

Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Комсомольский-на-Амуре государственный университет»

Кафедра «Математическое обеспечение и применение ЭВМ»

УТВЕРЖДАЮ

Первый проректор

И.В. Макурин

« 22 » 12 2017 г.



РАБОЧАЯ ПРОГРАММА

дисциплины «Структуры данных и алгоритмы»

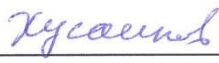
основной профессиональной образовательной программы
подготовки бакалавров

по направлению 09.03.01 – «Информатика и вычислительная техника»
профиль «Программное обеспечение средств вычислительной техники и
автоматизированных систем»

Форма обучения	очная
Технология обучения	традиционная

Комсомольск-на-Амуре 2017

Автор рабочей программы
Профессор, доктор физико-
математических наук, профессор

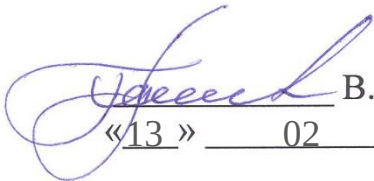
 А.А. Хусаинов
« 12 » 02 2015 г.

СОГЛАСОВАНО

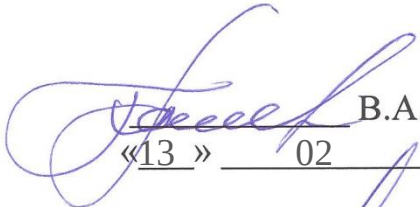
Директор библиотеки

 И.А. Романовская
« 15 » 02 2015 г.

Заведующий кафедрой «Матема-
тическое обеспечение и примене-
ние ЭВМ», кандидат технических
наук, профессор

 В.А. Тихомиров
« 13 » 02 2015 г.

Заведующий выпускающей кафед-
рой «Математическое обеспечение
и применение ЭВМ», кандидат
технических наук, профессор

 В.А. Тихомиров
« 13 » 02 2015 г.

Декан факультета компьютерных
технологий, кандидат физико- ма-
тематических наук, доцент

 Я.Ю. Григорьев
« 18 » 02 2015 г.

Начальник учебно-методического
управления

 Е.Е. Поздеева
« 20 » 02 2015 г.

Введение

Рабочая программа дисциплины «Структуры данных и алгоритмы» составлена в соответствии с требованиями федерального государственного образовательного стандарта высшего образования, утвержденного приказом Министерства образования и науки Российской Федерации от 12.01.2016 № 5, и основной профессиональной образовательной программы подготовки бакалавров по направлению 09.03.01 «Информатика и вычислительная техника».

Данная рабочая программа подготовлена для студентов набора 2016 года и далее.

1 Аннотация дисциплины

Наименование дисциплины	Структуры данных и алгоритмы							
Цель дисциплины	Освоение студентами знаний об основных структурах данных и алгоритмах, формирования практических навыков разработки программного обеспечения (ПО) с использованием типовых структур данных и стандартных алгоритмов для решения конкретных прикладных задач.							
Задачи дисциплины	1. Получение практических навыков разработки ПО с использованием линейных и нелинейных структур данных. 2. Получение практических навыков выбора и программной реализации алгоритмов обработки данных при решении практических задач. 3. Получение практических навыков программирования алгоритмов перебора. 4. Получение практических навыков программирования алгоритмов сортировки данных.							
Основные разделы дисциплины	Методы разработки алгоритмов. Структуры данных. Методы перебора. Методы сортировки.							
Общая трудоемкость дисциплины	9 з.е. / 324 академических часа							
	Семестр	Аудиторная нагрузка, ч				СРС, ч	Промежуточная аттестация, ч	Всего за семестр, ч
		Лекции	Пр. занятия	Лаб. работы	Курсовое проектирование			
		6	32	0	48	0	90	180
	7	16	0	32	0	60	36	144

Трудоемкость дисциплины представлена в календарном учебном графике, когда шестой семестр составляет 16 недель и когда шестой семестр составляет 16 недель

2 Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами образовательной программы

Дисциплина «Структуры данных и алгоритмы» нацелена на формирование компетенций, знаний, умений и навыков, указанных в таблице 1.

Таблица 1 – Компетенции, знания, умения, навыки

Наименование и шифр компетенции, в формировании которой принимает участие дисциплина	Перечень формируемых знаний, умений, навыков, предусмотренных образовательной программой		
	Перечень знаний (с указанием шифра)	Перечень умений (с указанием шифра)	Перечень навыков (с указанием шифра)
ПК-3 способностью обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности	32(ПК-3-6) Теорию организации и структурирования данных на ЭВМ	У2(ПК-3-6) Организовывать данные в разрабатываемом программном обеспечении (ПО) с использованием типовых структур	Н3(ПК-3-6) Навыками организации данных в разрабатываемом ПО и использования типовых структур данных
	31(ПК-3-7) Типовые алгоритмы обработки данных на ЭВМ	У1(ПК-3-7) Подбирать алгоритмы обработки данных для решения практических задач	Н1(ПК-3-7) Использования стандартных алгоритмов при организации и обработке данных на ЭВМ

3 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина (модуль) «Структуры данных и алгоритмы» студентами набора 2018 года и далее изучается на третьем и четвертом курсах, в шестом и седьмом семестрах (весеннем и осеннем).

Дисциплина является вариативной дисциплиной, входит в состав блока Б1 «Дисциплины (модули)» и относится к вариативной части.

Для освоения дисциплины необходимы знания, умения, навыки, сформированные на предыдущих этапах освоения компетенции ПК -3 «Способностью обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности», в процессе изучения дисциплин «Современные программные средства», «Проектирование программных средств» и «Тестирование программного обеспечения».

4 Объем дисциплины (модуля) в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся

Общая трудоемкость (объем) дисциплины составляет 9 зачетных единиц, 324 академических часов.

Распределение объема дисциплины (модуля) по видам учебных занятий представлено в таблице 2.

Таблица 2 – Объем дисциплины (модуля) по видам учебных занятий

Объем дисциплины	Всего академических часов	
	Очная форма обучения	
	16/16 недель	
Общая трудоемкость дисциплины	324	
Контактная аудиторная работа обучающихся с преподавателем (по видам учебных занятий), всего	128	
В том числе:		
занятия лекционного типа (лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации педагогическими работниками)	48	
занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия)	80	
Самостоятельная работа обучающихся и контактная работа , включающая групповые консультации, индивидуальную работу обучающихся с преподавателями (в том числе индивидуальные консультации); взаимодействие в электронной информационно-образовательной среде вуза	150	
Промежуточная аттестация обучающихся	36	

В соответствии с учебным планом в рамках дисциплины «Структуры данных и алгоритмы» в шестом семестре студент выполняет курсовую работу (КР), а в седьмом семестре – расчетно-графическую работу (РГР) .

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

Таблица 3 – Структура и содержание дисциплины (модуля) для очного обучения

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)		Форма проведения	Планируемые (контролируемые) результаты освоения	
		Для графика 16 недель			Компетенции	Знания, умения, навыки
Шестой семестр						
Раздел 1 Методы разработки алгоритмов						
Тема 1 <i>Рекурсивные алгоритмы и программы</i> Построение алгоритма и программы. Этапы построения алгоритма. Жизненный цикл ПО. Структурное программирование сверху вниз. Метод декомпозиции и рекурсивные подпрограммы. Задача о Ханойских башнях. Построение алгоритма с помощью рекуррентных соотношений. Задача о минимальном числе монет. Замена итерационного цикла вызовом рекурсивной функции. Генерация перестановок. Косвенная рекурсия. Приближение кривых Пеано с помощью кривых Гильберта. Синтаксический анализ и вычисление значения арифметического выражения методом рекурсивного спуска.	Лекция	8		Традиционная	ПК-3	32(ПК-3-6)
Тема 2 <i>Методы обхода вершин графа</i> Простой неориентированный граф, матрица смежности. Структура данных массив. Рекурсивный алгоритм раскраски вершин графа. Метод динамического программирования. Алгоритм Дейкстры поиска кратчайшего пути в графе. Алгоритм выделения компоненты связно-	Лекция	8		С использованием активных методов обучения	ПК-3	32(ПК-3-6)

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)		Форма проведения	Планируемые (контролируемые) результаты освоения	
		Для графика 16 недель			Компетенции	Знания, умения, навыки
сти графа. Жадный метод решения задачи о размене денег. Алгоритм Прима поиска остовного леса минимального веса. Алгоритм Крускала. Алгоритмы обхода вершин графа в глубину и в ширину.						
Тема <i>Разработка рекурсивных подпрограмм</i> Программная реализация рекурсивных алгоритмов.	Лабораторная работа	12		С использованием активных методов обучения	ПК-3	32(ПК-3-6) У2(ПК-3-6) Н3(ПК-3-6)
Тема <i>Алгоритмы на графах</i> Программная реализация алгоритмов на графах: алгоритма обхода вершин графа, алгоритма нахождения пути и направленного пути в графе.	Лабораторная работа	12		С использованием активных методов обучения	ПК-3	32(ПК-3-6) У2(ПК-3-6) Н3(ПК-3-6)
	Самостоятельная работа обучающихся	8		Подготовка к лабораторным занятиям	ПК-3	32(ПК-3-6) У2(ПК-3-6)
	Самостоятельная работа обучающихся	8		Изучение теоретических разделов дисциплины	ПК-3	36(ПК-2-7)
	Самостоятельная работа обучающихся	8		Выполнение, оформление и подготовка к защите лабораторных работ 1,2	ПК-3	32(ПК-3-6) У2(ПК-3-6) Н3(ПК-3-6)

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)		Форма проведения	Планируемые (контролируемые) результаты освоения	
		Для графика 16 недель			Компетенции	Знания, умения, навыки
	Самостоятельная работа обучающихся	14		Выполнение, оформление и подготовка к защите КР	ПК-3	32(ПК-3-6) У2(ПК-3-6) Н3(ПК-3-6)
	Текущий контроль			Тест	ПК-3	32(ПК-3-6)
	Текущий контроль			Защита лабораторной работы 1	ПК-3	32(ПК-3-6) У2(ПК-3-6) Н3(ПК-3-6)
ИТОГО по разделу 1	Лекции	16		-	-	-
	Лабораторные работы	24		-	-	-
	Самостоятельная работа обучающихся	45		-	-	-
Всего часов по разделу 1		90		-	-	-
Раздел 2 Структуры данных						
Тема 3 Линейные структуры данных Структура данных стек. Динамические структуры данных. Динамический стек. Исключение рекурсии с помощью стека. Применение стека для вычисления значений арифметического выражения. Очередь и дек. Ис-	Лекция	10		С использованием активных методов обучения	ПК-3	32(ПК-3-6) У2(ПК-3-6)

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)		Форма проведения	Планируемые (контролируемые) результаты освоения	
		Для графика 16 недель			Компетенции	Знания, умения, навыки
пользование стека при программировании алгоритма обхода вершин графа в глубину. Использование очереди при программировании алгоритма обхода вершин графа в ширину. Алгоритм поиска путей в графе с помощью стека. Алгоритм поиска направленных путей с помощью очереди.						
Тема 3 Нелинейные структуры данных Нелинейные структуры данных: иерархические списки, деревья, упорядоченные деревья, неупорядоченные деревья, бинарные деревья, ориентированные графы. Циклические списки. Применение циклического списка для решения задачи Джозефуса. Двусвязный циклический список. Представление упорядоченных деревьев. Алгоритм обхода упорядоченного дерева. Структура данных дерево поиска. Список смежности для ориентированного графа.	Лекция	6		С использованием активных методов обучения	ПК-3	32(ПК-3-6) У2(ПК-3-6)
Тема Динамические структуры данных и их применение Программная реализация заданной структуры данных. Определение структуры элемента, разработка функций добавления, удаления и чтения элемента. Разработка тестовой программы. Применение списков для построения и программной реализации алгоритмов.	Лабораторная работа	24		С использованием активных методов обучения	ПК-3	32(ПК-3-6) У2(ПК-3-6) Н3(ПК-3-6)

