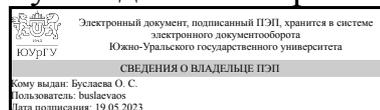


ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ:
Руководитель направления



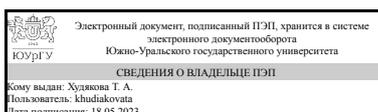
О. С. Буслаева

РАБОЧАЯ ПРОГРАММА

дисциплины 1.О.12 Структуры данных и прикладные алгоритмы
для направления 09.03.02 Информационные системы и технологии
уровень Бакалавриат
форма обучения очная
кафедра-разработчик Цифровая экономика и информационные технологии

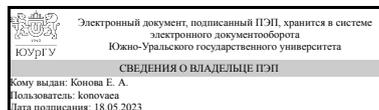
Рабочая программа составлена в соответствии с ФГОС ВО по направлению подготовки 09.03.02 Информационные системы и технологии, утверждённым приказом Минобрнауки от 19.09.2017 № 926

Зав.кафедрой разработчика,
Д.ЭКОН.Н., доц.



Т. А. Худякова

Разработчик программы,
старший преподаватель



Е. А. Конова

1. Цели и задачи дисциплины

Основная цель изучения дисциплины – изучение структур данных и прикладных алгоритмов, используемых в программировании задач различных прикладных областей, в том числе в экономике, и знакомство с библиотекой классов .Net Framework, реализующих основные структуры данных. Необходимой составляющей обучения является формирование практических навыков по разработке моделей прикладных задач на основе динамических данных. В результате успешного усвоения курса студент должен овладеть специальной подготовкой в предметной области и в области информационных технологий. Задача изучения дисциплины – реализация требований, установленных в квалификационной характеристике, а именно: 1) изучить динамические структуры данных; 2) получить навыки разработки динамических структур данных на основе объектного подхода; 3) изучить структуры данных библиотеки классов Net.Framework; 4) получить навыки проектирования практических задач на основе библиотек классов Net.Framework и опыт разработки Windows приложений. В результате изучения дисциплины формируется теоретическая и методологическая база для цикла общепрофессиональных дисциплин, в которых, так или иначе, используются динамические структуры данных.

Краткое содержание дисциплины

В дисциплине «Реализация структур данных в современных программных средах» изучаются динамические структуры данных, которые используются для эффективного хранения и реализации эффективных алгоритмов обработки данных. В дисциплине выделены следующие разделы: • динамические структуры данных: стек, очередь, список; • ассоциативные массивы; • деревья; • графы; • библиотека классов Net.Framework как инструмент разработки интерфейса приложений; • реализация динамических структур данных в библиотеке классов Net.Framework System.Collection.Generic; • практическая разработка динамических структур данных; • использование библиотек классов Net.Framework для реализации объектной модели приложений; • разработка прикладных задач на основе использования динамических данных. В качестве языка программирования используется язык C++/CLI, управляемый, один из базовых языков Net.Framework, в качестве среды разработчика используется Microsoft Visual Studio.

2. Компетенции обучающегося, формируемые в результате освоения дисциплины

Планируемые результаты освоения ОП ВО (компетенции)	Планируемые результаты обучения по дисциплине
ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий	Знает: особенности различных структур данных и применяемых к ним алгоритмов; принципы реализации алгоритмов обработки данных; основы разработки, тестирования и отладки программ Умеет: проводить анализ постановки задачи и выбирать оптимальные средства и методы решения задач; проектировать алгоритмическое решение на основе выбранной структуры

