

государственное бюджетное образовательное учреждение  
высшего образования Московской области  
«Университет «Дубна»  
(государственный университет «Дубна»)

Институт системного анализа и управления  
Кафедра распределенных информационно-вычислительных систем

УТВЕРЖДАЮ:

проректор

по учебно-методической работе

Деникин А.С.

29 » 06 2017 г.

Лист изменений и дополнений в рабочую программу

дисциплины «Структуры и алгоритмы обработки данных» для 2017 года набора

В редакцию для набора 2016г. рабочей программы дисциплины «Структуры и алгоритмы обработки данных» по направлению подготовки 02.03.02 «Фундаментальная информатика и информационные технологии» профиль «Сетевые технологии» изменения не вносятся.

Программа пересмотрена на заседании кафедры Распределенных информационно-вычислительных систем

Протокол заседания № 15 от «23» 06 20 17 г.

Заведующий кафедрой /  / Кореньков В.В.

СОГЛАСОВАНО

Заведующий выпускающей кафедрой /  / Токарева Н.А.

И.о. директора ИСАУ /  / Черемисина Е.Н.

государственное бюджетное образовательное учреждение  
высшего образования Московской области  
«Университет «Дубна»  
(государственный университет «Дубна»)

Институт системного анализа и управления  
Кафедра распределенных информационно-вычислительных систем

УТВЕРЖДАЮ:

проректор

по учебно-методической работе

Деникин А.С.



«29» 06 2017 г.

Лист изменений и дополнений в рабочую программу  
дисциплины «Структуры и алгоритмы обработки данных» для набора 2016 года


В рабочую программу дисциплины «Структуры и алгоритмы обработки данных»  
по направлению подготовки 02.03.02 «Фундаментальная информатика и информационные  
технологии» профиль «Сетевые технологии» изменения не вносятся.

Программа пересмотрена на заседании кафедры Распределенных информационно-  
вычислительных систем

Протокол заседания № 15 от «23» 06 2017 г.

Заведующий кафедрой /  /Кореньков В.В.

СОГЛАСОВАНО

Заведующий выпускающей кафедрой /  /Токарева Н.А.

И.о. директора ИСАУ /  /Черемисина Е.Н.

государственное бюджетное образовательное учреждение  
высшего образования Московской области  
«Университет «Дубна»  
(государственный университет «Дубна»)

Институт системного анализа и управления  
Кафедра распределенных информационно-вычислительных систем



УТВЕРЖДАЮ  
Проректор по учебно-методической работе

/ Деникин А.С./

« 28 » 06 20 17 г.

**Рабочая программа дисциплины**  
**Структуры и алгоритмы обработки данных**

Направление подготовки (специальность)

**02.03.02 Фундаментальная информатика и информационные технологии**

Уровень высшего образования

Бакалавриат

Направленность (профиль) программы (специализация)

**Сетевые технологии**

Форма обучения

очная

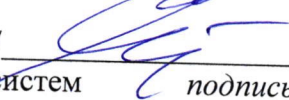
Для 2015 года набора

Дубна, 2017



Рабочая программа дисциплины для 2015 года набора обновлена в связи с принятием новой редакции Федерального государственного образовательного стандарта высшего образования по направлению подготовки 02.03.02 Фундаментальная информатика и информационные технологии, профиль «Сетевые технологии»

Преподаватель:

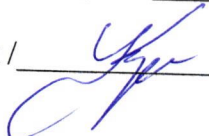
кандидат физико-математических наук, доцент Сеннер А.Е. /  /  
кафедра Распределенных информационно-вычислительных систем подпись

Рабочая программа разработана в соответствии с требованиями ФГОС ВО по направлению подготовки высшего образования

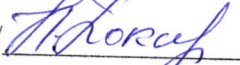
02.03.02 Фундаментальная информатика и информационные технологии

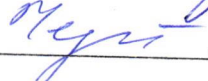
Программа рассмотрена на заседании кафедры Распределенных информационно-вычислительных систем

Протокол заседания № 15 от «23» 06 2014 г.


Заведующий кафедрой /  / Кореньков В.В.

СОГЛАСОВАНО

Заведующий выпускающей кафедрой /  / Токарева Н.А.

И.о. директора Института САУ /  / Черемисина Е.Н.

Эксперт: Объединенный институт ядерных исследований, лаборатория информационных технологий, ведущий научный сотрудник, доктор физико-математических наук

/  / Земляная Елена Валериевна

Подпись Земляной Е.В. заверяю  Подгайный Д.В.  
Учёный секретарь ЛИТ ОИЯИ, к.ф.-м.-н.





## Оглавление

1. Цели и задачи освоения дисциплины .....	4
2. Объекты профессиональной деятельности при изучении дисциплины .....	4
3. Место дисциплины в структуре ОПОП бакалавриата .....	4
4. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями выпускников).....	5
5. Объем дисциплины в зачетных единицах с указанием количества академических или астрономических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся.....	6
3. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических или астрономических часов и виды учебных занятий .....	7
7. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине и методические указания для обучающихся по освоению дисциплины .....	13
8. Применяемые образовательные технологии для различных видов учебных занятий и для контроля освоения обучающимися запланированных результатов обучения.....	15
9. Фонд оценочных средств для промежуточной аттестации по дисциплине .....	17
9.2 Перечень компетенций выпускников образовательной программы с указанием результатов обучения (знаний, умений, владений), характеризующих этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования .....	17
9.3 Описание шкал оценивания.....	17
9.3 Критерии и процедуры оценивания результатов обучения по дисциплине, характеризующих этапы формирования компетенций.....	19
9.4 Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения, характеризующих этапы формирования компетенций и (или) для итогового контроля сформированности компетенции .....	26
9.5 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций. ....	32
10. Ресурсное обеспечение .....	34
10.1 Перечень основной и дополнительной учебной литературы .....	34
10.2 Периодические издания .....	34
10.3 Перечень ресурсов информационно-телекоммуникационной сети «Интернет» .....	34
10.4 Перечень информационных технологий, используемых при осуществлении образовательного процесса, включая программное обеспечение, информационные справочные системы (при необходимости).....	35
11. Описание материально-технической базы .....	35
12. Язык преподавания .....	35

## **1. Цели и задачи освоения дисциплины**

Цель дисциплины – приобретение студентами теоретических знаний и устойчивых практических навыков в области организации базовых структур данных и алгоритмов работы с этими структурами.

Задачами являются:

- системное освоение базовых структур данных;
- осмысленное использование алгоритмов их обработки;
- выработка умения выбора структуры данных наиболее эффективной для решаемой задачи на основе предъявляемых решению требований.

В результате освоения курса студент должен обладать навыками осмысленного выбора структуры данных при решении конкретной задачи, разработки и программной реализации алгоритмов на основе объектно-ориентированной технологии программирования.