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)		Форма проведения	Планируемые (контролируемые) результаты освоения	
		Для графика 16 недель			Компетенции	Знания, умения, навыки
	Самостоятельная работа обучающихся	10		Подготовка к лабораторным занятиям	ПК-3	32(ПК-3-6) У2(ПК-3-6)
	Самостоятельная работа обучающихся	6		Изучение теоретических разделов дисциплины	ПК-3	32(ПК-3-6) У2(ПК-3-6)
	Самостоятельная работа обучающихся	6		Выполнение, оформление и подготовка к защите лабораторной работы 3	ПК-3	32(ПК-3-6) У2(ПК-3-6) Н3(ПК-3-6)
	Самостоятельная работа обучающихся	8		Выполнение, оформление и подготовка к защите КР	ПК-3	32(ПК-3-6) У2(ПК-3-6) Н3(ПК-3-6)
	Текущий контроль			Тест	ПК-3	32(ПК-3-6)
	Текущий контроль			Защита лабораторной работы 3	ПК-3	32(ПК-3-6) У2(ПК-3-6) Н3(ПК-3-6)
ИТОГО по разделу 2	Лекции	16		-	-	-
	Лабора-	24		-	-	-

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)		Форма проведения	Планируемые (контролируемые) результаты освоения	
		Для графика 16 недель			Компетенции	Знания, умения, навыки
	торные работы					
	Самостоятельная работа обучающихся	60		-	-	-
Всего часов по разделу 2		90		-	-	-
Курсовая работа		-		-	-	-
Промежуточная аттестация в 6 семестре (итоговая оценка)		-		-	-	-
ИТОГО в шестом семестре	Лекции	36		-	-	-
	Лабораторные работы	54		-	-	-
	Самостоятельная работа обучающихся	90		-	-	-
Всего часов в шестом семестре		180		-	-	-
Седьмой семестр						
Раздел 3 Методы перебора						
Тема 5 Метод перебора с возвратом Формулировка задачи. Проблема построения последо-	Лекция	8		С использованием активных ме-	ПК-3	З1(ПК-3-7) У1(ПК-3-7)

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)		Форма проведения	Планируемые (контролируемые) результаты освоения	
		Для графика 16 недель			Компетенции	Знания, умения, навыки
вательностей, удовлетворяющих заданным отношениям, и ее решение методом перебора с возвратом. Алгоритм перебора с возвратом как метод раскраски. Перебор слов длины n, не содержащих смежных идентичных подтекстов. Рекурсивная реализация алгоритма перебора с возвратом. Пример программы. Решение задачи Гаусса о ферзях рекурсивным алгоритмом перебора с возвратом. Перебор путей в графе. Алгоритм перебора гамильтоновых циклов в графе. Задача Эйлера о шахматном коне и алгоритм ее решения. Алгоритм решения задачи о размене денег. Генерация подмножеств конечного множества. Алгоритм перебора разложений числа в сумму. Генерация монотонно неубывающих последовательностей. Алгоритм генерации разбиений множества. Организация переменного числа вложенных циклов. Недетерминированный алгоритм распознавания и класс NP-полных задач. Шесть основных NP-полных задач.				тодов обучения		
Тема Методы перебора слов, функций и последовательностей чисел Программная реализация перебора функций, слов длины n, последовательностей чисел.	Лабораторная работа	6		С использованием активных методов обучения	ПК-3	31(ПК-3-7) У1(ПК-3-7) Н1(ПК-3-7)
Тема Реализация метода перебора с возвратом	Лабораторная работа	6		С использованием	ПК-3	31(ПК-3-7)

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)		Форма проведения	Планируемые (контролируемые) результаты освоения	
		Для графика 16 недель			Компетенции	Знания, умения, навыки
Программная реализация перебора с возвратом путей в графе с помощью рекурсивных и нерекурсивных подпрограмм.	торная работа			ем активных методов обучения		У1(ПК-3-7) Н1(ПК-3-7)
	Самостоятельная работа обучающихся	10		Подготовка к лабораторным занятиям	ПК-3	31(ПК-3-7) У1(ПК-3-7)
	Самостоятельная работа обучающихся	10		Изучение теоретических разделов дисциплины	ПК-3	31(ПК-3-7) У1(ПК-3-7)
	Самостоятельная работа обучающихся	10		Выполнение, оформление и подготовка к защите лабораторных работ 1,2	ПК-3	31(ПК-3-7) У1(ПК-3-7) Н1(ПК-3-7)
	Текущий контроль			Тест	ПК-3	31(ПК-3-7) У1(ПК-3-7)
	Текущий контроль			Защита лабораторных работ 1,2	ПК-3	31(ПК-3-7) У1(ПК-3-7) Н1(ПК-3-7)
ИТОГО по разделу 3	Лекции	8		-	-	-
	Лаборатор-	12		-	-	-

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)		Форма проведения	Планируемые (контролируемые) результаты освоения	
		Для графика 16 недель			Компетенции	Знания, умения, навыки
	ные работы					
	Самостоятельная работа обучающихся	30		-	-	-
Всего часов по разделу 3		54		-	-	-
Раздел 4 Методы сортировки						
Тема 6 Внутренняя и внешняя сортировки Основные понятия. Метод простых вставок. Метод пузырьков. Сортировка методом выбора. Сортировка подсчетом. Распределяющий подсчет. Модификации метода вставок. Пример программы. Сортировка методом Шелла. Быстрая сортировка Хоара. Обменная поразрядная сортировка. Пример программы. Сортировочные сети и алгоритм Бетчера. Турнир с выбыванием. Пирамидальная сортировка. Пример программы. Очередь с приоритетами. Алгоритм двухпутевого слияния. Сортировка простым двухпутевым слиянием. Естественное двухпутевое слияние. Двухленточная сортировка на основе поразрядной. Трехленточная сортировка слиянием. Четырехленточная сортировка слиянием. Пример программы.	Лекция	8		С использованием активных методов обучения	ПК-3	31(ПК-3-7) У1(ПК-3-7)
Тема Алгоритмы сортировки	Лабора-	12		С использовани-	ПК-3	31(ПК-3-7)

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)		Форма проведения	Планируемые (контролируемые) результаты освоения	
		Для графика 16 недель			Компетенции	Знания, умения, навыки
Программная реализация различных методов сортировки и сравнение их между собой по времени выполнения программы, числу сравнений и числу перестановок.	торная работа			ем активных методов обучения		У1(ПК-3-7) Н1(ПК-3-7)
	Самостоятельная работа обучающихся	10		Подготовка к лабораторным занятиям	ПК-3	31(ПК-3-7) У1(ПК-3-7)
	Самостоятельная работа обучающихся	6		Изучение теоретических разделов дисциплины	ПК-3	31(ПК-3-7) У1(ПК-3-7)
	Самостоятельная работа обучающихся	14		Выполнение, оформление и подготовка к защите РГР	ПК-3	31(ПК-3-7) У1(ПК-3-7) Н1(ПК-3-7)
	Текущий контроль			Тест	ПК-3	31(ПК-3-7) У1(ПК-3-7)
	Текущий контроль			Защита РГР	ПК-3	31(ПК-3-7) У1(ПК-3-7) Н1(ПК-3-7)
ИТОГО по разделу 4	Лекции	6		-	-	-
	Лабораторные работы	12		-	-	-

Наименование разделов, тем и содержание материала	Компонент учебного плана	Трудоемкость (в часах)		Форма проведения	Планируемые (контролируемые) результаты освоения	
		Для графика 16 недель			Компетенции	Знания, умения, навыки
	Самостоятельная работа обучающихся	30		-	-	-
Всего часов по разделу 4		52		-	-	-
ИТОГО в седьмом семестре	Лекции	16		-	-	-
	Лабораторные работы	32		-	-	-
	Самостоятельная работа обучающихся	60		-	-	-
Промежуточная аттестация по дисциплине в 7 семестре (экзамен)		36				
ИТОГО по дисциплине	Лекции	48		-	-	-
	Лабораторные работы	80		-	-	-
	Самостоятельная работа обучающихся	150		-	-	-
ИТОГО: общая трудоемкость дисциплины 324 часа, в том числе с использованием активных методов обучения 30 часов						

6 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)

Самостоятельная работа обучающихся, осваивающих дисциплину «Структуры данных и алгоритмы», состоит из следующих компонентов:

- изучение теоретических разделов дисциплины;
- подготовка к лабораторным занятиям;
- выполнение, оформление и подготовка к защите лабораторных работ;
- выполнение, оформление и подготовка к защите курсовой работы (КР);
- выполнение, оформление и подготовка к защите РГР.

Для успешного выполнения всех разделов самостоятельной работы учащимся рекомендуется использовать следующее учебно-методическое обеспечение:

1 Хусаинов, А.А. Структуры и алгоритмы обработки данных. Часть 1: Учеб. пособие/А.А. Хусаинов, Н.Н. Михайлова. – Комсомольск-на-Амуре: ГОУВПО «КНАГТУ», 2007. – 86с. // Собственные электронные ресурсы КНАГТУ: виртуальная библиотека института новых информационных технологий – Режим доступа: <http://www.initkms.ru/library/main?>.

2 Хусаинов, А.А. Структуры и алгоритмы обработки данных. Часть 2: Учеб. пособие/А.А. Хусаинов, Н.Н. Михайлова. – Комсомольск-на-Амуре: ГОУВПО «КНАГТУ», 2007. – 94с. // Собственные электронные ресурсы КНАГТУ: виртуальная библиотека института новых информационных технологий – Режим доступа: <http://www.initkms.ru/library/main?>.

3 РД ФГБОУ ВО «КНАГТУ» 013-2016. Текстовые студенческие работы. Правила оформления. – Введ. 2016-03-04. – Комсомольск-на-Амуре: ФГБОУ ВО «КНАГТУ», 2016. – 55 с.

Рекомендуемый график выполнения самостоятельной работы студента представлен в таблицах 4.1, 4.2.

Ниже приведены общие рекомендации по организации самостоятельной работы студентов.

В процессе обучения, работа студента над выполнением учебного плана складывается из двух составляющих: одна из них – это аудиторная работа в вузе по расписанию занятий, другая – внеаудиторная самостоятельная работа. Задания и материалы для самостоятельной работы выдаются во время учебных занятий по расписанию, на этих же занятиях преподаватель осуществляет контроль выполнения самостоятельной работы.

Для формирования у студентов практических навыков организации данных и использования типовых структур данных при выполнении заданий лабораторных работ и КР в шестом семестре студенту необходимо уметь организовывать данные в разрабатываемом ПО с использованием типовых структур данных. А для формирования у студентов практических навыков использования стандартных алгоритмов при организации и обработке данных на ЭВМ при выполнении заданий лабораторных работ РГР в седьмом

семестре студенту необходимо знать типовые алгоритмы обработки данных на ЭВМ и уметь выбирать алгоритмы обработки данных при решении практических задач. И в рамках подготовки к лабораторным занятиям, и при изучении теоретических разделов дисциплины, студенту необходимо осуществлять поиск, хранение, обработку и анализ информации в сети Интернет и в технической литературе, как при изучении теории организации и структурирования данных на ЭВМ, а также типовых алгоритмов обработки данных на ЭВМ, так и при самостоятельном проектировании, конструировании, отладке и тестировании ПО с использованием типовых структур данных и стандартных алгоритмов организации и обработки данных на ЭВМ.

При подготовке к защите лабораторных работ, РГР и КР студенту необходимо обратить внимание как на проработку теоретических вопросов по данной теме, так и на обоснование выбора средств языка программирования при разработке ПО, а также тестирование разработанного ПО при различных входных данных и параметрах.

При оформлении отчетов к лабораторным работам, РГР и КР студенту необходимо студенту необходимо осуществить поиск, хранение, обработку и анализ информации в сети Интернет и в технической литературе. Также при оформлении отчетов необходимо строго следовать РД ФГБОУ ВО «КНАГТУ» 013-2016. «Текстовые студенческие работы. Правила оформления».

После успешного выполнения и защиты РГР на лабораторном занятии отчет по РГР студенту необходимо разместить в его личном кабинете, расположенном на официальном сайте университета в информационно-телекоммуникационной сети «Интернет» по адресу <https://student.knastu.ru>.

Выполнение курсовой работы

Тема данной курсовой работы «Программирование игр с выигрышной стратегией».

Заданием к данной курсовой работе является проектирование, разработка и тестирование ПО для реализации одной из игр с выигрышной стратегией. Студентам предоставляется право самостоятельно выбрать индивидуальное задание.