	<p>данных; использовать средства разработки и отладки современной интегрированной среды программирования</p> <p>Имеет практический опыт: эффективной реализации задач, требующих создания алгоритмов сложных структур данных; программирования, отладки и тестирования алгоритмов для решения практических задач</p>
<p>ОПК-8 Способен применять математические модели, методы и средства проектирования информационных и автоматизированных систем</p>	<p>Знает: показатели эффективности алгоритмов и способы их оценки; основные виды типовых алгоритмов, применяемых при разработке программного обеспечения; основные виды типовых структур данных, применяемых при разработке программного обеспечения и основные операции, производимые с ними.</p> <p>Умеет: разрабатывать типовые алгоритмы на языках высокого уровня; предлагать эффективные алгоритмы решения типовых задач сортировки, поиска, оптимизации; выбирать структуры данных при реализации программного обеспечения; оценивать показатели эффективности предложенных решений.</p> <p>Имеет практический опыт: использования языков процедурного и объектно-ориентированного программирования; разработки, тестирования и отладки программ в объектно-ориентированном и процедурном стилях; инструментальными средствами разработки программ.</p>

3. Место дисциплины в структуре ОП ВО

Перечень предшествующих дисциплин, видов работ учебного плана	Перечень последующих дисциплин, видов работ
<p>1.О.15.01 Основы программирования, 1.О.15.02 Программирование на языках высокого уровня</p>	<p>1.О.20 Моделирование информационных систем, 1.О.16 Базы данных, Учебная практика (научно-исследовательская работа, получение первичных навыков научно-исследовательской работы) (4 семестр)</p>

Требования к «входным» знаниям, умениям, навыкам студента, необходимым при освоении данной дисциплины и приобретенным в результате освоения предшествующих дисциплин:

Дисциплина	Требования
<p>1.О.15.01 Основы программирования</p>	<p>Знает: основные конструкции языка программирования высокого уровня, основные компоненты современной среды программирования, основные структуры данных и алгоритмы их обработки</p> <p>Умеет: проектировать программу, кодировать программу, осуществлять тестирование программы, а также отлаживать программу с использованием инструментов среды программирования, разрабатывать</p>

	<p>алгоритмы и создавать программы на основе концепции структурного программирования Имеет практический опыт: работы с современной средой программирования, проектирования и решения простых задач, разработки алгоритмов и создания программ, а также использования встроенных структур данных языка программирования высокого уровня</p>
<p>1.О.15.02 Программирование на языках высокого уровня</p>	<p>Знает: возможности современных языков программирования, парадигмы программирования, библиотеки алгоритмов и классов, основные возможности современных интегрированных сред разработки программного обеспечения на языках высокого уровня, возможности компиляторов и компоновщиков под различные операционные системы, наборы инструкций для системных утилит автоматической сборки программ, методы разработки алгоритмов и программ в рамках парадигмы структурного программирования на языке высокого уровня; основные синтаксические конструкции языка программирования высокого уровня: операторы, выражения, блоки, ветвления, циклы; методы оценки сложности алгоритмов; функциональные возможности стандартной библиотеки языка высокого уровня Умеет: использовать функциональные возможности современных интегрированных сред разработки программного обеспечения на языках высокого уровня для разработки прикладных программ, использовать утилиты автоматической сборки и развертывания программ в операционных системах, разрабатывать алгоритмы и программ в рамках парадигмы структурного программирования на языке программирования высокого уровня с применением основных синтаксических конструкций и функциональных возможностей стандартной библиотеки языка высокого уровня Имеет практический опыт: работы с основными современными интегрированными средами разработки программного обеспечения на языках высокого уровня, разработки, отладки и тестирования разработанных программ, разработки алгоритмов и программ, отладки, поиска и устранения ошибок программного кода, оценки сложности алгоритмов, использования возможностей стандартной библиотеки, сторонних библиотек программного кода</p>

4. Объём и виды учебной работы

Общая трудоемкость дисциплины составляет 3 з.е., 108 ч., 56,5 ч. контактной работы

