, вычисляющий арифметические выражения из 4-х основных действий и скобок, записанные в постфиксной форме (postfix notation), используя подходящий порядок обхода и следующие правила:

1. дерево состоит только из узлов, у которых ровно 2 ребёнка и листьев;
2. листьям дерева соответствуют операнды – числа;
3. остальным узлам соответствуют бинарные операции – действия над 2-мя числами.

T1 и T2 - два очень больших бинарных дерева, причем T1 значительно больше T2. Создайте алгоритм, проверяющий, является ли T2 поддеревом T1. Дерево T2 считается поддеревом T1, если существует такой узел n в T1, что поддерево, «растущее» из n, идентично дереву T2. (Иначе говоря, если вырезать дерево в узле n, оно будет идентично T2.)

Как известно, слишком высокая стопка тарелок может развалиться. Следовательно, в реальной жизни, когда высота стопки превысила бы некоторое пороговое значение, мы начали бы складывать тарелки в новую стопку. Реализуйте структуру данных SetOfStacks, имитирующую реальную ситуацию. Структура SetOfStack должна состоять из нескольких стеков, новый стек создается, как только предыдущий достигнет порогового значения. Методы SetOfStacks.push () и SetOfStacks.pop () должны вести себя так же, как при работе с одним стеком (то есть метод pop () должен возвращать те же значения, которые бы он возвращал при использовании одного большого стека).

Задания для оценки владений

1. Задача:

На междугородной телефонной станции картотека абонентов, содержит сведения о телефонах и их владельцах.

Программа должна обеспечивать:

- ☐ начальное формирование картотеки в виде линейного списка;
- ☐ вывод всей картотеки в консоль;
- ☐ ввод номера телефона и вывод времени разговора;
- ☐ ввод данных абонента и вывод извещения на оплату телефонного разговора.

Анкета для опроса населения содержит две группы вопросов.

Первая группа содержит сведения о респонденте:

- ☐ возраст;
- ☐ пол;
- ☐ образование (начальное, среднее, высшее).

Вторая группа содержит собственно вопрос анкеты, ответ на который может быть ДА или НЕТ.

Программа должна обеспечивать:

- ☐ начальный ввод анкет и формирование из них линейного списка;
- ☐ ответы на следующие вопросы на основе анализа анкет:
 - а). сколько мужчин старше 40 лет, имеющих высшее образование, ответили ДА на вопрос анкеты;
 - б). сколько женщин моложе 30 лет, имеющих среднее образование, ответили НЕТ на вопрос анкеты;
 - в). сколько мужчин моложе 25 лет, имеющих начальное образование, ответили ДА на вопрос анкеты;
 - г). вывод всех анкет и ответов на вопросы в консоль.

Разработать консольное приложение, которое с помощью абстрактных структур данных СТЭК и ОЧЕРЕДЬ проверяет введенную с клавиатуры строку текста на палиндром. Приложение должно:

1. делать запрос на ввод строки текста;
2. выводить в консоль:

- ☐ в случае палиндрома – сообщение об этом;
 - ☐ в противном случае – сообщение об этом и символ, для которого не оказалось равного ему симметричного;
3. делать запрос на выход из приложения

Используя абстрактную структуру данных БИНАРНОЕ ДЕРЕВО, разработать калькулятор, вычисляющий арифметические выражения из 4-х основных действий и скобок, записанные в постфиксной форме (postfix notation), используя подходящий порядок обхода и следующие правила:

1. дерево состоит только из узлов, у которых ровно 2 ребёнка и листьев;
2. листьям дерева соответствуют операнды – числа;
3. остальным узлам соответствуют бинарные операции – действия над 2-мя числами.

T1 и T2 - два очень больших бинарных дерева, причем T1 значительно больше T2. Создайте алгоритм, проверяющий, является ли T2 поддеревом T1. Дерево T2 считается поддеревом T1, если существует такой узел n в T1, что поддерево, «растущее» из n, идентично дереву T2. (Иначе говоря, если вырезать дерево в узле n, оно будет идентично T2.)