При выполнении курсовой работы необходимо следовать рекомендациям СТО У.003-2017 «Курсовое проектирование. Положение», в котором определены порядок организации, выполнения и защиты курсовой работы, в том числе структура и содержание пояснительной записки к курсовой работе.

После успешного выполнения и защиты КР на лабораторном занятии отчет по КР студенту необходимо разместить в его личном кабинете, расположенном на официальном сайте университета в информационно-телекоммуникационной сети «Интернет» по адресу <https://student.knastu.ru>.

Таблица 4.1 – Рекомендуемый график выполнения самостоятельной работы студентов в шестом семестре (16 недель)

Виды самостоятельной работы	Число академических часов в неделю																		Итого по видам работы
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16			
Подготовка к лабораторным занятиям	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1			16
Изучение теоретических разделов дисциплины	2	2	1	1	1	1			1	1	1			1	1	1			14
Выполнение, оформление и подготовка к защите лабораторных работ	1	1	1	1	1	1	1	1	1	2	1	1	1						14
Выполнение и подготовка к защите курсовой работы			2	2	2	2	4	4	4	4	3	3	4	4	4	4			46
ИТОГО в 6 семестре:	4	4	5	5	5	5	6	6	7	8	6	5	6	6	6	6			90

Таблица 4.2 – Рекомендуемый график выполнения самостоятельной работы студентов в седьмом семестре (16 недель)

Вид самостоятельной работы	Число часов в неделю																	Итого по видам работ
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16		
Подготовка к лабораторным занятиям	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		16
Изучение теоретических разделов дисциплины	1	1		1	1		1	1		1	1	1		1	1			11
Выполнение, оформление и подготовка к защите лабораторных работ	1	1	1	2	1	1	1	2	2									12
Выполнение, оформление и подготовка к защите РГР										3	3	3	3	3	3	3		21
ИТОГО в 7 семестре:	3	3	2	4	3	2	3	4	3	5	5	5	4	5	5	4		60

**7 Фонд оценочных средств для проведения текущего контроля
и промежуточной аттестации обучающихся по дисциплине (модулю)**

Таблица 5 – Паспорт фонда оценочных средств

Контролируемые разделы (темы) дисциплины	Код контроли- руемой компе- тенции (или ее части)	Наименование оценочного средства	Показатели оценки
<i>Шестой семестр</i>			
Тема1- Рекурсивные ал- горитмы и про- граммы	ПК-3-6	Защита лабораторной работы 1 Тест Курсовая работа	Умеет находить рекур- рентные соотношения. Умеет строить рекур- сивные алгоритмы для решения конкретных задач. Умеет разраба- тывать и тестировать рекурсивные подпро- граммы.
Тема2-Методы обхода вершин графа	ПК-3-6	Защита лабораторной работы 2 Тест Курсовая работа	Применяет теорию графов при решении конкретных задач. Ис- пользует структуру данных массив при программной реализа- ции алгоритмов на графах.
Тема3- Линейные струк- туры данных	ПК-3-6	Защита лабораторной работы 3 Тест Курсовая работа	Умеет организовывать данные в разрабатыва- емом ПО с помощью линейных структур данных. Умеет ис- пользовать стек и оче- редь для реализации алгоритмов на графах.
Тема4- Нелинейные структуры дан- ных	ПК-3-6	Защита лабораторной работы 3 Тест Курсовая работа	Умеет организовывать данные в разрабатыва- емом ПО с помощью нелинейных структур данных. Умеет приме- нять упорядоченное дерево и дерево пойс- ка.
<i>Седьмой семестр</i>			
Тема5- Метод перебора с возвратом	ПК-3-7	Защита лабораторных работ 1,2 Тест Экзамен	Умеет разрабатывать ПО с использованием рекурсивного и нере- курсивного алгоритма перебора с возвратом для перебора функций, слов длины n, после-

			довательностей чисел и путей в графе.
Темаб- Внутренняя и внешняя сорти- ровки	ПК-3-7	Защита РГР Тест Экзамен	Умеет разрабатывать ПО с использованием различных алгоритмов сортировки данных. Умеет проводить сравнение между собой различных алгоритмов сортировки.

Промежуточная аттестация в шестом семестре у студентов набора 2016 года и далее проводится в форме итоговой оценки. Промежуточная аттестация в седьмом семестре у студентов набора 2016 года и далее проводится в форме экзамена.

Методические материалы, определяющие процедуры оценивания знаний, умений, навыков, характеризующих этапы формирования компетенций, представлены в виде технологической карты дисциплины (таблицы 6.1-6.2).

В таблице 6.1 сроки выполнения (столбцы 3 и 4) приведены для семестра продолжительностью в 18 недель (столбец 3) и 17 недель (столбец 4).

Таблица 6.1 – Технологическая карта для шестого семестра

	Наименование оценочного средства	Сроки выполнения/ 18	Сроки выполнения/ 17	Шкала оценивания	Критерии оценивания
шестой семестр					
Промежуточная аттестация в форме итоговой оценки					
1.	Тест	4-ая неделя	4-ая неделя	4 балла	4 – правильные ответы на все вопроса, 3- правильные ответы на три вопроса, 2- правильные ответы на два вопроса, 1 – правильный ответ на один вопрос, 0 - все ответы неправильные
2.	Защита лабораторной работы 1	4-ая неделя	4-ая неделя	28 баллов	28–лабораторная работа выполнена полностью, правильно, своевременно, даны полные ответы на дополнительные вопросы во время защиты работы, студент показал отличные владения навыками организации данных в разрабатываемом ПО, отчет выполнен аккуратно и в соответствии с предъявляемыми требованиями, 19-лабораторная работа выполнена с замечаниями, студент показал хорошее владение навыками организации данных в разрабатываемом ПО, не выдержаны сроки выполнения работы, даны неполные ответы на дополнительные вопросы во время защиты работы, 9- студент выполнил работу с существенными неточностями, не соблюдены сроки выполнения работы, студент показал удовлетворительное владение навыками организации данных в

	Наименование оценочного средства	Сроки выполнения/ 18	Сроки выполнения/ 17	Шкала оценивания	Критерии оценивания
					разрабатываемом ПО, 0- задание не выполнено.
3.	Тест	8-ая неделя	8-ая неделя	4 балла	4 – правильные ответы на все вопроса, 3- правильные ответы на три вопроса, 2- правильные ответы на два вопроса, 1 – правильный ответ на один вопрос, 0 - все ответы неправильные
4.	Защита лабораторной работы 2	8-ая неделя	8-ая неделя	28 баллов	28–лабораторная работа выполнена полностью, правильно, своевременно, даны полные ответы на дополнительные вопросы во время защиты работы, студент показал отличные владения навыками организации данных в разрабатываемом ПО, отчет выполнен аккуратно и в соответствии с предъявляемыми требованиями, 19-лабораторная работа выполнена с замечаниями, студент показал хорошее владение навыками организации данных в разрабатываемом ПО, не выдержаны сроки выполнения работы, даны неполные ответы на дополнительные вопросы во время защиты работы, 9- студент выполнил работу с существенными неточностями, не соблюдены сроки выполнения работы, студент показал удовлетворительное владение навыками организации данных в разрабатываемом ПО, 0- задание не выполнено.
5.	Тест	13-ая неделя	12-ая неделя	4 балла	4 – правильные ответы на все вопроса, 3- правильные ответы на три вопроса, 2- правильные ответы на два вопроса, 1 – правильный ответ на один вопрос, 0 - все ответы неправильные
6.	Защита лабораторной работы 3	13-ая неделя	12-ая неделя	32 балла	32–лабораторная работа выполнена полностью, правильно, своевременно, даны полные ответы на дополнительные вопросы во время защиты работы, студент показал отличные владения навыками использования типовых структур данных в разрабатываемом ПО, отчет выполнен аккуратно и в соответствии с предъявляемыми требованиями, 20–лабораторная работа выполнена с замечаниями, студент показал хорошее владение навыками использования типовых структур данных в разрабатываемом ПО, не выдержаны сроки выполнения работы, даны неполные ответы на дополнительные вопросы во время защиты работы, 10- студент выполнил работу с существенными неточностями, не соблюдены сроки выполнения работы, студент показал удовлетворительное владение навыками использования типовых структур данных в разрабатываемом ПО, 0- задание не выполнено.
ИТОГО:		-		100 бал-	-

	Наименование оценочного средства	Сроки выполнения/ 18	Сроки выполнения/ 17	Шкала оценки	Критерии оценивания
				ЛОВ	
Критерии оценки результатов обучения по дисциплине: 0 – 64 % от максимально возможной суммы баллов – «неудовлетворительно» (недостаточный уровень для текущей аттестации по дисциплине); 65 – 74 % от максимально возможной суммы баллов – «удовлетворительно» (пороговый (минимальный) уровень); 75 – 84 % от максимально возможной суммы баллов – «хорошо» (средний уровень); 85 – 100 % от максимально возможной суммы баллов – «отлично» (высокий (максимальный) уровень)					
7.	Курсовая работа	В конце семестра	В конце семестра	5 баллов	ОТЛИЧНО- студент правильно выполнил курсовую работу. Показал отличные владения навыками применения полученных знаний и умений при решении профессиональных задач в рамках усвоенного учебного материала. Ответил на все дополнительные вопросы на защите. ХОРОШО - студент выполнил курсовую работу с небольшими неточностями. Показал хорошие владения навыками применения полученных знаний и умений при решении профессиональных задач в рамках усвоенного учебного материала. Ответил на большинство дополнительных вопросов на защите. УДОВЛЕТВОРИТЕЛЬНО - студент выполнил курсовую работу с существенными неточностями. Показал удовлетворительное владение навыками применения полученных знаний и умений при решении профессиональных задач в рамках усвоенного учебного материала. При ответах на дополнительные вопросы на защите было допущено много неточностей. НЕУДОВЛЕТВОРИТЕЛЬНО - при выполнении курсовой работы студент продемонстрировал недостаточный уровень владения навыками применения полученных знаний и умений при решении профессиональных задач в рамках усвоенного учебного материала. При ответах на дополнительные вопросы на защите было допущено множество неточностей.

В таблице 6.2 представлена технологическая карта дисциплины для седьмого семестра.

Таблица 6.2 – Технологическая карта для осеннего семестра

	Наименование оценочного средства	Сроки выполнения	Шкала оценивания	Критерии оценивания
седьмой семестр <i>Промежуточная аттестация в форме экзамена</i>				
1.	Защита лабораторной работы 1	4-ая неделя	8 баллов	8 - лабораторная работа выполнена полностью, правильно, своевременно, даны полные ответы на дополнительные вопросы во время защиты работы, студент показал отличное владения навыками использования стандартных алгоритмов при организации и обработке данных, отчет выполнен аккуратно и в соответствии с предъявляемыми требованиями, 5–лабораторная работа выполнена с замечаниями, студент показал хорошее владение навыками использования стандартных алгоритмов при организации и обработке данных, не выдержаны сроки выполнения работы, даны неполные ответы на дополнительные вопросы во время защиты работы, 2- студент выполнил работу с существенными неточностями, не соблюдены сроки выполнения работы, студент показал удовлетворительное владение навыками использования стандартных алгоритмов при организации и обработке данных, 0- задание не выполнено.
2.	Тест	7-ая неделя	4 балла	4 – правильные ответы на все вопроса, 3- правильные ответы на три вопроса, 2- правильные ответы на два вопроса, 1 – правильный ответ на один вопрос, 0 - все ответы неправильные
3.	Защита лабораторной работы 2	7-ая неделя	8 баллов	8 - лабораторная работа выполнена полностью, правильно, своевременно, даны полные ответы на дополнительные вопросы во время защиты работы, студент показал отличное владения навыками использования стандартных алгоритмов при организации и обработке данных, отчет выполнен аккуратно и в соответствии с предъявляемыми требованиями, 5–лабораторная работа выполнена с замечаниями, студент показал хорошее владение навыками использования стандартных алгоритмов при организации и обработке данных, не выдержаны сроки выполнения работы, даны неполные ответы на дополнительные вопросы во время защиты работы, 2- студент выполнил работу с существенными неточностями, не соблюдены сроки выполнения работы, студент показал удовлетворительное владение навыками использования стандартных алгоритмов при организации и обработке данных, 0- задание не выполнено.
4.	Тест	17-ая неделя	4 балла	4 – правильные ответы на все вопроса, 3- правильные ответы на три вопроса, 2- правильные ответы на два вопроса, 1 – правильный ответ на один вопрос, 0 - все ответы неправильные
5.	Защита РГР	17-ая неделя	16 балла	16–РГР выполнена полностью, правильно, своевременно, даны полные ответы на дополнительные вопросы во время защиты работы, студент показал отличное владения навыками использования алгоритмов сорти-