Вид учебной работы	Всего часов	Распределение по семестрам в часах	
		Номер семестра	
		3	
Общая трудоёмкость дисциплины	108	108	
<i>Аудиторные занятия:</i>	48	48	
Лекции (Л)	16	16	
Практические занятия, семинары и (или) другие виды аудиторных занятий (ПЗ)	16	16	
Лабораторные работы (ЛР)	16	16	
<i>Самостоятельная работа (СРС)</i>	51,5	51,5	
Подготовка к экзамену.	32	32	
Выполнение заданий на самостоятельную разработку	11,5	11,5	
Изучение теоретического материала	8	8	
Консультации и промежуточная аттестация	8,5	8,5	
Вид контроля (зачет, диф.зачет, экзамен)	-	экзамен	

5. Содержание дисциплины

№ раздела	Наименование разделов дисциплины	Объем аудиторных занятий по видам в часах			
		Всего	Л	ПЗ	ЛР
1	Концепция типов данных. Введение в структуры данных. Полустатические структуры данных – стек, очередь, дек, множество.	6	2	2	2
2	Динамические структуры данных – списки. Классификация структур данных на основе списков. Реализация структур данных на основе списков.	6	2	2	2
3	Рекурсия как инструмент программиста.	6	2	2	2
4	Контейнерные классы Net.Framework как инструмент реализации динамических структур данных.	12	4	4	4
5	Введение в ассоциативные массивы. Хеш-таблицы.	6	2	2	2
6	Деревья. Основные определения. Представление деревьев. Бинарные деревья. Алгоритмы обхода дерева. AVL деревья. Алгоритмы балансировки.	6	2	2	2
7	Графы. Основные определения. Представление графов. Алгоритмы на графах.	6	2	2	2

5.1. Лекции

№ лекции	№ раздела	Наименование или краткое содержание лекционного занятия	Кол-во часов
1	1	Полустатические структуры данных. Концепция типов данных, динамическая память, статические и динамические объекты. Введение в структуры данных: общие понятия и определения. Полустатические типы данных: стек на основе массива, очередь на основе массива, двусторонняя очередь – дек, очередь с приоритетами, множество на основе массива. Примеры прикладных задач на	2

		основе стека и очереди	
2	2	Динамические структуры данных. Динамическая память. Утилита gcnew захвата динамической памяти в управляемом C++ . Список как структура данных. Реализация списка как объекта. Методы списка – добавление, удаление, поиск элементов. Сравнение функциональных возможностей списков и массивов. Разновидности структур на основе списков – список с заглавным звеном, упорядоченный список, двунаправленный список, кольцевой список. Стек и очередь на основе линейного списка, примеры реализации, примеры прикладных задач на основе списков. Мультисписок – определение, система методов, примеры реализации.	2
3	3	Рекурсия. Основные определения. Примеры использования рекурсии в различных областях знаний. Особенности рекурсивных алгоритмов и механизм реализации. Примеры реализации рекурсивных алгоритмов. Фрактальная графика.	2
4	4	Контейнерные классы Net.Framework. Знакомство с библиотеками классов Net.Framework System.Collection и System.Collection.Generic. Реализация динамических структур в Net.Framework. Решение прикладных задач на основе библиотек классов.	2
5	4	Контейнерные классы Net.Framework в библиотеке System.Collection.Generic. Словари. Решение прикладных задач на основе библиотек классов.	2
6	5	Хеш-таблицы. Общие определения, способы организации хеш-таблиц, реализация процедур добавления, удаления, поиска. Способы разрешения коллизий. Примеры реализации словаря на основе хеш-таблицы	2
7	6	Деревья. Понятие дерева и двоичного дерева. Реализация бинарного дерева на основе объектного подхода. Алгоритм формирования дерева. Деревья бинарного поиска. Функциональность дерева. AVL деревья. Преимущества AVL деревьев. Алгоритмы балансировки. Алгоритм построения идеально сбалансированного дерева.	2
8	7	Графы. Основные определения. Графы как структура данных. Взвешенный граф. Ориентированный граф. Представление графов: матрица смежности, матрица инцидентности, матрица весов, списочные представления. Алгоритмы на графах: обход графов (поиск на графах), определение путей в графе. Поиск кратчайшего пути.	2