Как известно, слишком высокая стопка тарелок может развалиться. Следовательно, в реальной жизни, когда высота стопки превысила бы некоторое пороговое значение, мы начали бы складывать тарелки в новую стопку. Реализуйте структуру данных SetOfStacks, имитирующую реальную ситуацию. Структура SetOfStack должна состоять из нескольких стеков, новый стек создается, как только предыдущий достигнет порогового значения. Методы SetOfStacks.push () и SetOfStacks.pop () должны вести себя так же, как при работе с одним стеком (то есть метод pop () должен возвращать те же значения, которые бы он возвращал при использовании одного большого стека).

2. Оценочные средства для промежуточной аттестации

1. Экзамен

Вопросы к экзамену:

1. Алгоритм поиска строки в тексте Кнута, Морриса и Пратта.
2. Деревья. Представление деревьев. Понятия узла, уровня, степени узла.
3. Алгоритм поиска строки в тексте Бойера и Мура.
4. Линейные списки. Упорядоченный линейный список.
5. Цикл Дейкстры. Примеры использования цикла Дейкстры.
6. Бинарные Б-деревья. Симметричные ББ-деревья.
7. Простые методы сортировки. Простая сортировка вставками. Сортировка бинарными
8. вставками.
9. Рекурсивные типы данных. Указатели. Линейные списки.
10. Простые методы сортировки. Простая сортировка выбором.
11. Простые методы сортировки. Простая сортировка обменами (Пузырьковая). Шейкер-
12. сортировка.
13. Бинарные деревья. Основные операции с бинарными деревьями.
14. Простой поиск строки в тексте.
15. Алгоритмы с возвратом.
16. Поиск в таблице.
17. Вставка элемента в Б-дерево. Удаление элемента из Б-дерева. Степень дерева, длина
18. внутреннего и внешнего пути дерева. Идеально сбалансированные деревья.
19. Степень дерева, длина внутреннего и внешнего пути дерева.
20. Идеально сбалансированные деревья.
21. Задача поиска. Линейный поиск. Бинарный поиск.
22. Эффективные методы сортировки. Быстрая сортировка: рекурсивная и итерационная.
23. Динамические структуры данных.
24. Эффективные методы сортировки. Сортировка Пирамида.
25. Хэширование. Коллизии. Хэш-функции.
26. Определения и свойства алгоритма
27. Представление упорядоченных ориентированных деревьев. Представление бинарных деревьев.

28. Определение В-дерева. Алгоритмы поиска в В-дереве.
29. Алгоритм вставки в В-дерево. Алгоритм удаления из В-дерева.
30. Алгоритмы обхода графов: поиск в глубину и поиск в ширину.
31. Алгоритмы поиска кратчайших путей в графе: алгоритм Флойда
32. Алгоритмы поиска кратчайших путей в графе: алгоритм Дейкстры.
33. Построение кратчайших остовов графа: алгоритм Краскала.
34. Сортировка включениями с убывающим приращением (сортировка Шелла).
35. Сортировка с помощью дерева (сортировка кучей).
36. Пирамидальная сортировка.
37. Сортировка с разделением (быстрая сортировка).
38. Сравнение методов сортировки.
39. Базовые циклические алгоритмы: табулирование функций; организация счетчика.
40. Базовые циклические алгоритмы: накопление суммы или произведения.
41. Базовые циклические алгоритмы: поиск минимального или максимального члена последовательности, поиск минимального или максимального элемента двумерной матрицы.
42. Базовые циклические алгоритмы: сортировка элементов одномерного массива.
43. Алгоритмические стратегии: методы «грубой силы» (перебор всех вариантов); жадные алгоритмы (локально оптимальные).
44. Алгоритмические стратегии: алгоритмы типа «разделяй и властвуй» (декомпозиции); эвристические алгоритмы.