	Наименование оценочного средства	Сроки выполнения	Шкала оценивания	Критерии оценивания
				ровки при обработке данных, отчет выполнен аккуратно и в соответствии с предъявляемыми требованиями, 10–РГР выполнена с замечаниями, студент показал хорошее владение навыками использования алгоритмов сортировки при обработке данных, не выдержаны сроки выполнения работы, даны неполные ответы на дополнительные вопросы во время защиты работы, 5-студент выполнил работу с существенными неточностями, не соблюдены сроки выполнения работы, студент показал удовлетворительное владение навыками использования алгоритмов сортировки при обработке данных, 0- задание не выполнено.
ИТОГО текущий контроль : 40 баллов				
6.	Экзамен	Два вопроса оценивание уровня усвоенных знаний	по 20 баллов за каждый вопрос	20 баллов – студент дал полный и точный ответ на теоретический вопрос, ответил на дополнительные вопросы, 16 - студент ответил на теоретический вопрос билета с небольшими неточностями, показал хорошие знания в рамках усвоенного учебного материала, ответил на большинство дополнительных вопросов, 12-студент ответил на теоретический вопрос билета с существенными неточностями, показал удовлетворительные знания в рамках усвоенного учебного материала, при ответах на дополнительные вопросы было допущено много неточностей, 0 баллов - в ответе более трех грубых ошибок или задание не выполнено.
		Практическое задание оценивание уровня усвоенных умений и навыков	20 баллов	20 баллов - студент правильно выполнил практическое задание билета, показал отличные умения в рамках усвоенного учебного материала, продемонстрировал владение навыками разработки ПО, ответил на все дополнительные вопросы, 16- студент выполнил практическое задание билета с небольшими неточностями, показал хорошие умения и владение навыками разработки ПО в рамках усвоенного учебного материала, ответил на большинство дополнительных вопросов, 10 - студент выполнил практическое задание билета с существенными неточностями, показал удовлетворительные умения и владение навыками разработки ПО в рамках усвоенного учебного материала, при ответах на дополнительные вопросы было допущено много неточностей, 0 баллов - при выполнении практического задания билета студент продемонстрировал недостаточный уровень умений и навыков, при ответах на дополнительные вопросы были допущены грубые ошибки.
Текущий контроль:		-	40	-
Экзамен:		-	60	-
ИТОГО:		-	100 бал-	-

	Наименование оценочного средства	Сроки выполнения	Шкала оценки важности	Критерии оценивания
			ЛОВ	
Критерии оценки результатов обучения по дисциплине: 0 – 64 % от максимально возможной суммы баллов – «неудовлетворительно» (недостаточный уровень для текущей аттестации по дисциплине); 65 – 74 % от максимально возможной суммы баллов – «удовлетворительно» (пороговый (минимальный) уровень); 75 – 84 % от максимально возможной суммы баллов – «хорошо» (средний уровень); 85 – 100 % от максимально возможной суммы баллов – «отлично» (высокий (максимальный) уровень)				

Задания для текущего контроля

Шестой семестр

Лабораторная работа 1

«Разработка рекурсивных подпрограмм»

1. Разработать рекурсивный алгоритм, написать реализующую его рекурсивную подпрограмму на Си++ и проверить её на тестовых данных.

Варианты заданий

1. Имеются гири весом $w_0, w_1, \dots, w_{n-1}$. Надо определить, можно ли из них составить сумму, равную p . В случае положительного ответа вывести веса гирь, составляющих этот вес.

2. С клавиатуры вводится алгебраическое выражение, состоящее из целых чисел, скобок и операций ‘–’ (унарный минус), ‘<’ (сдвиг влево), ‘>’ (сдвиг вправо). Вычислить значение этого выражения.

3. Определить такую рекурсивную функцию, что ее значение $fix(f, a, b, eps)$ равно корню уравнения $x = f(x)$ на отрезке $[a, b]$, с точностью eps . Известно, что функция f непрерывна и отображает отрезок $[a, b]$ в себя. Для нахождения приближенного значения корня рекомендуется использовать метод деления отрезка пополам.

4. Вычислить значение логического выражения, заданного с клавиатуры, состоящего из натуральных чисел, скобок и операций ‘&’ (поразрядное логическое ‘и’) и ‘~’ (поразрядное логическое ‘не’). Например, $\sim 8 \& (64 \& 125)$ будет равно $11110111 \& 1000000 = 64$. Операции ‘&’ и ‘~’ предполагаются имеющими одинаковый приоритет.

5. Вычислить наибольший общий делитель двух натуральных чисел, введенных с клавиатуры.

6. Имеются гири весом $w_0 < w_1 < \dots < w_{n-1}$. Для каждого из чисел w_i достаточное число гирь веса w_i . Какое минимальное число гирь необходимо для получения суммарного веса t ?

7. Символьная строка t называется обращением строки s , если ее длина равна длине строки s , а символы расположены в обратном порядке. Например, обращением строки $s = \text{"abcbs"}$ является $t = \text{"cbsba"}$. Написать подпрограмму обращения строки, имеющую прототип

`void revers( char *s, int i, int j );`
 вызывающуюся как `reverse( t, 0, strlen(t) )`.

8. Определить рекурсивную функцию `int Lat`, возвращающую количество латинских букв в текстовом файле.

9. Найти все цифры в текстовом файле и вывести их в обратном порядке.

10. Заданы целое число x и текстовый файл, содержащий записанные через пробелы целые числа. Вывести сначала все числа, меньшие X , а затем все остальные.

11. Вычислить сумму

$$\sum_{i_1=1}^N \sum_{i_2=1}^N \dots \sum_{i_m=1}^N \frac{1}{i_1 + i_2 + \dots + i_m}$$

Указание. Обозначить $P(N, m, i) = \sum_{i_1=1}^N \sum_{i_2=1}^N \dots \sum_{i_m=1}^N \frac{1}{i_1 + i_2 + \dots + i_m + i}$ и доказать

соотношение $P(N, m, i) = \sum_{j=1}^N P(N, m-1, i+j)$. Тогда искомая сумма равна $P(N, m, 0)$.

12. С клавиатуры вводится логическое выражение, состоящее из булевых констант 0 (ложь) и 1 (истина), скобок и логических символов $\vee$ ("или") и $\neg$ ("не"). Написать программу вычисления значения этого логического выражения.

13. Имеются гири весом $w_0 < w_1 < \dots < w_n$. Гирь каждого веса – любое достаточное количество. Найти хотя бы один набор гирь, в сумме дающих заданное с клавиатуры число. Например, для гирь $2 < 3 < 5$ и числа 6, таким решением будет сумма $3 + 3$ (или $2 + 2 + 2$).

14. Задан текстовый файл. Вывести сначала содержащиеся в нем цифры, а затем – латинские буквы в обратном порядке.

15. Вычислить коэффициенты $a[0], a[1], \dots, a[n-1]$ многочлена

$$p(x) = a[0] + a[1]x + \dots + a[n-1]x^{n-1} + x^n,$$

если известны его действительные корни $x[1], x[2], \dots, x[n]$.

Указание. Обозначив через $a(n, i)$ коэффициент $a[i]$ многочлена степени n , легко получить рекуррентные соотношения, с помощью теоремы Безу:

$$a(n, n-1) = a(n-1, n-2) - x[n]$$

$$a(n, i) = a(n-1, i-1) - a(n-1, i) * x[n], \quad 1 \leq i < n-1$$

$$a(n, 0) = -a(n-1, 0) * x[n]$$

Значения $a(n, i)$ для всех $0 \leq i \leq n-1$ вычисляются с помощью рекурсивной функции, построенной по этим соотношениям.

16. Задана матрица $a[i][j]$ цен на авиабилеты между городами i и j . Всего рассматривается 6 городов. Генерируя перестановки $(i_1, i_2, i_3, i_4, i_5)$, найти путь коммивояжера

$$0, i_1, i_2, i_3, i_4, i_5, 0$$

минимальной стоимости и записать его в массив `int b[7]`.

17. Две примыкающие равные строки называются идентичными подтекстами, например, строка “abcdecde” содержит идентичные подтексты “cde”. Используя рекурсивную функцию, проверить слово на присутствие смежных идентичных подтекстов. Под словом понимается строка, состоящая из букв и цифр.

18. Задан текстовый файл. Игнорируя пробелы, вывести сначала все цифры, а затем – остальные символы в обратном порядке.

19. Найти число $B(m)$ всех разбиений множества, состоящего из m элементов (число Белла). Воспользоваться формулой

$$B(m+1) = \sum_{i=0}^m C_m^i B(i)$$

и значением $B(0) = 1$.

20. Построить треугольную таблицу для чисел разбиений $S(m, n)$ множества $\{1, 2, \dots, m\}$ на n блоков (чисел Стирлинга второго рода), при $0 \leq n \leq m$. Воспользоваться соотношениями $S(m, 0) = 0$, при $m > 0$; $S(m, m) = S(0, 0) = 1$; $S(m, n) = 0$, при $n > m$; $S(m, n) = S(m-1, n-1) + nS(m-1, n)$, при $1 \leq n \leq m$.

21. Задан текстовый файл, содержащий целые числа, записанные через пробелы. Вывести сначала все четные числа, а затем в обратном порядке все нечетные числа, делящиеся на 3.

22. Имеется набор монет $(p_0, p_1, \dots, p_{n-1})$ по одной монете каждого достоинства. Определить, можно ли этими монетами разменять сумму q .

23. Написать рекурсивную подпрограмму проверки строки символов на симметричность. Функция должна вызываться как `t=sym(s, strlen(s))`, где строка определена как `char s[]`, и возвращать значение `t=1` если строка симметрична и `t=0` в противном случае.

24. Текстовый файл содержит числа с плавающей точкой, записанные через пробелы. Вывести отрицательные числа в обратном порядке.

25. Написать рекурсивную подпрограмму для вычисления значений функции $f(\text{int } n)$, заданной с помощью рекуррентного соотношения

$$f(n) = \begin{cases} 1, & \text{при } n = 1, \\ \sum_{i=2}^n f\left(\frac{n}{i}\right), & \text{при } n \geq 2. \end{cases}$$

Здесь $\frac{n}{i}$ - целая часть дроби, полученной делением n на i .

26. Вычислить число двоичных деревьев $T(n)$, имеющих n вершин, с помощью рекуррентного соотношения

$$T(n+1) = T(0) T(n) + T(1) T(n-1) + \dots + T(i) T(n-i) + \dots + T(n)T(0).$$

Проверить формулу $T(n) = \frac{1}{n+1} C_{2n}^n$ для $n \leq 5$, применяя для вычисления C_n^k

соотношения $C_n^0 = C_n^n = 1$, $C_n^k = C_{n-1}^k + C_{n-1}^{k-1}$.

27. С клавиатуры вводится логическое выражение, состоящее из целых чисел, логических операций $\&$ и $|$ и скобок. Вычислить значение этого логического выражения, если операции $\&$ и $|$ вычисляются поразрядно и имеют равные приоритеты.

28. Текстовый файл состоит из слов, записанных через пробелы. Слова состоят из латинских букв и цифр. Вывести сначала прописные (большие), а потом строчные (маленькие) буквы в обратном порядке. Цифры не выводить.

29. С клавиатуры вводится арифметическое выражение, состоящее из целых чисел, операций $*$ и $\%$ (остаток от деления) и скобок. Написать программу для вычисления этого выражения.

30. Число $f(n)$ путей на двумерной целочисленной решетке, не имеющих самопересечений и состоящих из n звеньев типа $(1, 0)$, $(-1, 0)$ и $(0, 1)$, удовлетворяет соотношениям:

$$f(n) = 2f(n-1) + f(n-2), \quad f(0) = 1, \quad f(1) = 3.$$

Вычислить $f(n)$ и проверить с точностью 0.01 формулу

$$f(n) = \frac{1}{2} ((1 + \sqrt{2})^{n+1} + (1 - \sqrt{2})^{n+1}).$$

Лабораторная работа 2 «Алгоритмы на графах»

1. Разработать алгоритм обхода вершин графа, нахождения пути или направленного пути в графе, написать реализующую его подпрограмму на Си++ и выдать результаты.