## **2. Объекты профессиональной деятельности при изучении дисциплины**

Объектами профессиональной деятельности в рамках изучаемой дисциплины (модуля) являются: программное и информационное обеспечение компьютерных средств, сетей, информационных систем; алгоритмы, библиотеки и пакеты программ; языки программирования, языки описания информационных ресурсов, языки спецификаций, а также инструментальные средства проектирования и создания систем, продуктов и сервисов информационных технологий.

## **3. Место дисциплины в структуре ОПОП бакалавриата**

Учебный курс относится к обязательным дисциплинам вариативной части учебного плана, изучается в 4 (2 курс) и 5 (3 курс) семестрах, форма промежуточной аттестации зачет в 4 семестре и зачет с оценкой в 5 семестре.

Для успешного овладения курса ООП студенты должны иметь удовлетворительные знания, читаемые в учебных курсах ПЯВУ, Компьютерный практикум, Математический анализ, Линейная алгебра, ООП, Дискретная математика, Математическая логика и теория алгоритмов.

**Формы работы студентов** в ходе изучения дисциплины предусмотрены лекционные, практические (семинарские) занятия, выполнение домашних работ.

**Самостоятельная работа студентов**, предусмотренная учебным планом, выполняется в ходе семестра в форме завершения проектов, теоретически проанализированных на практическом (семинарском) занятии и выполнение этапов которых начато на практических (семинарских) занятиях.

Отдельные темы теоретического курса прорабатываются студентами самостоятельно в соответствии с планом самостоятельной работы и конкретными заданиями преподавателя с учетом индивидуальных особенностей студентов.

**Виды текущего контроля** – проверка домашних заданий, индивидуальная защита выполненных проектов.

#### 4. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями выпускников)

После изучения данной дисциплины студенты приобретают знания, умения и владения (навыки), соответствующие результатам основной профессиональной образовательной программы.

Формируемые компетенции	Планируемые результаты обучения по дисциплине характеризующие этапы формирования компетенций
ОПК-2. Способностью применять в профессиональной деятельности современные языки программирования и языки баз данных, методологии системной инженерии, системы автоматизации проектирования, электронные библиотеки и коллекции, сетевые технологии, библиотеки и пакеты программ, современные профессиональные стандарты информационных технологий	<p>32 (ОПК-2) Иметь представление о роли и месте знаний современных языков программирования и языков баз данных, библиотек и пакетов программ при освоении смежных дисциплин и в сфере профессиональной деятельности *)</p> <p>33 (ОПК-2) Знать базовые структуры данных и основные алгоритмы их обработки</p> <p>34 (ОПК-2) Знать основные парадигмы и языки программирования *)</p> <p>36 (ОПК-2) Знать методы построения и оценки эффективности алгоритмов и их программной реализации</p>
ОПК-3. Способностью к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям	<p>У3 (ОПК-3) Умение применять в профессиональной деятельности современные языки программирования для разработки алгоритмических и программных решений, в том числе параллельных вычислений на базе технологий MPI и OpenMP *)</p> <p>В3 (ОПК-3) Владеть современными средами и средствами разработки программного обеспечения *)</p> <p>В4 (ОПК-3) Владеть методами проектирования и конструирования программного обеспечения</p>
ПК-3. Способностью использовать современные инструментальные и вычислительные средства	<p>32 (ПК-3) Знать технологии создания информационно-вычислительных систем</p> <p>У2 (ПК-3) Уметь выбирать алгоритмы и осуществлять их программную реализацию для решения типовых задач предметной области</p> <p>В2 (ПК-3) Владеть современными средами и средствами разработки программного обеспечения</p> <p>В3 (ПК-3) Владеть методами проектирования и конструирования программного обеспечения</p>

\*) Результат обучения сформулирован на основании требований профессионального стандарта: 06.015 СПЕЦИАЛИСТ ПО ИНФОРМАЦИОННЫМ СИСТЕМАМ, утв. Приказом Минтруда России от 18.11.2014, № 896н (ред. от 12.12.2016, № 727н) , регистрационный номер - 153.

**5. Объем дисциплины в зачетных единицах с указанием количества академических или астрономических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся**

Объем дисциплины составляет 7 зачетных единицы, всего 252 часа из которых:

В 4-том семестре (4 зачетные единицы 144 часа):

- **68 часов составляет контактная работа обучающегося с преподавателем:**
  - 34 часа – лекционные занятия,
  - 34 часа – практические занятия,
- **76 часов составляет самостоятельная работа обучающегося.**
- Мероприятия промежуточной аттестации – **зачет**

В 5-том семестре (3 зачетных единицы 108 часов):

- **50 часов составляет контактная работа обучающегося с преподавателем:**
  - 18 часов – лекционные занятия,
  - 32 часа – практические занятия,
- **58 часов составляет самостоятельная работа обучающегося.**
- Мероприятия промежуточной аттестации - **зачет с оценкой**



**6. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических или астрономических часов и виды учебных занятий**

**В 4-том семестре**

Наименование и краткое содержание разделов и тем дисциплины Форма промежуточной аттестации по дисциплине	Всего (часы)	Контактная работа (работа во взаимодействии с преподавателем), часы из них <sup>1</sup>					Всего	Самостоятельная работа обучающегося, часы, из них		
		Лекционные занятия	Семинарские занятия	Практические занятия	Лабораторные занятия	Учебные занятия, направленные на проведение текущего контроля успеваемости (коллоквиумы, практические контрольные занятия и др.)*		Выполнение домашних заданий	Подготовка рефератов и т.п.	Всего
Предмет и задачи дисциплины. Понятие термина структура данных. Важность эффективного и обоснованного выбора структуры данных. Основные принципы выбора структуры данных при проектировании программного обеспечения. Содержание курса, его связь с другими дисциплинами учебного плана.	8	2		2		ПР-1	4	4		4
Статические и динамические структуры данных. Указатель – базовый элемент программирования при реализации структур данных. Понятия статических и динамических структур данных. Достоинства и недостатки. Понятие указателя. Операции над указателями. Использование указателей как базового элемента при программной реализации различных структур данных.	11	2		3		ПР-1	5	6		6
Простейшие структуры данных массив и запись. Определения структур данных массив и запись. Прямой и последовательный методы доступа к данным. Достоинства и недостатки этих методов доступа. Обоснование возможности реализации прямого метода доступа в рассматриваемых	10	4		4		ПР-1	8	2		2

<sup>1</sup> Перечень видов учебных занятий уточняется в соответствии с учебным планом.