Практические задания:

1. Дан граф. Из вершины 1 выполняется поиск кратчайших расстояний при помощи алгоритма Дейкстры. Опишите первую и вторую итерацию данного алгоритма.
2. Составить алгоритм выстраивания массива целых чисел в порядке возрастания методом сортировки выбором (определяя на каждом этапе минимальный элемент массива и переставляя его в первый раз в начало массива, а потом – на следующее слева место после поставленных ранее минимальных элементов).
3. Пассажирский лифт не может поднять больше W кг. В лифт пытаются влезть n человек, причем для каждого из них известен его вес: W_1, W_2, \dots, W_n . Составить алгоритм, определяющий какое максимальное число людей смогут уехать на этом лифте за один раз.
4. Составить алгоритм поиска подстроки в строке символов, в которой всего N символов. Использовать функцию вырезания подстроки из строки символов.
5. Составить алгоритм выбора K -того по величине значения из числового массива. K -тое по величине значение отсчитывается от наибольшего в массиве в сторону уменьшения значений.
6. Составить алгоритм сортировки методом «пузырька».
7. Составить рекурсивный алгоритм вычисления заданного по номеру числа Фибоначчи. Числами Фибоначчи называют последовательность натуральных чисел, первое и второе из которых равны, соответственно, 0 и 1, а каждое следующее получается сложением двух предыдущих. Т.е. 3 число $1=0+1$, 4 число $2=1+1$, 5 число $3=1+2$ и т.д.
8. Даны n заявок на проведение занятий в некоторой аудитории. В каждой заявке указаны начало и конец занятия (s_i и f_i для i -й заявки). Заявки с номерами i и j совместны, если интервалы $[s_i, f_i]$ и $[s_j, f_j]$ не пересекаются (т.е. $f_i \leq s_j$ или $f_j \leq s_i$). Задача о выборе заявок состоит в том, чтобы набрать максимальное количество совместных друг с другом заявок (чтобы аудитория была максимально загружена занятиями). Составить «жадный» алгоритм, решающий данную задачу.

Раздел 4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

1. Для текущего контроля используются следующие оценочные средства:

1. Задача

Задачи позволяют оценивать и диагностировать знание фактического материала (базовые понятия, алгоритмы, факты) и умение правильно использовать специальные термины и понятия, узнавание объектов изучения в рамках определенного раздела дисциплины;

умения синтезировать, анализировать, обобщать фактический и теоретический материал с формулированием конкретных выводов, установлением причинно-следственных связей.

Алгоритм решения задач:

1. Внимательно прочитайте условие задания и уясните основной вопрос, представьте процессы и явления, описанные в условии.
2. Повторно прочтите условие для того, чтобы чётко представить основной вопрос, проблему, цель решения, заданные величины, опираясь на которые можно вести поиск решения.
3. Произведите краткую запись условия задания.
4. Если необходимо, составьте таблицу, схему, рисунок или чертёж.
5. Установите связь между искомыми величинами и данными; определите метод решения задания, составьте план решения.
6. Выполните план решения, обосновывая каждое действие.
7. Проверьте правильность решения задания.
8. Произведите оценку реальности полученного решения.
9. Запишите ответ.

2. Описание процедуры промежуточной аттестации

Оценка за зачет/экзамен может быть выставлена по результатам текущего рейтинга. Текущий рейтинг – это результаты выполнения практических работ в ходе обучения, контрольных работ, выполнения заданий к лекциям (при наличии) и др. видов заданий.

Результаты текущего рейтинга доводятся до студентов до начала экзаменационной сессии.

Экзамен преследует цель оценить работу обучающегося за определенный курс: полученные теоретические знания, их прочность, развитие логического и творческого мышления, приобретение навыков самостоятельной работы, умения анализировать и синтезировать полученные знания и применять их для решения практических задач.

Экзамен проводится в устной или письменной форме по билетам, утвержденным заведующим кафедрой (или в форме компьютерного тестирования). Экзаменационный билет включает в себя два вопроса и задачи. Формулировка вопросов совпадает с формулировкой перечня вопросов, доведенного до сведения обучающихся не позднее чем за один месяц до экзаменационной сессии.

В процессе подготовки к экзамену организована предэкзаменационная консультация для всех учебных групп.

При любой форме проведения экзаменов по билетам экзаменатору предоставляется право задавать студентам дополнительные вопросы, задачи и примеры по программе данной дисциплины. Дополнительные вопросы также, как и основные вопросы билета, требуют развернутого ответа.