Варианты заданий

Пояснение. В вариантах 1 – 6 заданы n городов, между которыми определены рейсы самолетов. Матрица $n \times n$ состоит из чисел $a_{ij} = 1$, если существует рейс самолета из i в j , и 0 – в противном случае.

1. Найти все города, в которые можно попасть из города p самолетом.
2. Заданы два города – p и q . Можно ли попасть из p в q ?
3. Изолированными странами называются группы городов, такие, что между городами различных групп не существует летных сообщений. Найти число изолированных стран.
4. Найти по крайней мере один маршрут между заданными городами p и q .
5. Для заданной вершины p найти маршруты в другие города, хотя бы один маршрут для каждого города.

6. Найти маршрут из p в q и вывести посещаемые города в обратном направлении.

Пояснение. В вариантах 7 – 14 заданы n человек и два массива натуральных чисел **mother**[n] и **father**[n], такие, что **mother**[i] – номер матери i -го человека, а **father**[i] – номер его отца, для каждого i , удовлетворяющего неравенствам $0 \leq i \leq n-1$.

7. Найти всех предков человека с номером p .

8. Найти всех потомков человека с номером p .

9. Найти всех родственников человека с номером p .

10. Заданы два человека – p и q . Ответить, являются ли они родственниками.

11. Родом называется максимальное подмножество людей, любые два из которого являются родственниками. Найти число родов, содержащихся в данном множестве $\{0, 1, 2, \dots, n-1\}$.

12. Заданы два человека p и q . Найти цепь детей и родителей, соединяющих p и q , если такая цепь существует.

13. Заданы два человека p и q . Найти соединяющую их цепь родителей, доказывающих, что p – потомок q , при условии, что это верно.

14. Заданы два человека p и q . Определить, имеют ли они общего предка.

Пояснение. В вариантах 15 – 18 задан лабиринт, соединяющий узлы $0, 1, 2, \dots, n-1$. Узлы лабиринта $0 \leq i, j \leq n-1$ считаются соединенными тоннелями, если $a_{ij} = 1$. Узел 0 является выходом из лабиринта.

15. Найти все узлы, из которых существуют пути к выходу.

16. Задан узел p . Существует ли путь к выходу из узла p ?

17. Задан узел p . Найти путь в лабиринте, приводящий к выходу.

18. Найти пути к выходу из всех узлов лабиринта.

Пояснение. В вариантах 19 – 23 через (x, y) обозначается поле шахматной доски, находящееся на горизонтали y и на вертикали x .

19. Из шахматной доски удалили клетки $(a_0, b_0), (a_1, b_1), \dots, (a_{n-1}, b_{n-1})$. Найти все поля, в которые можно попасть конем из заданного поля (x, y) .

20. Из шахматной доски удалили клетки $(a_0, b_0), (a_1, b_1), \dots, (a_{n-1}, b_{n-1})$. Определить, можно ли попасть конем с поля (x, y) на поле (x', y') .

21. Из шахматной доски удалили клетки $(a_0, b_0), (a_1, b_1), \dots, (a_{n-1}, b_{n-1})$. Разбить оставшиеся поля на подмножества, поля которых связаны маршрутами шахматного коня.

22. Для каждого поля шахматной доски найти по крайней мере один маршрут шахматного коня в поле $(0, 0)$.

23. Найти маршруты шахматного коня во все поля шахматной доски из поля $(0, 0)$, по одному маршруту для каждого поля.

Пояснение. В вариантах 24 – 30 заданы n человек и квадратная матрица (a_{ij}) размерности n , элементы которой принимают значения $a_{ij} = 1$, если i – родитель человека с номером j , $0 \leq i, j \leq n-1$, в других случаях $a_{ij} = 0$.

24. Найти всех предков человека с номером p .

25. Найти всех потомков человека с номером p .

26. Найти всех родственников человека с номером p .

27. Ответить, являются ли родственниками заданные p и q .

28. Найти все роды, содержащиеся во множестве $\{0, 1, 2, \dots, n-1\}$.

Определение рода см. в варианте 11.

29. Заданы два человека p и q . Найти цепь детей и родителей, соединяющих p и q и содержащую информацию о родственных связях.

30. Найти цепь предков человека p , соединяющую его с предком q .

Лабораторная работа 3 «Динамические структуры данных и их применение»

Задание 1. Организовать заданную структуру данных (очередь, стек, дек, односвязный циклический список, двухсвязный циклический список, дерево поиска). Определить структуру элемента и написать подпрограммы добавления, удаления и чтения элемента. Написать тестовую программу.

Варианты заданий приведены в таблице 7.

Таблица 7 – Варианты заданий

	Очередь	Стек	Дек	Односвязный циклический список	Двухсвязный циклический список	Дерево поиска
точка (x, y) плоскости	1	2	3	4	5	6
точка (x, y, z) пространства	7	8	9	10	11	12
Строка символов	13	14	15	16	17	18
Символ s и позиция (x, y)	19	20	21	22	23	24
Длинное целое число	25	26	27	28	29	30

Пояснение. Элементы списков определяются как

```
struct NODE{ double x, y; struct NODE *next; }
struct NODE{ double x, y, z; struct NODE *next; }
struct NODE{ char *s; struct NODE *next; }
struct NODE{ char c; int x, y; struct NODE *next; }
struct NODE{ long x; struct NODE *next; }
```

(Для двухсвязного циклического списка и дерева поиска определено два указателя.)

Задание 2. Разработать алгоритм (согласно варианту) с использованием списков, написать и протестировать программу на языке Си++.

Варианты заданий

1. Задан текстовый файл. Вычислить частоты повторения слов, с помощью двоичного дерева, информационная часть которого состоит из слова и частоты.
2. Организовать двоичное дерево поиска, состоящее из целых чисел. Вывести содержимое его узлов, обходя это дерево в ширину.
3. Применить стек для обхода вершин графа, заданного с помощью матрицы смежности, в ширину.
4. Применить очередь для преобразования рекурсивной подпрограммы обхода дерева поиска в прямом порядке.
5. Написать программу вычисления значения логического выражения, состоящего из натуральных чисел, скобок и поразрядных операций ‘–’ (“не”), ‘&’ (“и”), ‘|’ (“или”). Приоритет у ‘–’ самый высокий, далее идет ‘&’. Самый низкий приоритет у операции ‘|’. Применить стеки для вычисления этого значения.
6. Решить задачу Джозефуса с помощью двухсвязного циклического списка.
7. Организовать стек, написать подпрограммы конкатенации (присоединения) одного стека к другому и обращения стека.
8. Организовать двухсвязный циклический список и написать подпрограмму обращения этого списка.
9. Применить стек для преобразования рекурсивной подпрограммы обхода дерева поиска в прямом порядке.
10. Написать программу для вычисления введенного с клавиатуры логического выражения, состоящего из операций & (“и”), | (“или”), > (“больше”), < (“меньше”), = (“равно”), скобок и целых чисел. Приоритет операции “и” равен 3, приоритет “или” – двум, приоритеты остальных операций равны 1. Значения логического выражения равны 1 (“истина”) или 0 (“ложь”).
11. Число задается в виде стека составляющих его цифр. Написать подпрограмму сложения двух чисел.
12. Организовать двухсвязный циклический список. Написать подпрограмму объединения двух таких списков.
13. Применить стек для преобразования рекурсивной подпрограммы обхода дерева в симметричном порядке.
14. Применить дерево поиска для исключения одинаковых чисел из массива и расположения их в возрастающем порядке.
15. Написать программу обхода вершин графа в ширину. Граф задан с помощью матрицы смежности.
16. Вычислить значение заданного с клавиатуры арифметического выражения, состоящего из целых чисел, скобок и операций ‘*’

(“умножение”), ‘:’ (“деление”), ‘%’ (“остаток от деления”), ‘+’ (“сложение”), ‘–’ (“вычитание”). Приоритеты первых трех операций равны 2, остальных –1.

17. Написать программу обхода дерева поиска в глубину.

18. Организовать линейный динамический список чисел, расположенных в монотонно неубывающем порядке. Написать подпрограммы добавления в конец списка, удаления первого элемента и вывода всех элементов на экран.

19. Написать программу конкатенации (присоединения) двух деков.

20. Задан текстовый файл, содержащий записанные через пробелы целые числа. Используя стек, вывести сначала в прямом порядке неотрицательные, а затем в обратном порядке отрицательные числа.

21. Задан текстовый файл. Проверить его на соответствие угловых и квадратных скобок.

22. Написать подпрограмму обращения очереди целых чисел.

23. Написать подпрограмму обращения односвязного циклического списка.

24. Организовать множество целых чисел как дерево поиска. Написать подпрограмму объединения двух множеств.

25. Задан текстовый файл. Проверить его на соответствие фигурных и угловых скобок.

26. Применить очередь для генерации перестановок с помощью преобразования рекурсивной подпрограммы.

27. С помощью стека решить уравнение $f(x) = 0$ методом деления отрезка пополам. Предполагается, что функция f непрерывна на отрезке $[a, b]$, и значения $f(a)$ и $f(b)$ имеют противоположные знаки.

28. Определить многоугольник как циклический однонаправленный список. Написать подпрограмму его вывода на экран.

29. Применить очередь для преобразования рекурсивной подпрограммы обхода дерева в обратном порядке.

30. Задан текстовый файл, содержащий записанные через пробелы целые числа. Используя стек, вывести сначала все четные числа, а затем – нечетные.

Седьмой семестр

Лабораторная работа 1

«Методы перебора слов, функций и последовательностей чисел»

Задание. Написать программу перебора слов, функций или чисел, удовлетворяющих заданным условиям. Каждое слово, функция или последовательность чисел выводится с новой строки. Числа m и n вводятся с клавиатуры.

Варианты заданий

1. Перебрать все функции $f: \{0, 1, \dots, m-1\} \rightarrow \{0, 1, \dots, n-1\}$.

2. Перебрать все сюръекции $f: \{0, 1, \dots, m-1\} \rightarrow \{0, 1, \dots, n-1\}$.
3. Перебрать все инъекции $f: \{0, 1, \dots, m-1\} \rightarrow \{0, 1, \dots, n-1\}$.
4. Перебрать все монотонно неубывающие функции $f: [m] \rightarrow [n]$.
5. Перебрать все монотонно возрастающие функции $f: [m] \rightarrow [n]$.
6. Перебрать все монотонно неубывающие сюръекции $f: [m] \rightarrow [n]$.
7. Перебрать все монотонно невозрастающие функции $f: [m] \rightarrow [n]$.
8. Перебрать все монотонно неубывающие функции $f: [m] \rightarrow [n]$.
9. Перебрать все монотонно невозрастающие сюръекции $f: [m] \rightarrow [n]$.
10. Перебрать все слова длиной не более n , составленные из букв "a" и "b".
11. Найти слово длины n , составленное из букв "a", "b", "c" и "d", не содержащие смежных идентичных (равных) подтекстов.
12. Найти слово длины n , составленное из букв "a", "b", и "c", не содержащие смежных инверсно равных подтекстов.
13. Перебрать все слова длины n , составленные из букв "a", "b" и "c", не содержащие смежных инверсно равных подтекстов.
14. Вывести на экран все десятичные числа, состоящие не более чем из n ненулевых цифр, у которых стоящие рядом цифры отличаются не более чем на 1.
15. Вывести на экран все последовательности натуральных чисел $(x_1, x_2, \dots, x_{2n})$, таких, что $0 \leq x_i \leq m$, и $x_1 \leq x_2 \geq x_3 \leq \dots \geq x_{2k-1} \leq x_{2k} \geq x_{2k+1} \leq \dots \geq x_{2n-1} \leq x_{2n}$.
16. Вывести на экран все последовательности натуральных чисел $(x_1, x_2, \dots, x_n)$, таких, что $0 \leq x_i \leq m$, и $x_1 < x_2 > x_3 < \dots > x_{2k-1} < x_{2k} > x_{2k+1} < \dots > x_{2n-1} > x_{2n}$.
17. Организовать перебор последовательностей $x[1], x[2], \dots, x[n]$, удовлетворяющих условиям $1 \leq x[k] \leq k$ и $x[x[k]] = x[k]$ для всех $k=1, 2, 3, \dots, n$. Вывести результаты при $n=4$ и при $n=5$.
18. Организовать перебор последовательностей $x_1 > \dots > x_m$ таких, что $n \geq x_1$ и $x_m \geq 0$.
19. Организовать перебор последовательностей целых чисел $n \geq x_1 \geq \dots \geq x_m \geq 0$.
20. Организовать перебор последовательностей целых чисел $0 \leq x_1 \leq \dots \leq x_m \leq n$.
21. Перебрать последовательности неотрицательных целых чисел $x_1, x_2, \dots, x_m$, сумма квадратов которых $x_1^2 + x_2^2 + \dots + x_m^2$ равна n .
22. Вывести на экран десятичные числа, состоящие из последовательностей n монотонно неубывающих цифр.
23. Вывести на экран все десятичные числа, состоящие из строго возрастающих последовательностей не равных между собой цифр.
24. Вывести на экран все десятичные числа, состоящие из строго убывающих последовательностей не равных между собой цифр.