Наименование и краткое содержание разделов и тем дисциплины Форма промежуточной аттестации по дисциплине	Всего (часы)	Контактная работа (работа во взаимодействии с преподавателем), часы из них <sup>1</sup>					Всего	Самостоятельная работа обучающегося, часы, из них		
		Лекционные занятия	Семинарские занятия	Практические занятия	Лабораторные занятия	Учебные занятия, направленные на проведение текущего контроля успеваемости (коллоквиумы, практические контрольные занятия и др.)*		Выполнение домашних заданий	Подготовка рефератов и т.п.	Всего
структурах. Преимущества и недостатки этих структур.										
Структура данных стек. Понятие стека. Базовые операции над стеками. Алгоритмы выполнения базовых операций. Программная реализация этих алгоритмов. Примеры областей использования стеков. Организация вызовов процедур и функций и возврата в вызвавший модуль. Обработка алгебраических выражений (обратная польская запись).	16	4		2		ПР-1	6	10		10
Структура данных список. Линейные структуры данных. Понятие списка. Основные виды списков. Базовые операции над списками. Алгоритмы выполнения базовых операций над сцепленными списками. Программная реализация этих алгоритмов. Связанное представление элементов в памяти. Алгоритмы выполнения базовых операций над списками в векторной форме. Программная реализация этих алгоритмов.	19	6		3		ПР-1	9	10		10
Рекурсия и ее применение при работе со структурами данных. Понятие рекурсии. Рекурсивные алгоритмы и структуры. Примеры рекурсивных алгоритмов. Области применения рекурсивных алгоритмов.	14	2		4		ПР-1	6	8		8
Древовидные структуры данных. Иерархические структуры. Отображение этих структур в виде древовидных структур. Древовидные структуры данных. Отображение в виде графа. Основные понятия и определения. Понятия сбалансированное, идеально сбалансированное и упорядоченное дерево. Бинарное дерево. Базовые операции с бинарными деревьями. Алгоритмы выполнения базовых операций над древовидными структурами. Программная реализация этих алгоритмов.	18	4		4		ПР-1	8	10		10

Наименование и краткое содержание разделов и тем дисциплины Форма промежуточной аттестации по дисциплине	Всего (часы)	Контактная работа (работа во взаимодействии с преподавателем), часы из них <sup>1</sup>					Всего	Самостоятельная работа обучающегося, часы, из них		
		Лекционные занятия	Семинарские занятия	Практические занятия	Лабораторные занятия	Учебные занятия, направленные на проведение текущего контроля успеваемости (коллоквиумы, практические контрольные занятия и др.)*		Выполнение домашних заданий	Подготовка рефератов и т.п.	Всего
Структура данных таблица. Хеширование данных. Структура данных таблица. Особенности доступа к данным в таблицах. Понятие хеширования. Организация данных с помощью хеширования. Алгоритмы хеширования. Взаимосвязь размера таблицы и скорости поиска в ней конкретного элемента. Программная реализация алгоритмов хеширования.	16	2		4		ПП-1	6	10		10
Сортировка массивов. Понятие сортировки. Основные составные элементы сортировки. Критерии сравнения различных алгоритмов сортировки. Простейшие алгоритмы сортировки простыми включениями, простым выбором, пузырьком, с помощью дерева. Алгоритм сортировки Шелла. Алгоритм сортировки QuickSort. Сравнительный анализ алгоритмов.	26	6		4		ПП-1	10	16		16
Итоговый обзор курса. Важность обоснованного выбора структур данных и алгоритмов их обработки при проектировании и реализации программного обеспечения.	6	2		4		ПП-1	6			
Промежуточная аттестация (зачет)										
Итого	144	34		34			68	76		76

\*Опрос (ПП-1), Практические работы (ПП-2), Реферат (ПП-3), \*Экзамен (УО-4). Текущий контроль проводится за счет времени, отведенного на аудиторные занятия

**В 5-том семестре:**

Наименование и краткое содержание разделов и тем дисциплины (модуля) Форма промежуточной аттестации по дисциплине (модулю)	Всего (часы)	Контактная работа (работа во взаимодействии с преподавателем), часы из них <sup>2</sup>					Всего	Самостоятельная работа обучающегося, часы, из них		
		Лекционные занятия	Семинарские занятия	Практические занятия	Лабораторные занятия	Учебные занятия, направленные на проведение текущего контроля успеваемости (коллоквиумы, практические контрольные занятия и др.)*		Выполнение домашних заданий	Подготовка рефератов и т.п.	Всего
5-ый семестр										
Структура программы на C++. Шаги построения. Препроцессор. Функции и передача параметров. Указатели и массивы. Z-строки.	10	2		2		ПР-1.1, ПР-2.1	4	6		6
Классы в C++. Виды конструкторов. Время жизни объектов в C++ и C# (.Net). Динамические объекты в C++, управление памятью. Классы с подсчетом ссылок. Наследование в C++. Шаблоны в C++.	15	3		4		ПР-2.2	7	8		8
Алгоритм Бойера-Мура поиска подстроки. Реализация. Таблица суффиксов и префикс-функция. Сложность алгоритма.	10	2		2		ПР-1.2, ПР-2.3	4	6		6
Стандартная библиотека C++ (STL). Основные последовательные и ассоциативные контейнеры и их свойства. Контейнер Pair. Умные указатели. Примеры применения контейнеров.	12	2		4		ПР-2.4	6	6		6
B-Tree. Кучи и их реализация на основе динамического массива vector. Очереди приоритетов. Применения. Контейнеры в .Net. λ-выражения. LINQ.	12	2		4		ПР-1.3, ПР-2.5	6	6		6
Близость строк по Левенштейну. Реализация Вагнера-Фишера. Сложность алгоритма. Применения. Разреженные векторы и матрицы. Реализация на основе STL. Применения. Поиск кратчайшего пути. Волновой алгоритм. Сложность и	14	2		4		ПР-2.6	6	8		8

<sup>2</sup> Перечень видов учебных занятий уточняется в соответствии с учебным планом.

Наименование и краткое содержание разделов и тем дисциплины (модуля) Форма промежуточной аттестации по дисциплине (модулю)	Всего (часы)	Контактная работа (работа во взаимодействии с преподавателем), часы из них <sup>2</sup>					Всего	Самостоятельная работа обучающегося, часы, из них		
		Лекционные занятия	Семинарские занятия	Практические занятия	Лабораторные занятия	Учебные занятия, направленные на проведение текущего контроля успеваемости (коллоквиумы, практические контрольные занятия и др.)*		Выполнение домашних заданий	Подготовка рефератов и т.п.	Всего
5-ый семестр										
применения.										
Кодирование информации. Алгоритмы компрессии данных. Свойства. Другие применения кодирования.	13	2		5		ПР-2.7	7	6		6
Особенности множественного наследования в C++. Динамическое приведение типов. Расширение классов. Регулярные выражения.	15	2		5		ПР-2.8	7	8		8
Неявные типы. Сериализация и десериализация. Иерархии классов и реляционные БД. XML-представление данных. JSON представление данных. Пулы объектов в .Net.	7	1		2		ПР-1.4, ПР-2.9	3	4		4
Промежуточная аттестация (зачет с оценкой)										
Итого	108	18		32			50	58		58

\*Опрос (ПР-1), Практические работы (ПР-2). Текущий контроль проводится за счет времени, отведенного на аудиторные занятия