5.2. Практические занятия, семинары

№ занятия	№ раздела	Наименование или краткое содержание практического занятия, семинара	Кол-во часов
1	1	Реализация полустатических структур данных на основе класса <code>agau</code> : стек или очередь, множество. Полная реализация функционала структуры данных и решение прикладной задачи на основе структуры. Цель работы: изучение полустатических структур данных, знакомство с системой функций, реализация логики прикладной задачи.	2
2	2	Реализация динамической структуры списка одного из указанных типов: линейный список с заглавным звеном, упорядоченный список, двунаправленный список. Реализация актуальных методов списка и решение прикладной задачи на основе списка. Цель работы: изучение динамических структур данных на основе списка, реализация методов списка, реализация логики прикладной задачи.	2
3	3	Изучение механизмов рекурсивных функций. Прямая и косвенная рекурсия. Цель работы: знакомство с рекурсивным определением функции, изучение механизма рекурсии, реализация рекурсивного алгоритма, сравнение рекурсии и итерации.	2

4, 5	4	Знакомство с реализацией динамических структур в Net.Framework. Решение прикладных задач на основе объектов библиотеки классов System.Collection.Generic. Цель работы: знакомство с динамическими объектами Net.Framework, решение прикладной задачи на основе одного из контейнерных типов.	4
6	5	Изучение принципов разработки хеш-таблицы на основе примера, исследование правил подбора хеш-функции. Цель работы: знакомство с хеш-таблицами, реализация словаря на основе примера.	2
7	6	Изучение способов представления дерева, реализация бинарного дерева на основе примера. Реализация алгоритмов добавления, удаления узла. Реализация алгоритмов обхода дерева. Реализация алгоритмов поиска. Цель работы: знакомство со способом представления дерева, реализация рекурсивных алгоритмов обработки дерева.	2
8	7	Изучение способов представления графов на примере. Изучение алгоритмов на графах. Цель работы: реализация алгоритмов на графе.	2

5.3. Лабораторные работы

№ занятия	№ раздела	Наименование или краткое содержание лабораторной работы	Кол-во часов
1	1	Реализация полустатических структур данных на основе класса array: стек или очередь, множество. Полная реализация функционала структуры данных и решение прикладной задачи на основе структуры. Цель работы: изучение полустатических структур данных, знакомство с системой функций, реализация логики прикладной задачи.	2
2	2	Реализация динамической структуры списка одного из указанных типов: линейный список с заглавным звеном, упорядоченный список, двунаправленный список. Реализация актуальных методов списка и решение прикладной задачи на основе списка. Цель работы: изучение динамических структур данных на основе списка, реализация методов списка, реализация логики прикладной задачи.	2
3	3	Изучение механизмов рекурсивных функций. Разработка алгоритмов фрактальной графики. Цель работы: знакомство с рекурсивным определением функции, изучение механизма рекурсии, реализация рекурсивного алгоритма, сравнение рекурсии и итерации.	2
4, 5	4	Знакомство с реализацией динамических структур в Net.Framework. Решение прикладных задач на основе объектов библиотеки классов System.Collection.Generic. Цель работы: знакомство с динамическими объектами Net.Framework, решение прикладной задачи на основе одного из контейнерных типов.	4
6	5	Изучение принципов разработки хеш-таблицы на основе примера, исследование правил подбора хеш-функции. Цель работы: знакомство с хеш-таблицами, реализация словаря на основе примера.	2
7	6	Изучение способов представления дерева, реализация бинарного дерева на основе примера. Реализация алгоритмов добавления, удаления узла. Реализация алгоритмов обхода дерева. Реализация алгоритмов поиска. Цель работы: знакомство со способом представления дерева, реализация рекурсивных алгоритмов обработки дерева.	2
8	7	Изучение способов представления графов на примере. Изучение алгоритмов на графах. Цель работы: реализация алгоритмов на графе.	2