25. Вывести на экран десятичные числа, состоящие из последовательностей не более чем n невозрастающих цифр.

26. Найти сумму $\sum_{i_1=1}^m \sum_{i_2=1}^m \dots \sum_{i_n=1}^m \frac{1}{i_1 \cdot i_2 \cdot \dots \cdot i_n}$.

27. Найти произведение $\prod_{i_1=1}^m \prod_{i_2=1}^m \dots \prod_{i_n=1}^m (1 + \frac{1}{i_1 + i_2 + \dots + i_n})$.

28. Найти сумму $\sum_{i_1=1}^m \sum_{i_2=1}^m \dots \sum_{i_n=1}^m \frac{1}{i_1^2 + i_2^2 + \dots + i_n^2}$.

29. Найти сумму дробей $\frac{1}{i_1 + i_2 + \dots + i_m}$ по всем подмножествам $\{i_1, i_2, \dots, i_m\}$ множества $\{1, 2, \dots, n\}$.

30. Найти произведение $\prod_{i_1=1}^m \prod_{i_2=1}^m \dots \prod_{i_n=1}^m (1 + \frac{1}{i_1^2 + i_2^2 + \dots + i_n^2})$.

Пояснения к заданию . Функцией $f: \{a_1, \dots, a_m\} \rightarrow \{b_1, \dots, b_n\}$ называется такое множество пар, для которого выполнены следующие условия:

- Для каждого $x \in \{a_1, \dots, a_m\}$ существует такой $y \in \{b_1, \dots, b_n\}$, что $(x, y) \in f$; иными словами, область определения отношения f равна всему множеству $\{a_1, a_2, \dots, a_m\}$;

- Если $(x, y_1) \in f$ и $(x, y_2) \in f$, то $y_1 = y_2$; для произвольной функции $f: \{a_1, \dots, a_m\} \rightarrow \{b_1, \dots, b_n\}$ и элемента $x \in \{a_1, \dots, a_m\}$ существует единственный элемент $y \in \{b_1, \dots, b_n\}$, для которого $(x, y) \in f$. Обозначим этот элемент через $f(x)$.

Функция $f: \{a_1, \dots, a_m\} \rightarrow \{b_1, \dots, b_n\}$ называется *сюръективной* (сюръекцией), если для каждого $y \in \{b_1, \dots, b_n\}$ существует $x \in \{a_1, \dots, a_m\}$, такой, что $y = f(x)$. Очевидно, что при $m < n$ таких функций не существует. Функция $f: \{a_1, \dots, a_m\} \rightarrow \{b_1, \dots, b_n\}$ называется *инъективной* (или инъекцией), если верна импликация $x_1 \neq x_2 \Rightarrow f(x_1) \neq f(x_2)$.

Обозначим через $[k]$ множество $\{0, 1, \dots, k\}$. Функция $f: \{0, 1, \dots, m\} \rightarrow \{0, 1, \dots, n\}$ называется монотонно неубывающей, если $f(0) \leq f(1) \leq \dots \leq f(m)$,

- монотонно невозрастающей, если $f(0) \geq f(1) \geq \dots \geq f(m)$,
- убывающей, если $f(0) > f(1) > \dots > f(m)$,
- возрастающей, если $f(0) < f(1) < \dots < f(m)$.

Словом называется произвольная последовательность букв. *Длиной* слова $x_1 x_2 \dots x_n$, составленного из букв x_i , называется число n . Пустое слово имеет длину 0. *Подсловом*, или подтекстом слова $x_1 x_2 \dots x_n$, называется слово, равное для некоторых i и j , $1 \leq i \leq j \leq n$, слову $x_i x_{i+1} \dots x_j$. Подтексты $x_i x_{i+1} \dots x_j$ и $x_{j+1} x_{j+2} \dots x_k$ называются *смежными*. Два слова $x_1 x_2 \dots x_m$ и $y_1 y_2 \dots y_n$ называются *идентичными* или *равными*, если $m = n$ и для всех $1 \leq i \leq m$ и имеют место равенства $x_i = y_i$. Два слова $x_1 x_2 \dots x_m$ и $y_1 y_2 \dots y_n$ называются *инверсно равными*, если $m = n$ и справедливы равенства $y_1 = x_n, x_2 = y_{n-1}, \dots, x_i = y_{n-i+1}, \dots, x_n = y_1$.

Отметим, что в варианте 23 максимально возможное количество цифр в числах равно 9, а в варианте 24 максимально возможное количество – 10.

Лабораторная работа 2

«Реализация метода перебора с возвратом»

Задание. Разработать алгоритм (согласно варианту) с помощью алгоритма перебора с возвратом, написать и протестировать программу на языке Си++ с использованием рекурсивных подпрограмм.

Варианты заданий

1. Вывести все расположения 8 слонов на шахматной доске, не угрожающих друг другу и расположенных на различных вертикалях.

2. С помощью нерекурсивного перебора с возвратом найти гамильтонов цикл в графе, заданном с помощью матрицы смежности.

3. Сверхферзем называется фигура, которая умеет ходить и как ферзь, и как конь. Какое максимальное количество не угрожающих друг другу сверхферзей можно расставить на шахматной доске?

4. Решить задачу коммивояжера методом перебора с возвратом. Цены билетов заданы с помощью симметричной $n \times n$ матрицы.

5. Решить задачу Гаусса о ферзях с помощью нерекурсивного перебора с возвратом.

6. Перебрать все пути в графе между двумя заданными вершинами, не содержащие одинаковых вершин.

7. Найти путь наименьшей длины между двумя заданными вершинами с помощью нерекурсивного перебора с возвратом.

8. Задано множество $A = \{a_1, a_2, \dots, a_n\}$, натуральные числа $s(a_j) > 0$, и $k > 0$.

Найти разбиение $\bigcup_{j=1}^k A_j = A$ множества A на попарно непересекающиеся

подмножества, при котором $\sum_{j=1}^k (\sum_{a \in A_j} s(a))^2$ минимальна.

9. Кликкой называется множество вершин графа, любые две из которых смежны. Найти клику, имеющую максимальное количество вершин.

10. Задача о рюкзаке. Имеются предметы, веса которых равны $w_1, w_2, \dots, w_n$, а цены которых равны $c_1, c_2, \dots, c_n$. Выбрать из них предметы, суммарный вес которых меньше 20 кг, а стоимость – максимальна.

11. Найти раскраску вершин графа минимальным числом цветов так, что ни одно ребро не соединяло две вершины одного цвета.

12. Найти все пути шахматного коня между двумя заданными полями, не содержащие повторяющихся полей.

13. Перебрать разбиения множества $\{1, 2, \dots, n\}$ с помощью нерекурсивной реализации алгоритма перебора с возвратом.

14. Дана $m \times n$ матрица с неотрицательными коэффициентами $a_{ij} > 0$. Найти такое множество пар $\{(i_1, j_1), (i_2, j_2), \dots, (i_k, j_k)\}$, что $i_s \neq i_t$ и $j_s \neq j_t$ для всех $s \neq t$, для которого сумма $a_{i_1 j_1} + a_{i_2 j_2} + \dots + a_{i_k j_k}$ максимальна.

15. Решить задачу Эйлера о шахматном коне с помощью нерекурсивной реализации алгоритма перебора с возвратом.

16. Пегасом (рисунок 1) называется шахматная фигура, ходы которой от ходов шахматного коня отличаются размерами буквы «Г». Шахматный пегас, стоящий на поле (i,j) может сделать ход на одно из восьми полей: $(i+4,j+3)$, $(i+3,j+4)$, $(i-3,j+4)$, $(i-4,j+3)$, $(i-4,j-3)$, $(i-3,j-4)$, $(i+3,j-4)$, $(i+4,j-3)$. Найти путь пегаса, состоящий из всех полей шахматной доски, и не содержащий ни одного поля более чем один раз. (Аналогично задаче Эйлера.)

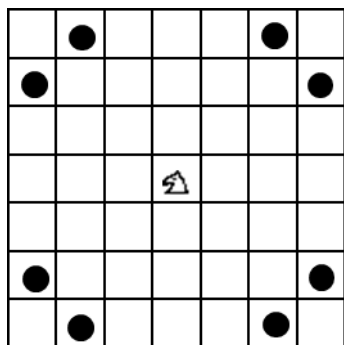


Рисунок 1. Шахматный пегас

17. Написать программу перебора всех простых циклов в графе, выходящих из заданной вершины и имеющих заданную длину k .

18. Пусть $[m] \times [n]$ – декартово произведение линейно упорядоченных множеств $\{0,1,\dots,m\}$ и $\{0,1,\dots,n\}$, с отношением порядка $(i,j) \leq (i',j') \Leftrightarrow (i \leq i' \text{ и } j \leq j')$. Организовать перебор последовательностей $(i_1, j_1) < (i_2, j_2) < \dots < (i_k, j_k)$ при заданном k .

19. В центре шахматной доски стоит конь. Найти все расположения восьми ладей, не угрожающих друг другу, при которых конь не угрожает ни одной из ладей.

20. Дано натуральное число m . Вставить между цифрами 1,2,3,4,5,6,7,8,9, записанными именно в таком порядке, знаки "+" и "-", так, чтобы значением получившегося выражения было число m . Например, если $m=118$, то подойдет расстановка $123+45-67+8+9$. Вывести сообщение, если таких расстановок нет.

21. Организовать перебор всех клик заданной мощности k в простом графе.

22. Найти все гамильтоновы циклы в графе, заданном с помощью матрицы смежности.

23. Найти не содержащие кратных вершин пути в графе, не имеющие продолжения и выходящие из заданной вершины. Граф задан с помощью матрицы смежности.

24. Найти циклы шахматного коня, не содержащие кратных вершин и проходящие через заданную позицию. Рассматриваются циклы длины 2,4,6,8 и 10.

25. Найти путь максимальной длины в графе, не имеющий кратных вершин.
26. Задан ориентированный простой граф. Найти все ориентированные пути между двумя заданными вершинами, не имеющие кратных вершин.
27. Найти все простые циклы в ориентированном графе, не содержащие кратных вершин и проходящие через заданную вершину.
28. Найти путь в ориентированном графе между двумя заданными вершинами, не содержащий кратных вершин и имеющий максимальную стоимость.
29. Найти гамильтонов цикл в простом графе, имеющий максимальную длину.
30. Найти кратчайший путь между двумя заданными вершинами графа, предполагая, что допустимы отрицательные длины для ребер.

Расчетно-графическая работа «Алгоритмы сортировки»

Задание. Написать программу, которая сортирует данные, а затем сравнивает два указанных в варианте задания метода сортировки по

- 1) времени выполнения;
- 2) числу сравнений;
- 3) числу перестановок.

Варианты заданий приведены в таблице 8.