### Тематика семинарских занятий 4-го семестра

Наименование проектов.	Недели учебного процесса																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Разработка проекта «Стек»	ВЗ	РК	РК	СЗ													
Разработка проекта «Список связанный»				ВЗ		СЗ											
Разработка проекта «Список векторный»						ВЗ		СЗ									
Разработка проекта «Упорядоченное дерево»								ВЗ	РК	РК	СЗ						
Разработка проекта «Идеально сбалансированное дерево»											ВЗ		СЗ				
Разработка проекта «Хеширование»													ВЗ		СЗ		
Разработка проекта «Сортировка массива»															ВЗ		СЗ

ВЗ – выдача задания на самостоятельную работу,

РК – рубежный контроль,

СЗ – сдача и защита задания



### Тематика семинарских занятий 5-го семестра

	Наименование практических занятий (семинаров)
1.	Структура программы на C++. Препроцессор и указатели.
2.	Время жизни объектов в C++ и .Net. Конструкторы и деструкторы в C++.
3.	Объекты с подсчетом ссылок. Алгоритм быстрого поиска подстроки.
4.	Стандартная библиотека C++. Умные указатели.
5.	Кучи и их реализация. Очереди приоритетов.
6.	Близость строк. Реализация алгоритма. Применение для поиска различий в тексте.
7.	Реализация LZ-алгоритма компрессии данных.
8.	Регулярные выражения. Применение для обработки текста.
9.	JSON-сериализация и десериализация в .Net. Пулы объектов в .Net.

## 7 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине и методические указания для обучающихся по освоению дисциплины

При проведении занятий

- лектору:
  - структурировано излагать материал, руководствуясь в большинстве случаев иерархией изложения «сверху-вниз»;
  - четко формулировать определения базовых понятий, методик, алгоритмов и т.п.;
  - по ходу изложения нового материала, базирующегося либо связанным с ранее изложенным материалом кратко повторять базовые положения ранее изложенного, подключая к этому повтору студентов;
  - приводить конкретные примеры использования на практике теоретического излагаемого материала;
  - устраивать совместно со студентами кратковременные «мозговые штурмы» для решения поставленной задачи, а после проводить аналитический разбор предложенных решений и излагать обоснованно оптимальное решение;
  - использовать при проведении лекций презентации, иллюстрирующие базовые положения и примеры иллюстрирующие излагаемый материал;
- Ведущему семинары:
  - четко ставить практическую задачу;

- обсуждать с обучаемыми возможные пути решения, достоинства и недостатки предлагаемых решений;
  - предоставлять обучаемым свободу выбора, реализуемого им решения;
  - консультировать и направлять обучаемого при планировании выполнения и непосредственного выполнения работы;
  - обеспечивать регулярный контроль выполнения заданий;
  - закреплять базовые понятия, методики, алгоритмы и т.п. изученные на лекциях;
  - студенту:
    - вести конспектирование учебного материала;
    - на лекциях применять аббревиатуры и сокращение слов с целью ускорения записи;
    - активно задавать преподавателю уточняющие вопросы с целью эффективного усвоения излагаемого материала;
    - самостоятельно просматривать материал последней лекции, сформулировать возникшие вопросы и выяснять их с преподавателем на последующих аудиторных занятиях;
    - самостоятельно осваивать предложенный на лекции материал, используя рекомендованные и самостоятельно найденные источники требуемой информации;
    - использовать конспекты лекций при выполнении практических работ, подготовке к семинарам, опросам, зачету.
    - По результатам работы на семинарах и выполнения практических заданий выставляются оценки (баллы). Совокупность этих оценок служит основой для проведения зачета. Чем больше сумма баллов, полученных на семинарах и за выполнение заданий, тем меньше вопросов придется рассматривать на зачете и тем выше будет оценка.
- Сроки сдачи заданий ограничены 1-2 неделями с момента их выдачи. Поэтому, необходимо стремиться своевременно выполнять их по мере поступления.

В учебном процессе используются формы компьютерных симуляций, ролевые игры, разбор конкретных ситуаций, проведение «мозговых штурмов», выполнение групповых заданий. В сочетании с внеаудиторной работой это способствует формированию и развитию профессиональных навыков обучающихся.

### **Методическое обеспечение инновационных форм учебных занятий**

1. **Тренинг** – форма активного обучения, целью которого является передача знаний, развитие компетентности профессионального проведения изучаемого предмета и

темы. В рамках тренинга создаются условия для самостоятельного поиска способов решения поставленных задач с использованием полученных на лекциях знаний.

2. **Анализ конкретных ситуаций (CASE-STUDY)** – техника обучения, использующая описание реальных ситуаций. Обучающиеся должны исследовать ситуацию, разобраться в сути проблем, предложить возможные решения и выбрать лучшее из них. Задания основываются на реальном фактическом материале или же приближены к реальной ситуации.

3. **Проектный метод обучения** – это совокупность приемов и способов обучения, при которых студенты с помощью коллективной или индивидуальной деятельности по отбору, распределению и систематизации материала по заданной теме, составляют проект, предлагают и обосновывают его реализацию, доводят проект до работающего состояния.

#### **Методические указания для самостоятельной работы обучающихся**

**Самостоятельная работа** студентов, предусмотренная учебным планом, выполняется в ходе семестра в форме выполнения домашних заданий. Отдельные темы теоретического курса прорабатываются студентами самостоятельно в соответствии с планом самостоятельной работы и конкретными заданиями преподавателя с учетом индивидуальных особенностей студентов. Основной акцент сделан на завершение проектной деятельности, начатой на семинарских занятиях.

### **8. Применяемые образовательные технологии для различных видов учебных занятий и для контроля освоения обучающимися запланированных результатов обучения**

**Формы работы студентов** в ходе изучения дисциплины предусмотрены лекционные, практические занятия (семинары), выполнение домашних работ.

В рамках преподавания дисциплины используются инновационные технологии обучения: чтение лекций, анализ конкретных ситуаций (CASE-STUDY), тренинги, проектное обучение при выполнении практических заданий.

В учебном процессе, помимо чтения лекций используются активные и интерактивные формы обучения: обсуждение отдельных разделов дисциплины, выбор оптимального решения поставленной задачи «мозговым штурмом», выполнение практических работ, домашних заданий. В сочетании с внеаудиторной работой это способствует формированию и развитию профессиональных навыков обучающихся.

Перечень обязательных видов работы студента:

- посещение семинарских занятий;
- ответы на теоретические вопросы на семинаре;
- решение практических задач и заданий на семинаре;
- выполнение и сдача домашних работ.

Интерактивные образовательные технологии, используемые в аудиторных занятиях. Инновационные формы занятий.