5.4. Самостоятельная работа студента

Выполнение СРС			
Подвид СРС	Список литературы (с указанием разделов, глав, страниц) / ссылка на ресурс	Семестр	Кол-во часов
Подготовка к экзамену.	Курс в электронном ЮУрГУ: конспект лекций, учебные пособия, список вопросов для подготовки.	3	32
Выполнение заданий на самостоятельную разработку	Курс в электронном ЮУрГУ: практические работы с методическими указаниями, примеры кода.	3	11,5
Изучение теоретического материала	Курс в электронном ЮУрГУ: конспект лекций, учебные пособия.	3	8

6. Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации

Контроль качества освоения образовательной программы осуществляется в соответствии с Положением о балльно-рейтинговой системе оценивания результатов учебной деятельности обучающихся.

6.1. Контрольные мероприятия (КМ)

№ КМ	Се-местр	Вид контроля	Название контрольного мероприятия	Вес	Макс. балл	Порядок начисления баллов	Учитывается в ПА
1	3	Текущий контроль	Практическая работа 1. Полустатические структуры магазинного типа	1	7	Разработка объектной модели - 3 б., решение прикладной задачи - 4б. Код должен быть отлажен и протестирован.	экзамен
2	3	Текущий контроль	Практическая работа 2. Динамические структуры данных: линейный список.	1	7	Разработка объектной модели - 3 б., решение прикладной задачи - 4б. Код должен быть отлажен и протестирован.	экзамен
3	3	Текущий контроль	Практическая работа 3. Рекурсия. Разработка алгоритмов фрактальной графики.	1	7	Объектная модель - 1 б., алгоритм - 2 б, визуализация и управление - 4 б. Код должен быть отлажен и протестирован.	экзамен
4	3	Текущий контроль	Практическая работа 4. Контейнерный класс List	1	10	Разработка объектной модели - 3 б., решение прикладной задачи на списке list - 4 б., интерфейс - 3 б. Код должен быть отлажен и протестирован.	экзамен
5	3	Текущий контроль	Практическая работа 5. Интерфейсы	1	5	Разработка объектной модели - 2 б., визуализация - 3б. Код должен быть отлажен и протестирован.	экзамен
6	3	Текущий контроль	Практическая работа 6. Ассоциативные массивы. Хеш-	1	6	Разработка кода - 3 б., визуализация - 3б. Код должен быть отлажен и протестирован.	экзамен

			таблица.				
7	3	Текущий контроль	Практическая работа 7. Деревья.	1	6	Разработка объектной модели - 2 б., решение прикладной задачи - 4б. Код должен быть отлажен и протестирован.	экзамен
8	3	Промежуточная аттестация	Собеседование по вопросам дисциплины	-	10	<p>Студент получает задание на разработку, для выполнения которого отводится один час.</p> <p>В процессе собеседования по заданию осуществляется контроль освоения компетенций студентом. Собеседование проводится с целью проверки уровня знаний, умений, приобретенного опыта, понимания студентом основных методов и законов изучаемой дисциплины, возможности дополнительно повысить свой рейтинг.</p> <p>Критерии оценивания работы:</p> <ul style="list-style-type: none"> - задание выполнено в полном объеме, имеет необходимую функциональность, работает на всех наборах данных - 5 баллов; - задание выполнено, но имеет ограниченный функционал - 4 балла; - задание выполнено частично, имеет ограниченный функционал, не работает на некоторых наборах данных - 3 балла; - задание имеет частичный функционал, - 2 балла; - задание имеет существенные ошибки реализации, работает на ограниченном наборе данных - 1 балл. <p>Критерии оценивания защиты</p> <ul style="list-style-type: none"> - даны полные, развернутые ответы на заданные вопросы, студент ориентируется в основных категориях курса - 5 баллов; - даны ответы на заданные вопросы, студент ориентируется в основных категориях курса с некоторыми затруднениями - 4 балла; - даны частичные ответы на заданные вопросы, студент имеет затруднения в описании основных категорий курса - 3 балла; - даны поверхностные ответы на заданные вопросы, студент не ориентируется в основных категориях курса - 2 балла; - студент не ответил на большую часть заданных вопросов, имеет существенные затруднения в категориях курса - 1 балл; 	экзамен