Таблица 8 – Варианты заданий

	Метод пу- зырь- ков	Метод выбо- ра	Дво- ичные встав- ки	Метод Шел- ла	Быст- рая Хоара	Обмен- ная по- разряд- ная	Турнир с вы- быва- нием	Пира- ми- даль- ная
Простые вставки	1	2	3	4	5	6	7	8
Метод пузырьков	-	9	10	11	12	13	14	15
Метод выбора	-	-	16	17	18	19	20	21
Двоичные вставки	-	-	-	22	23	24	25	26
Быстрая Хоара	-	-	-	-	-	27	28	29
Обменная поразряд- ная	-	-	-	-	-	-	30	-

Примерная структура билетов тестирования

Тест «Разработка рекурсивных подпрограмм»

Вопрос 1. Расположить этапы разработки алгоритма в порядке их прохождения:

1. постановка задачи;
2. построение модели;
3. построение блок-схемы алгоритма;
4. проверка правильности;
5. программная реализация;
6. анализ алгоритма и его сложности;
7. проверка программы;
8. документирование.

Вопрос 2. Подпрограмма называется рекурсивной, если

Варианты ответа:

1. она, по крайней мере один раз, вызывает себя;
2. она реализует примитивно рекурсивную функцию;
3. она содержит вложенные циклы;
4. она не имеет аргументов.

Вопрос 3. Установить соответствие между *задачами* и *методами* их решения

- | | |
|---------------------------------------|----------------------------------|
| 1. Задача о Ханойских башнях | 1. Декомпозиция |
| 2. Кривые Гильберта | 2. Косвенная рекурсия |
| 3. Синтаксический анализ и вычисление | 3. Рекурсивный спуск |
| 4. Построение кратчайшего пути | 4. Динамическое программирование |

Вопрос 4. Рассмотрим числа Фибоначчи: 0,1,1,2,3,5,8,13,21... Указать, какая из представленных ниже рекурсивных функций, является рекурсивной функцией для вычисления этих чисел.

Варианты ответа:

1.
`Int F(int n)
{
if(n==0) return 0;
if(n==1) return 1;
return F(n-1) + F(n-2);
}`

2.
`Int F(int n)
{
if(n==0) return 1;
if(n==1) return 1;
return F(n-1) + F(n-2);
}`

3.
`Int F(int n)
{
if(n==0) return 0;`


```

if(n==1) return 1;
return F(n-1) + 2*F(n-2);
}

```

Тест «Метод динамического программирования. Алгоритмы на графах»

Вопрос 1. Рассмотрим числа Фибоначчи: 0,1,1,2,3,5,8,13,21... Указать, в каком из представленных ниже фрагментов программы, вычисляются эти числа методом динамического программирования, а их значения заносятся в массив F.

Варианты ответа:

1.

```

F[0]=0; F[1]=1;
for(i=2; i<=n; i++)
    F[i]=F[i-1]+ F[i-2];

```

2.

```

Int F(int n)
{
    if(n==0) return 0;
    if(n==1) return 1;
    return F(n-1) + F(n-2);
}

```

3.

```

F[0]=1; F[1]=1;
for(i=2; i<=n; i++)
    F[i]=F[i-1]+ F[i-2];

```

Вопрос 2. При нахождении пути между вершинами A и B последовательность номеров вершин выстраивается в обратном порядке, первая вершина равна B, а вторая - A. Как превратить этот порядок обхода в прямой, от A к B?

Варианты ответа:

1. с помощью рекурсивной подпрограммы;
2. методом декомпозиции;
3. с помощью циклов;
4. с помощью двойного цикла.

Вопрос 3. На рисунке 2 приведен пример графа. Используя алгоритм Дейкстры, построить кратчайший путь между вершинами 1 и 4.

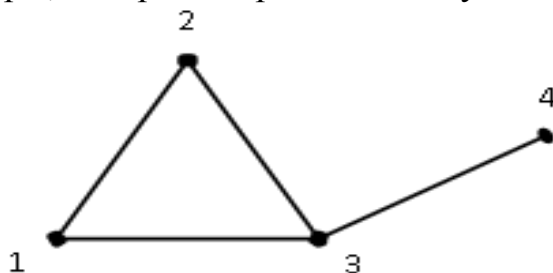


Рисунок 2. Пример графа

Ответ.....

Вопрос 4. Используя метод обхода в ширину, указать компоненту связности для графа, приведенного на рисунке 2.

Ответ.....

Тест «Динамические структуры данных и их применение»

Вопрос 1. Списком называется

Варианты ответа:

1. набор переменных одного типа вместе с функциями добавления и удаления элементов;
2. массив данных вместе с функциями добавления, удаления и поиска элементов;
3. набор переменных одного типа;
4. линейно упорядоченный набор переменных одного типа вместе с функциями добавления и удаления элементов.

Вопрос 2. Установить соответствие между *типом списка*

1. стек
2. очередь
3. дек
4. двоичное дерево поиска

и *свойствами его функций* добавления и удаления элементов

1. добавление и удаление элементов производится с одной стороны списка;
2. добавление и удаление элемента производятся с разных сторон;
3. добавление и удаление возможно как с начала, так и с конца списка;
4. место добавляемого элемента зависит от его значения.

Вопрос 3. Установить соответствие между *задачами*

1. обход вершин графа в глубину;
2. обход вершин графа в ширину;
3. считалка Джозефуса;
4. задача нахождения частоты слов в текстовом файле.

и *типами списков*, применяемых для решения этих задач

1. стек;
2. очередь;
3. циклический список;
4. двоичное дерево поиска.

Вопрос 4. Какой структуры данных из перечисленных ниже не существует

Варианты ответа:

1. циклический список;
2. односвязный циклический список;
3. двухсвязный циклический список;
4. трехсвязный циклический список.

Тест «Методы перебора»

Вопрос 1. Метод перебора с возвратом применяется

Варианты ответа:

1. для построения конечных последовательностей элементов, связанных заданными отношениями;
2. для поиска элемента в упорядоченном массиве;
3. для сортировки чисел;
4. для перебора вершин направленного графа.

Вопрос 2. Какому условию должна удовлетворять последовательность отношений $P_n(x_1, \dots, x_n)$, позволяющих использовать алгоритм перебора с возвратом?

Варианты ответа:

1. $P_n(x_1, \dots, x_n) \Rightarrow P_{n-1}(x_1, \dots, x_{n-1})$;
2. $P_n(x_1, \dots, x_n) \Rightarrow P_{n+1}(x_1, \dots, x_{n+1})$;
3. $P_n(x_1, \dots, x_n) = P_1(x_1) \& \dots \& P_1(x_n)$;
4. $P_n(x_1, \dots, x_n) = P_n(x_n, \dots, x_1)$.

Вопрос 3. Указать, какие из приведенных ниже слов, являются словами, не содержащими смежных идентичных подтекстов:

1. ***abacab***;
2. ***acabab***;
3. ***acacab***;
4. ***acbaac***.

Вопрос 4. Установить соответствие между *задачами*, решаемыми методом перебора с возвратом

5. задача Гаусса
6. задача Эйлера
7. поиск гамильтоновых циклов
8. генерация разбиений конечного множества

и их *формулировками*:

1. перебор расположений восьми шахматных ферзей, не угрожающих друг другу;
2. нахождение маршрута шахматного коня, содержащего все поля шахматной доски по одному разу каждое поле;
3. нахождение цикла в графе, содержащего все вершины по одному разу каждая вершина;
4. перебор разбиений конечного множества на попарно непересекающиеся подмножества.

Тест «Алгоритмы сортировки»

Вопрос 1. Установить соответствие между группами *методов* сортировки

1. метод вставок
2. метод обменов
3. посредством выбора
4. метод подсчета

и алгоритмами:

1. сортировка Шелла;
2. метод пузырьков;
3. турнир с выбыванием;
4. распределяющий подсчет.

Вопрос 2. Какой из перечисленных ниже алгоритмов не относится к группе алгоритмов сортировок посредством выбора

Варианты ответа:

1. турнир с выбыванием;
2. пирамидальная сортировка;
3. быстрая сортировка Хоара.

Вопрос 3. Установить соответствие между *алгоритмами* сортировки

1. метод двоичных вставок
2. двухпутевые вставки
3. вставки в список
4. вставки в дерево

и их *описанием*:

1. при добавлении элемента из входного массива в выходном массиве производится поиск места очередного элемента с помощью двоичного поиска;
2. первый элемент входного массива записывается в массив длины $2n-1$ элементов, места для последующих элементов освобождаются с помощью сдвигов влево и вправо;
3. входные данные из массива записываются в упорядоченный список, который будет содержать нужную отсортированную последовательность;
4. входные данные из массива записываются в двоичное дерево поиска, обход полученного дерева дает отсортированный массив.

Вопрос 4. В каком методе сортировки используются два файла, причем данные считываются с начала и с конца первого файла и до тех пор, пока сохраняется монотонность сливаются то в начало, то в конец второго файла, а потом роли файлов меняются, пока не появится пустой файл. Как называется этот алгоритм?

Варианты ответа:

1. естественное двухпутевое слияние;
2. простое двухпутевое слияние;
3. двухленточная сортировка на основе поразрядной;
4. четырехленточная сортировка слиянием.

Задания для промежуточной аттестации

Задания на курсовую работу

Тема курсовой работы: «Программирование игр с выигрышной стратегией».

Заданием к данной курсовой работе является проектирование, разработка и тестирование ПО для реализации одной из игр с выигрышной стратегией. Студентам предоставляется право самостоятельно выбрать индивидуальное задание.

Примерные задания

1. Игра «Морской бой».
2. Игра «Крестики-нолики».
3. Игра «Спички».
4. Игра дат.
5. Игра «24 карты».
6. Игра города Нима.
7. Игра Норкотта.
8. Игра Кейлеса.
9. Игра Сима.
10. Игра «Спички Бергсона».
11. Игра «Гениальный отладчик».
12. Игра «Погоня за сокровищами».
13. Игра «Могущественная четверка».
14. Игра Шеннона.
15. Игра «Гекс».

Первоначально студент разрабатывает алгоритм, выполняющий расчет выигрышной стратегии и определяется с выбором структур данных. Этот алгоритм может быть рекурсивным, а может опираться на алгоритмы обхода вершин графа в глубину и в ширину. Затем студент определяется с выбором структур данных и проектирует, разрабатывает, тестирует ПО, реализующее игровой процесс. Разработанное программное обеспечение должно обеспечить решение следующих задач:

- реализацию графического интерфейса;
- реализацию управления ходом игры с помощью клавиатуры и мыши;
- реализацию игрового процесса.

Компьютерная игра разрабатывается на языке программирования высокого уровня C++, в процессе ее проектирования и отладки у студентов формируются практические навыки организации данных в разрабатываемом ПО и использования типовых линейных и нелинейных структур данных, включая двоичные деревья поиска.

Пояснительная записка к КР, согласно СТО У.003-2017, имеет следующую структуру:

- титульный лист;

- задание на КР;
- содержание;
- введение;
- основная часть;
- заключение;
- список использованных источников;
- приложения.

Во введении дается краткое обоснование выбора темы, раскрывается актуальность, новизна, теоретическая и практическая значимость работы и приводится программа исследования.

Обязательными подразделами основной части являются «Техническое задание», «Текст программы», «Описание программы», «Руководство программиста», «Руководство оператора», «Программа и методика испытаний» (таблица 9).

Таблица 9 – Содержание подразделов основной части курсовой работы

Вид программного документа	Содержание программного документа
Техническое задание	Приводятся основание и назначение разработки, требования к ПО, программной документации, стадии и этапы разработки
Текст программы	Запись программы с необходимыми комментариями
Описание программы	Сведения о логической структуре и функционировании программы
Руководство программиста	Назначение и условия применения программы, характеристика программы, обращение к программе, входные и выходные данные, сообщения
Руководство оператора	Назначение программы, условия выполнения программы, выполнение программы, сообщения оператору
Программа и методика испытаний	Требования, подлежащие проверке при испытании программы, а также порядок и методы их контроля

В заключении КР приводятся общие выводы и предложения, а также краткое описание проделанной работы. Заключение должно носить конкретный характер и показывать, что именно сделал студент в рамках выполнения КР. Список использованных источников состоит из нормативно-правовых актов, учебников и учебных пособий, научных статей, нормативно-правовых актов, использованных в ходе выполнения индивидуального задания.

Контрольные вопросы к экзамену

1. Проблема построения последовательностей, удовлетворяющих заданным отношениям, и ее решение методом перебора с возвратом.
2. Алгоритм перебора с возвратом как метод раскраски.
3. Рекурсивная реализация алгоритма перебора с возвратом.
4. Задача Гаусса о ферзях.