Семестр	Вид занятия (Л, ПЗ)	Используемые интерактивные образовательные технологии	Количество часов
4	ПЗ	Анализ конкретных ситуаций (CASE-STUDY), тренинги	4
		Проектный метод обучения	2
Итого интерактивных занятий, часов:			6

При изучении теоретического курса на лекциях предусматривается изложение материала в виде презентации. Активно используется электронная доска, обеспечивающая возможность выделять логические информационные группы разными цветами, и позволяющая изображать стадии процессов в их развитии.

При выполнении некоторых практических занятий студентам предлагается работать в малых группах: учебная группа разбивается на несколько небольших групп – по 2-3 человека. Процесс выполнения заданий осуществляется на основе обмена мнениями и выбора оптимального пути решения. После этого каждая группа самостоятельно и независимо выполняет задание. По завершению выполнения заданий группы сравнивают полученные результаты. Такая форма обучения ориентирована на развитие у обучаемых навыка работы в коллективе.

## **9. Фонд оценочных средств для промежуточной аттестации по дисциплине**

### **9.2 Перечень компетенций выпускников образовательной программы с указанием результатов обучения (знаний, умений, владений), характеризующих этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования**

Полный перечень компетенций выпускников образовательной программы 02.03.02 с указанием результатов обучения (знаний, умений, владений), характеризующих этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования приведён в картах компетенций образовательной программы.

Перечень компетенций выпускников образовательной программы 02.03.02, в формировании которых участвует данная дисциплина представлен в разделе 4 рабочей программы дисциплины. Указание результатов обучения (знаний, умений, владений), характеризующих этапы формирования компетенций, описание показателей и критериев оценивания компетенций на различных этапах их формирования представлено ниже.

### **9.3 Описание шкал оценивания**

*Критерии оценивания ответов студентов на зачете 4-го семестра:*

Зачтено (3-4-5 по критериям оценивания их п 9.3.)	Студентом набрано в течение семестра не менее 50 баллов. Студент владеет терминологией и понятиями, связанными со структурами данных и алгоритмами их обработки.
Не зачтено (1-2 по критериям оценивания их п 9.3.)	Студент набрал менее 50 баллов, крайне плохо владеет терминологией.

Описание шкал оценивания для различных заданий, выполняемых в рамках текущего контроля, представлено в методических материалах, определяющих процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

*Критерии оценивания для зачета 5-го семестра:*

Оценка «отлично»	Своевременно и на хорошем уровне выполнены домашние и практические задания (набрано более 75 баллов). Студент владеет терминологией и понятиями, связанными со структурами данных и алгоритмами их обработки. Ориентируется в возможных решениях для конкретных ситуаций.
Оценка «хорошо»	Своевременно и на хорошем уровне выполнены домашние и практические задания (набрано более 50 баллов). Студент владеет терминологией и понятиями, связанными со структурами данных и алгоритмами их обработки (допускаются отдельные неточности). Ориентируется в возможных решениях для конкретных ситуаций.

Оценка «удовлетворительно»	Выполнены и сданы обязательные домашние и практические задания. набрано не менее 25 баллов. Студент владеет терминологией и понятиями, связанными со структурами данных и алгоритмами их обработки (допускаются отдельные неточности), знает среду разработки Visual Studio.
Оценка «неудовлетворительно»	Не выполнены и не сданы обязательные домашние и практические задания (набрано менее 25 баллов). Студент не владеет терминологией и понятиями, связанными со структурами данных и алгоритмами их обработки, не знает среду разработки Visual Studio.

Описание шкал оценивания для различных заданий, выполняемых в рамках текущего контроля, представлено в методических материалах, определяющих процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.



### 9.3 Критерии и процедуры оценивания результатов обучения по дисциплине, характеризующих этапы формирования компетенций

**Компетенция** ОПК-2. Способность применять в профессиональной деятельности современные языки программирования и языки баз данных, методологии системной инженерии, системы автоматизации проектирования, электронные библиотеки и коллекции, сетевые технологии, библиотеки и пакеты программ, современные профессиональные стандарты информационных технологий

РЕЗУЛЬТАТ ОБУЧЕНИЯ по дисциплине	Уровень освоения компетенции	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ					ПРОЦЕДУРЫ ОЦЕНИВАНИЯ
		по дисциплине (модулю)					
		ШКАЛА оценивания					
		1	2	3	4	5	
32 (ОПК-2) Иметь представление о роли и месте знаний современных языков программирования и языков баз данных, библиотек и пакетов программ при освоении смежных дисциплин и в сфере профессиональной деятельности *)	Базовый	Не знает (не представляет)	Имеет слабое представление о роли и месте знаний современных языков программирования и языков баз данных, библиотек и пакетов программ при освоении смежных дисциплин и в сфере профессиональной деятельности	Имеет общее представление о роли и месте знаний современных языков программирования и языков баз данных, библиотек и пакетов программ при освоении смежных дисциплин и в сфере профессиональной деятельности	Имеет полное представление о роли и месте знаний современных языков программирования и языков баз данных, библиотек и пакетов программ при освоении смежных дисциплин и в сфере профессиональной деятельности	Имеет полное представление о роли и месте знаний современных языков программирования и языков баз данных, библиотек и пакетов программ при освоении смежных дисциплин и в сфере профессиональной деятельности; способен самостоятельно обобщать и систематизировать знания	Индивидуальное устное собеседование, зачет

РЕЗУЛЬТАТ ОБУЧЕНИЯ по дисциплине	Уровень освоения компетенции	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ					ПРОЦЕДУРЫ ОЦЕНИВАНИЯ
		по дисциплине (модулю)					
		ШКАЛА оценивания					
33 (ОПК-2) Знать базовые структуры данных и основные алгоритмы их обработки	Базовый	Не знает	Фрагментарны е знания базовых структур данных и основных алгоритмов их обработки	Имеет общее представление о базовых структурах данных и основных алгоритмах их обработки	Имеет полное представление о базовых структурах данных и основных алгоритмах их обработки; допускает незначительные ошибки	Сформированные и систематизирован ные знания о базовых структурах данных и основных алгоритмах их обработки	Индивидуальное устное собеседование, зачет
34 (ОПК-2) Знать основные парадигмы и языки программирования *)	Базовый	Не знает	Демонстрирует фрагментарные знания основных парадигм и языков программирова ния	Демонстрирует несистематическ ие знания основных парадигм и языков программирован ия	Демонстрирует уверенные и в целом систематические знания основных парадигм и языков программировани я, допускает небольшие ошибки	Демонстрирует уверенные и систематические знания основных парадигм и языков программировани я	Индивидуальное устное собеседование, зачет с оценкой
36 (ОПК-2) Знать методы построения и оценки эффективности алгоритмов и их программной реализации	Базовый	Не знает	Демонстрирует фрагментарные знания методов построения и оценки эффективности алгоритмов и их программной реализации	Демонстрирует знания основных методов построения и оценки эффективности алгоритмов, испытывает затруднения в и их программной	Демонстрирует знания методов построения и оценки эффективности алгоритмов, допускает ошибки в и их программной реализации	Сформированное знания методов построения и оценки эффективности алгоритмов и их программной реализации	Индивидуальное устное собеседование, зачет с оценкой