	эффективности предложенных решений.								
ОПК-8	Имеет практический опыт: использования языков процедурного и объектно-ориентированного программирования; разработки, тестирования и отладки программ в объектно-ориентированном и процедурном стилях; инструментальными средствами разработки программ.	++	++	++	++	++	++	++	++

Типовые контрольные задания по каждому мероприятию находятся в приложениях.

7. Учебно-методическое и информационное обеспечение дисциплины

Печатная учебно-методическая документация

а) основная литература:

Не предусмотрена

б) дополнительная литература:

Не предусмотрена

в) отечественные и зарубежные журналы по дисциплине, имеющиеся в библиотеке:

Не предусмотрены

г) методические указания для студентов по освоению дисциплины:

1. Конова Е.А. Учебное пособие с методическими указаниями по решению задач на проектирование абстрактных типов данных

из них: учебно-методическое обеспечение самостоятельной работы студента:

1. Конова Е.А. Учебное пособие с методическими указаниями по решению задач на проектирование абстрактных типов данных

Электронная учебно-методическая документация

№	Вид литературы	Наименование ресурса в электронной форме	Библиографическое описание
1	Основная литература	Электронный каталог ЮУрГУ	Структуры данных. Программирование на С и С++. Учеб. пособие: Е.А. Конова, Г.А. Поллак, А.М. Ткачев. https://elib.susu.ru/cgi-bin/koha/opac-detail.pl?biblionumber=455993&query_desc=Конова%2С Е.
2	Дополнительная литература	Электронный каталог ЮУрГУ	Алгоритмы и структуры данных. Учебное пособие: Катаргин М. Ю. https://elib.susu.ru/cgi-bin/koha/opac-detail.pl?biblionumber=487174&query_desc=Катаргин%2С М.

Перечень используемого программного обеспечения:

1. Microsoft-Windows(бессрочно)
2. Microsoft-Office(бессрочно)
3. ФГАОУ ВО "ЮУрГУ (НИУ)"-Портал "Электронный ЮУрГУ" (<https://edu.susu.ru>)(бессрочно)
4. Microsoft-Visual Studio(бессрочно)

Перечень используемых профессиональных баз данных и информационных справочных систем:

Нет

8. Материально-техническое обеспечение дисциплины

Вид занятий	№ ауд.	Основное оборудование, стенды, макеты, компьютерная техника, предустановленное программное обеспечение, используемое для различных видов занятий
Самостоятельная работа студента	258 (36)	Специализированный компьютерный класс каф. ЦЭиИТ, среда разработчика Visual Studio.Net, доступ к справочной системе.
Контроль самостоятельной работы	258 (36)	Специализированный компьютерный класс каф. ЦЭиИТ, среда разработчика Visual Studio.Net, доступ к справочной системе.
Лабораторные занятия	258 (36)	Специализированный компьютерный класс каф. ЦЭиИТ, среда разработчика Visual Studio.Net, доступ к справочной системе.
Лекции	229 (36)	Специализированный компьютерный класс каф. ЦЭиИТ, среда разработчика Visual Studio.Net, доступ к справочной системе.
Экзамен	258 (36)	Специализированный компьютерный класс каф. ЦЭиИТ, среда разработчика Visual Studio.Net, доступ к справочной системе.
Практические занятия и семинары	335 (36)	Специализированный компьютерный класс каф. ЦЭиИТ, среда разработчика Visual Studio.Net, доступ к справочной системе.