5. Алгоритм перебора гамильтоновых циклов в графе.
6. Задача Эйлера о шахматном коне и алгоритм ее решения.
7. Алгоритм решения задачи о размене денег.
8. Алгоритмы перебора подмножеств конечного множества.
9. Алгоритм перебора разложений числа в сумму.
10. Алгоритм перебора монотонно неубывающих последовательностей.
11. Алгоритм генерации разбиений множества.
12. Организация переменного числа вложенных циклов.
13. Недетерминированный алгоритм распознавания и класс NP-полных задач.
14. Шесть основных NP-полных задач.
15. Поиск в отсортированном массиве.
16. Внутренние и внешние методы сортировки.
17. Метод простых вставок.
18. Метод пузырьков.
19. Сортировка методом выбора.
20. Сортировка подсчетом.
21. Распределяющий подсчет.
22. Модификации метода вставок.
23. Сортировка методом Шелла.
24. Быстрая сортировка Хоара.
25. Обменная поразрядная сортировка.
26. Сортировочные сети и алгоритм Бетчера.
27. Турнир с выбыванием.
28. Пирамидальная сортировка.
29. Очередь с приоритетами.
30. Алгоритм двухпутевого слияния.
31. Сортировка простым двухпутевым слиянием.
32. Естественное двухпутевое слияние.
33. Двухленточная сортировка на основе поразрядной.
34. Трехленточная сортировка слиянием.
35. Четырехленточная сортировка слиянием.

Типовые экзаменационные задачи

1. Написать программу нахождения слов длины n , состоящих из букв "А", "В", "С" и не содержащих смежных идентичных подтекстов.
2. Написать программу перебора монотонно возрастающих функций $\{0, 1, 2, \dots, m\} \rightarrow \{0, 1, 2, \dots, n\}$.
3. Написать программу перебора последовательностей *чисел* $0 \leq x_0 < x_1 < \dots < x_m \leq n$ с помощью рекурсивной реализации метода перебора с возвратом.

4. Написать программу нахождения маршрута коня, обходящего все поля доски по одному разу и возвращающегося на исходное поле.
5. Написать программу перебора всех разменов заданной суммы денег.
6. Написать программу перебора всех подмножеств заданного конечного множества, при котором очередное множество получается из предшествующего либо добавлением, либо удалением одного элемента.
7. Написать подпрограмму, зависящую от m и n и вычисляющую сумму $f(i_1, i_2, \dots, i_m)$, где i_k пробегает все значения от 0 до $m-1$. Функция f предполагается внешней заданной функцией.
8. Для заданных входных данных показать, как работает алгоритм сортировки методом простых вставок.
9. Для заданных входных данных продемонстрировать метод простых пузырьков.
10. Для заданного массива чисел продемонстрировать сортировку методом выбора.
11. Для заданного массива чисел продемонстрировать алгоритм сортировки подсчетом.
12. Для заданного массива чисел продемонстрировать алгоритм сортировки распределяющим подсчетом.
13. Для заданных входных данных продемонстрировать сортировку методом двухпутевых вставок.
14. Написать программу сортировки методом двухпутевых вставок.
15. Написать программу сортировки методом вставок в список.
16. Для заданных входных данных продемонстрировать алгоритм Шелла.
17. Для заданных входных данных продемонстрировать быструю сортировку Хоара.
18. Для заданных входных данных продемонстрировать обменную поразрядную сортировку.
19. Нарисовать сортировочную сеть для алгоритма Бетчера сортировки восьми чисел.
20. Для заданных входных данных продемонстрировать турнирную сортировку.
21. Для заданных входных данных продемонстрировать пирамидальную сортировку.
22. Задан файл, состоящий из 10 000 чисел. Вывести на экран наименьшие 20 чисел из этого файла.
23. Для заданных входных данных продемонстрировать сортировку простым двухпутевым слиянием.
24. Для заданных входных данных продемонстрировать сортировку естественным двухпутевым слиянием.
25. Для заданных входных данных продемонстрировать двухленточную сортировку на основе поразрядной.

26. Для заданных входных данных продемонстрировать трехленточную сортировку слиянием.

27. Для заданных входных данных продемонстрировать четырехленточную сортировку слиянием.

Пример экзаменационного билета

1. Задача Эйлера о шахматном коне и алгоритм ее решения.
2. Сортировка подсчетом. Распределяющий подсчет.
3. Написать программу перебора монотонно неубывающих отображений между множествами $\{0,1,2, \dots, m\}$ и $\{0,1,2, \dots, n\}$.

8 Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)

8.1 Основная литература

1 Белов, В. В. Алгоритмы и структуры данных [Электронный ресурс]: учебник / В.В.Белов, В.И.Чистякова. - М.: КУРС, НИЦ ИНФРА-М, 2017. - 240 с. // ZNANIUM.COM : электронно-библиотечная система. – Режим доступа: <http://www.znanium.com/catalog.php?>.

2 Канцедаль, С. А. Алгоритмизация и программирование [Электронный ресурс] : учебное пособие / С.А. Канцедаль. – М.: ИД ФОРУМ: НИЦ ИНФРА-М, 2014. – 352 с. // ZNANIUM.COM : электронно-библиотечная система. – Режим доступа: <http://www.znanium.com/catalog.php?>.

3 Колдаев, В. Д. Основы алгоритмизации и программирования [Электронный ресурс]: учебное пособие / В.Д. Колдаев; под ред. Л.Г. Гагариной. – М.: ИД ФОРУМ: ИНФРА-М, 2015. – 416 с. // ZNANIUM.COM : электронно-библиотечная система. – Режим доступа: <http://www.znanium.com/catalog.php?>.

4 Программная инженерия: учебник для вузов / под ред. Б.Г.Трусова. – М.: Академия, 2014. – 282с. – (Высшее образование. Бакалавриат). – Библиогр.: с.273–280.

8.2 Дополнительная литература

1 Баженова, И.Ю. Языки программирования: учебник для вузов / И. Ю. Баженова; под ред. В.А.Сухомлинова. - М.: Академия, 2012. – 358с. - (Высшее профессиональное образование. Бакалавриат).

2 Кузин, А. В. Программирование на языке Си [Электронный ресурс] /А.В.Кузин, Е.В.Чумакова – М.: Форум, НИЦ ИНФРА-М, 2015. – 144с.// ZNANIUM.COM : электронно-библиотечная система. – Режим доступа: <http://www.znanium.com/catalog.php?>.

3 Павловская, Т.А. C/C++. Программирование на языке высокого уровня: учебник для вузов / Т. А. Павловская. – СПб.: Питер, 2010. – 460с.

9 Перечень ресурсов информационно-телекоммуникационной сети «Интернет» (далее – сеть «Интернет»), необходимых для освоения дисциплины (модуля)

- 1 Библиотека программиста. Изучаем алгоритмы: полезные книги, веб-сайты, онлайн-курсы и видеоматериалы [http://www. proglib.io](http://www.proglib.io) .
- 2 Национальный открытый университет ИНТУИТ <http://www.intuit.ru>.
- 3 Единое окно доступа к образовательным ресурсам // Электронный ресурс [Режим доступа: свободный] <http://window.edu.ru/>.

10 Методические указания для обучающихся по освоению дисциплины (модуля)

Обучение дисциплине предполагает проведение аудиторных занятий и выполнение студентом самостоятельной работы. Аудиторные занятия проходят в форме лекций и лабораторных занятий.

Во время лекционных занятий при написании конспекта лекций студенту рекомендуется кратко, схематично, последовательно фиксировать основные положения, формулировки, выводы, отмечать важные мысли, выделять ключевые слова, термины, делать пометки на вопросах, терминах, блоках в тексте, которые вызывают затруднения, после чего постараться найти ответ в рекомендуемой литературе. Если ответ не найден, то обратиться к преподавателю.

Во время лабораторных занятий студенту рекомендуется работать с конспектом лекций, использовать интернет-ресурсы для построения алгоритмов выполнения заданий лабораторной работы или КР. Также студенту необходимы знания теории организации и структурирования данных на ЭВМ и типовых алгоритмов обработки данных. В случае затруднений, обратиться с вопросом к преподавателю.

Выполнение лабораторных работ и КР способствуют лучшему освоению практических навыков по данному предмету, закреплению и углублению навыков организации данных и использования стандартных алгоритмов.

В рамках выполнения самостоятельной работы студент готовится к лабораторным занятиям, изучает и повторяет отдельные теоретические разделы дисциплины, выполняет и оформляет лабораторные работы и КР, а также готовится к их защите.

Текущий контроль учебной деятельности студентов осуществляется на лабораторных занятиях при тестировании и защите студентом лабораторных работ. Проведение контроля текущей успеваемости, с одной стороны, позволяет получить достоверную информацию, как о степени освоения студентом теоретических разделов дисциплины, так и приобретения им практических навыков, с другой стороны, стимулирует ритмичность учебной деятельности.

На тестирование выносятся практические задания, соответствующие всем теоретическим разделам дисциплины.

Защита лабораторных работ проводится как в форме собеседования на лабораторном занятии, что позволяет определить уровень знаний студента основных понятий, алгоритмов и методов, так и в форме тестирования разработанного студентом ПО при различных входных данных и параметров, что позволяет оценить его умения организовывать данные в разрабатываемом ПО и выбирать алгоритмы обработки данных при решении практических задач, а также навыки использования типовых структур данных и стандартных алгоритмов при организации и обработке данных на ЭВМ.

11 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем (при необходимости)

Освоение дисциплины «Структуры данных и алгоритмы» основывается на активном использовании студентом Microsoft Visual C++, интегрированной среде разработки приложений на языке C++.

С целью повышения качества ведения образовательной деятельности в университете создана электронная информационно-образовательная среда. Она подразумевает организацию взаимодействия между обучающимися и преподавателями через систему личных кабинетов студентов, расположенных на официальном сайте университета в информационно-телекоммуникационной сети «Интернет» по адресу <https://student.knastu.ru>.

Созданная информационно-образовательная среда позволяет контролировать ход образовательного процесса посредством размещения студентами в личных кабинетах отчетов о выполненной КР, проверкой преподавателем КР, по результатам которой либо КР засчитывается, либо отправляется на доработку, но при этом преподаватель обязательно указывает конкретные замечания. Также информационно-образовательная среда предоставляет возможность преподавателю размещать объявления и информацию для студентов с указанием номеров студенческих групп.

В процессе подготовки отчетов студентом активно используется текстовый редактор Word.

При изучении дисциплины, для выполнения лабораторных работ и контрольной работы используется распространяемое бесплатно программное обеспечение – бесплатная полнофункциональная интегрированная среда разработки Visual Studio Community 2017. Visual Studio Community может использовать неограниченное число пользователей в организации в следующих случаях: в учебных аудиториях, для научных исследований или участия в проектах с открытым кодом.

12 Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Для реализации программы дисциплины «Структуры данных и алгоритмы» используется материально-техническое обеспечение, перечисленное в таблице 10.

Таблица 10 – Материально-техническое обеспечение дисциплины

Аудитория	Наименование аудитории (лаборатории)	Используемое оборудование	Назначение оборудования
303-3, 305-3, 312-3	компьютерные классы ФКТ	10 персональных ЭВМ, каждая из которых оснащена процессором Intel(R) Core (TM) i3-2100 CPU @3.10 GHz и оперативной памятью 2ГБ. Операционная система - Windows 7. В классе имеется сетевой коммутатор Cisco catalyst 2960 с ПО IOS ver 12.2(55)SE5.	Проведение лекционных и лабораторных занятий

Лист регистрации изменений к РПД

№ п/п	Содержание изменения/основание	Кол-во стр. РПД	Подпись автора РПД
1	Изменение листа подписей в связи со сменой декана ФКТ /пр.№ 271-ЛС «к» от 29.12.2016	1	
2	Изменение КУГ/пр. № 326-О «а» от 04.09.2017	8	
3	Изменение титульного листа в связи с переименованием вуза/пр. №997-О от 03.11.2017	1	
4	Актуализация литературы/ 20.11.2017	2	
5	Актуализация используемого ПО/ 28.11.2017	1	
6	Актуализация расписания дисциплины согласно учебного плана 2020-2021 года	6	