РЕЗУЛЬТАТ ОБУЧЕНИЯ по дисциплине	Уровень освоения компетенции	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ					ПРОЦЕДУРЫ ОЦЕНИВАНИЯ
		по дисциплине (модулю)					
		ШКАЛА оценивания					
				реализации			

**Компетенция** ОПК-3. Способность к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям

РЕЗУЛЬТАТ ОБУЧЕНИЯ  по дисциплине	Уровень освоения компетенци и	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ					ПРОЦЕДУ РЫ ОЦЕНИВАНИЯ
		по дисциплине (модулю)					
		ШКАЛА оценивания					
		1	2	3	4	5	
У3 (ОПК-3) Умение применять в профессиональной деятельности современные языки программирования для разработки алгоритмических и программных решений, в том числе параллельных вычислений на базе технологий MPI и OpenMP *)	Базовый	Не умеет	Поверхнос тно ориентируется в современных языках программировани я, не умеет разрабатывать алгоритмические и программные решения в области профессионально й деятельности	В целом успешно ориентируется в современных языках программировани я, умеет разрабатывать алгоритмические и программные решения в области профессионально й деятельности; допускает существенные	Уверенно ориентируется в современных языках программирован ия, умеет разрабатывать алгоритмические и программные решения в области профессиональн ой деятельности; допускает несущественные	Сформир ованное умение применять современные языки программирован ия и разрабатывать алгоритмические и программные решения в области профессиональн ой деятельности	Выполнение практического задания

РЕЗУЛЬТАТ ОБУЧЕНИЯ  по дисциплине	Уровень освоения компетенци и	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ					ПРОЦЕДУ РЫ ОЦЕНИВАНИЯ
		по дисциплине (модулю)					
		ШКАЛА оценивания					
				ошибки	ошибки		
В3 (ОПК-3) Владеть современными средами и средствами разработки программного обеспечения *)	Базовый	Не владеет	Не способен самостоятельно осуществлять выбор современных средств разработки программного обеспечения	В целом способен самостоятельно осуществлять выбор некоторых современных сред и средств разработки программного обеспечения и применять их на практике; допускает существенные ошибки	Способен самостоятельно осуществлять выбор современных сред и средств разработки программного обеспечения и применять их на практике с незначительным и ошибками	Способен самостоятельно осуществлять выбор из большого количества современных сред и средств разработки программного обеспечения и применять их на практике без существенных ошибок и недочетов	Выполнение практического задания
В4 (ОПК-3) Владеть методами проектирования и конструирования программного обеспечения	Базовый	Не владеет	Не способен самостоятельно осуществлять выбор современных методов проектирования и конструирования программного обеспечения	В целом способен самостоятельно осуществлять выбор некоторых методов проектирования и конструирования программного обеспечения и применять их на	Способен самостоятельно осуществлять выбор современных методов проектирования и конструирования программного обеспечения и	Способен самостоятельно осуществлять выбор методов проектирования и конструирования программного обеспечения и применять их на практике без	Выполнение практического задания

РЕЗУЛЬТАТ ОБУЧЕНИЯ  по дисциплине	Уровень освоения компетенци и	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ					ПРОЦЕДУ РЫ ОЦЕНИВАНИЯ
		по дисциплине (модулю)					
		ШКАЛА оценивания					
				практике ;допускает существенные ошибки	применять их на практике с незначительным и ошибками	существенных ошибок и недочетов	

**Компетенция ПК-3. Способность использовать современные инструментальные и вычислительные средства**

РЕЗУЛЬТАТ ОБУЧЕНИЯ  по дисциплине	Урове нь освоения компетенции	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ					ПРОЦЕДУ РЫ ОЦЕНИВАНИЯ
		по дисциплине (модулю)  ШКАЛА оценивания					
		1	2	3	4	5	
32 (ПК-3) Знать технологии создания информационно- вычислительных систем	Базовый	Не знает	Знает фрагментарно современные технологии создания информационно- вычислительных систем	Слабо ориентируется в особенностях технологии создания информационно- вычислительных систем; не ориентируется в областях	В целом успешно, но с небольшими пробелами, демонстрирует знание современных технологий создания информационно- вычислительных	Полностью ориентируется в современных технологиях создания информационно- вычислительных систем, относящихся к области профессиональной деятельности ; знает	Индивидуальное устное собеседование, зачет

РЕЗУЛЬТАТ ОБУЧЕНИЯ	Урове нь освоения компетенции	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ					ПРОЦЕДУ РЫ ОЦЕНИВАНИЯ
		по дисциплине (модулю)					
		ШКАЛА оценивания					
				применения и методиках обоснования эффективности технологий создания информационно-вычислительных систем	систем; может обосновать эффективность некоторых технологий	область применения каждой технологии и факторы, обуславливающие эффективность их использования	
У2 (ПК-3) Уметь выбирать алгоритмы и осуществлять их программную реализацию для решения типовых задач предметной области	Базовый	Не умеет	Умеет строить простейшие алгоритмы; не способен осуществлять программную реализацию поставленной задачи	При программной реализации задачи допускает существенные ошибки, не может обосновать выбор методов и приемов программирования	В целом способен осуществлять программную реализацию задачи с небольшими недочетами, не всегда может обосновывать выбор некоторых методов программирования,	Способен осуществлять программную реализацию задачи без ошибок, обосновывать выбор методов и приемов программирования	Выполнение практического задания
В2 (ПК-3) Владеть современными средами и средствами разработки программного обеспечения	Базовый	Не владеет	Не способен самостоятельно осуществлять выбор современных средств разработки программного обеспечения	В целом способен самостоятельно осуществлять выбор некоторых современных сред и средств разработки программного обеспечения и применять их на практике со значительными	В целом способен самостоятельно осуществлять выбор некоторых современных сред и средств разработки программного обеспечения и применять их на практике с незначительными	Способен самостоятельно осуществлять выбор из большого количества современных сред и средств разработки программного обеспечения и применять их на практике без существенных	Выполнение практического задания



РЕЗУЛЬТАТ ОБУЧЕНИЯ  по дисциплине	Урове нь освоения компетенции	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ  по дисциплине (модулю)  ШКАЛА оценивания					ПРОЦЕДУ РЫ ОЦЕНИВАНИЯ
				ошибками	ошибками	ошибок и недочетов	
В3 (ПК-3) Владеть методами проектирования и конструирования программного обеспечения	Базовый	Не владеет	Не владеет теоретическими и практическими методами проектирования и конструирования программного обеспечения	Частично владеет методами проектирования и конструирования программного обеспечения	Владеет на достаточном уровне методами проектирования и конструирования программного обеспечения	Владеет на высоком уровне необходимыми методами проектирования и конструирования программного обеспечения	Выполнение практического задания

#### **9.4 Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения, характеризующих этапы формирования компетенций и (или) для итогового контроля сформированности компетенции**

**Пример материала и вопросы для самоподготовки студента к зачету к 4 семестру**

### **Базовые понятия, используемые в учебном курсе САОД**

**Требование:** Дать ТОЛЬКО ОПРЕДЕЛЕНИЕ.

Т.е. представьте, что Вы объясняете термин человеку, который не знает, что это такое. И ваша задача, используя НЕ БОЛЕЕ ТРЁХ предложений внятно и однозначно объяснить ему этот термин. Если сможете уложиться в два или лучше одно предложение – это только плюс.

**Зачем это нужно?** Во-первых, когда вы четко даете определение – это обеспечивает наилучшее понимание его вам самим. Во-вторых, обеспечивает возможность профессионально разговаривать с коллегами. В-третьих, приучает излагать свои мысли кратко и четко. Можно продолжать эти в-...дцатых, но и этого достаточно.

#### **Термины:**

1. Структура.
2. Структура данных.
3. Элемент.
4. Ключ.
5. Скорость поиска элемента в структуре.
6. Массив.
7. Упорядоченный массив.
8. Стек.
9. Список.
10. Список линейный.
11. Список сцепленный.
12. Список в векторной форме.
13. Структура дерево.
14. Бинарное дерево.
15. Упорядоченное дерево.
16. Сбалансированное дерево.
17. Идеально сбалансированное дерево.
18. Функция хэширования.
19. Сортировка элементов массива.

#### **ВОПРОСЫ к зачету по курсу 4-го семестра**

##### ***ОБЩИЕ ПОНЯТИЯ***

1. Понятие структуры вообще и структуры данных.
2. Понятие элемента. Структура элемента.
3. Понятие ключа.

## ***МАССИВЫ***

1. Структура массив. Механизм доступа к элементам массива.
2. Упорядоченный и неупорядоченный массивы.
3. Алгоритм поиска элемента в неупорядоченном массиве.
4. Алгоритм поиска элемента в упорядоченном массиве.
5. Скорость поиска элемента в массиве.
6. Достоинства и недостатки структуры массив.

## ***СТЕК***

1. Структура стек. Дисциплина обслуживания занесения в стек и извлечения из стека данных.
2. Базовые операции над стеком.
3. Примеры использования стека из жизни и из ИТ-области.
4. Принципы реализации структуры стек.

## ***СПИСКИ***

1. Списки. Основные виды списков. Операции над ними. Достоинства и недостатки списков.
2. Упорядоченные списки. Основные операции над ними. Принципы реализации упорядоченных списков.
3. Списки. Операции над ними. Сортировка списков.
4. Связные и двусвязные списки. Операции над ними. Примеры применения.
5. Списки. Операции над ними. Поиск элементов в ветвящихся списках и их изменение.
6. Списки. Операции над ними. Поиск в упорядоченных ветвящихся списках.

## ***ДЕРЕВЬЯ***

1. Основные понятия, определения и классификация древесных структур данных. Основные операции с деревьями. Изображение древовидной структуры.
2. Древовидные структуры данных и принципы их реализации.
3. Древовидные структуры данных. Алгоритмы подсчета числа узлов и удаления дерева.
4. Древовидные структуры данных. Поиск по дереву и добавление нового элемента в дерево.
5. Древовидные структуры данных. Упорядоченные и сбалансированные деревья. Поиск по упорядоченному дереву и добавление элемента.
6. Древовидные структуры данных. Упорядоченные бинарные деревья. Их применение для индексирования файлов. Поиск данных по ключу.
7. Древовидные структуры данных. Упорядоченные бинарные деревья. Включение и удаление из бинарного дерева.

## **ТАБЛИЦЫ**

1. Таблицы как структуры данных. Операции над таблицами. Примеры применения таблиц.
2. Таблицы с вычисляемыми адресами (хеширование). Достоинства и недостатки.

## **ПОИСК и СОРТИРОВКА**

1. Задача поиска информации. Поиск информации в массивах. Последовательный просмотр. Двоичный поиск.
2. Сортировка массивов. Алгоритм сортировки пузырьком. Оценка количества сравнений и перестановок.
3. Сортировка массивов. Алгоритм сортировки Шелла. Оценка количества сравнений и перестановок.
4. Сортировка массивов. Алгоритм сортировки QuickSort. Оценка количества сравнений и перестановок.
5. Сортировка массивов. Алгоритм сортировки с помощью двоичного дерева. Оценка эффективности алгоритма.
6. Сортировка массивов. Алгоритм сортировки пирамидальный. Оценка эффективности алгоритма.
7. Критерии выбора алгоритма. Сравнение алгоритмов сортировки.

## **Перечень вопросов к зачету с оценкой 5-го семестра**

1. Структура модуля в C++. Порядок построения программ на C++. Роль препроцессора. Инструкции препроцессора.
2. Понятие указателя. Операции над указателями в C++. Указатели и ссылки. Копирование, сравнение и вычисление длины z-строк.
3. Передача параметров в методы и функции в C++ и C#. Особенности массивов в C++, ReferenceType и ValueType в .Net.
4. Создание и разрушение объектов в C++. Конструктор копирования. Порядок создания производных и составных объектов.
5. Создание и уничтожение объектов в C#. Интерфейс IDisposable и освобождение ресурсов. Понятие равенства и тождественности объектов, копирование объектов в C#.
6. Особенности наследования в C++ и C#. Проблемы множественного наследования в C++.
7. Переопределение операций в C++ и C#. Inline методы и эффективность программ.
8. Ассоциативные и последовательные контейнеры STL. Понятие итератора. Общие свойства контейнеров. Особенности и применение каждого из контейнеров.

9. Использование контейнеров STL. Методы резервирования памяти контейнерами STL. Требования к элементам контейнеров.
10. Обобщенные контейнеры .Net. Интерфейс IEnumerable и оператор foreach. Сравнение обобщенных контейнеров и ArrayList. Назначение метода GetHashCode в .Net.
11. Реализация объектов с подсчетом ссылок. Эффективность создания копий в C++. Другие применения объектов с подсчетом ссылок. Умные указатели в STL.
12. Средства обработки текстовых данных. Форматирование строк в C++ и C#. Эффективность формирования строк и класс StringBuilder в C#.
13. Язык регулярных выражений. Классы символов, исчислители, последовательности и несимвольные подстановки. Валидация email-адреса при помощи регулярных выражений.
14. Технология обработки данных LINQ в .Net (C#). Парадигма технологии. Основные виды задач, решаемых LINQ. Синтаксис и примеры запросов в LINQ.
15. Проекция в LINQ. Анонимные классы в C#.
16. Алгоритм Бойера – Мура быстрого поиска строки в тексте.
17. Понятие расстояния между строками (метрика Левенштейна). Алгоритм вычисления расстояния между двумя строками. Применения алгоритма. Сложность алгоритма.
18. Разреженные матрицы. Их назначение. Реализация разреженных матриц на основе шаблоном map библиотеки STL в C++ или обобщенного типа Dictionary в C#.
19. Задача поиска кратчайшего пути на графе. Волновой алгоритм. Область применимости. Сложность.
20. Кодирование информации. Применения. LZ-методы компрессии данных.
21. Множественное наследование в C++. Особенности множественного наследования. Виртуальное наследование.
22. Динамическое приведение типов в C++ и C#.
23. Понятие сериализации/десериализации. JSON представление данных. Роль в хранении и обмене информацией.
24. Методы реализации иерархии классов в реляционной БД.
25. Методы и функции как параметры. Делегаты в C# и указатели на функции в C++. Лямбда-выражения и анонимные функции в C#.
26. Расширения классов и частичные классы в .Net (C#).
27. Буферизация объектов в .Net для решения проблемы большого количества висячих объектов.

## Материалы для текущего контроля в 5-том семестре

Формы текущего контроля: проверка выполнения заданий, прием заданий и лабораторных работ. Могут проводиться и дополнительные опросы, если не удастся закрепить тот или иной практически важный материал в упражнениях или домашних заданиях.

*Вопросы к 1-му опросу:*

1. Из каких шагов состоит построения программы на C++?
2. Назовите 3-4 инструкции препроцессора C++ и объясните, как они обрабатываются.
3. Из чего (каких файлов) состоит исходный текст модуля C++?
4. Что содержится в заголовочном файле C++?
5. Что содержится в сrr файлах?
6. Что такое указатель? Как его объявить?
7. Как связаны указатели и имя массива в C++?
8. Какие операции определены над указателями?
9. Что такое z-строка?

*Вопросы к 1-ой лабораторной работе:*

10. Опишите, как создать “автоматический” объект в C++. Укажите различия с C#.
11. Опишите, как уничтожить “автоматический” объект в C++. Укажите различия с C#.
12. Охарактеризуйте время жизни глобальных “автоматических” объектов в C++.
13. Опишите правила работы и время жизни “динамических” объектов в C++.
14. Опишите различие в понятии ссылка в смысле C++ и в смысле .Net.
15. Какие случаи неявного копирования объектов встречаются в C++?
16. Для чего предназначен интерфейс IDisposable в C#?
17. Опишите порядок создания сложных объектов. Что создается раньше, предок или сам объект? Объекты члены класса или сам объект?
18. Как вызвать конструктор предка с параметром в C++? В C#?
19. Как вызвать конструктор члена класса с параметром?
20. Как можно заменить отсутствие глобальных объектов в C#?

*Вопросы ко 2-ой работе:*

1. Опишите, как выделяется память под экземпляр строки и вектора. Что происходит при добавлении элементов в контейнер?



2. С чем связано такое нетривиальное поведение контейнеров STL в отношении памяти?
3. Если нужно построить массив целых чисел от 1 до 1000. Что будет более эффективным, объявить массив на 1000 целых и в цикле заполнить его значениями или объявить массив нулевой длины и заполнить его в цикле операцией `push_back`? Почему? Измеряйте время необходимое в одном и в другом случае.
4. Почему у списка нет метода `capacity`?
5. Опишите какими методами и почему должны обладать классы, которые могут являться элементами контейнеров STL.
6. В каком случае понадобится конструктор копирования?
7. В каком случае понадобится операция `<`?
8. В каком случае понадобится операция `==`?
9. Почему при определении операций в заданиях необходимо указывать `const` и `&`?
10. Почему переопределять операции не обязательно, если в контейнере хранятся указатели на объекты?
11. Нужно ли переопределять операции для класса, если в контейнере хранятся умные указатели на объекты?
12. Опишите судьбу объектов, указатели на которые хранятся в контейнере STL. Будут ли эти объекты автоматически уничтожены, если разрушится контейнер? Объясните почему.
13. Опишите судьбу объектов, умные указатели на которые хранятся в контейнере STL. Будут ли эти объекты автоматически уничтожены, если разрушится контейнер? Объясните почему.
14. Объясните различие в эффективности списка и множества при поиске элементов.

*Вопросы к 3-ей работе:*

1. Что означает символ `@` в начале строкового литерала в C#? Как будет выглядеть литерал `@("(\W*)(\w+)(\W+)(\w+)(\W+)(\w+)(\W*)"` без использования символа `@`?
2. Что означают группы: `\w*`, `\W*`, `\w+` и `\W+`?
3. Что означает группа `[^<]*`?
4. Как сделать именованную группу? Приведите пример использования именованной группы.

5. Где при выполнении лабораторной работы вы использовали:
  - a. Исчислители?
  - b. Классы символов?
  - c. Обычные символы?
  - d. Обратные ссылки?
  - e. Подстановки?

*Примерный перечень тем заданий для самостоятельной работы*

1. Алгоритм Бойера-Мура поиска шаблона в тексте.
2. Технология LINQ в .Net.
3. Кодирование данных. Компрессия данных.
4. Кодирование данных. Структурирование потоков данных.
5. Вычисление расстояния по Левенштейну и редакционного предписания для строк.
6. Реализация разреженных матриц и STL.
7. Сериализация/десериализация JSON в .Net.
8. Парсинг XML-документов в .Net.
9. Обработка текстов при помощи регулярных выражений в .Net.

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

Для контроля знаний студентов по данной дисциплине необходимо проводить текущий и промежуточный контроль. Текущий контроль выполняется в виде приема допусков и защит практических и индивидуальных работ, устного опроса на лекциях и практических занятиях. Промежуточный контроль проводится в виде учета выполняемой в семестре работы на основе балльно-рейтинговой системы. При рейтинговой системе все знания, умения и навыки, приобретаемые студентами в результате изучения дисциплины, оцениваются в баллах.

Оценка качества работы в семестре в рейтинговой системе является накопительной и используется для оценивания системной работы студентов в течение всего периода обучения. Максимально возможная сумма баллов по всем видам заданий приравнивается 100. Учитывается посещаемость занятий (максимально 30 баллов). Каждое практическое задание оценивается в баллах в зависимости от его сложности. Максимальный балл за каждый вид задания определяется преподавателем в зависимости от уровня сложности задания в начале семестра и доводится до сведения студентов.

Чтобы быть допущенным к зачету, студент обязан проделать и сдать все практические работы. Работы, пропущенные без уважительной причины, а также работы,

не защищенные студентом в течение 2-х недель после проведения практической части, оцениваются преподавателем с понижающим коэффициентом 0,5.

Результаты работы студентов фиксируются преподавателем в электронном журнале успеваемости и размещаются на ресурсе доступном студентам.

Контроль в конце семестре – зачет, зачет с оценкой. Оценка выставляется на основе устной проверки полученных теоретических знаний и результатов практической работы в семестре.

## 10. Ресурсное обеспечение

### 10.1 Перечень основной и дополнительной учебной литературы

#### Основная литература

1. Алексеев, В.Е. Структуры данных. Модели вычислений / В.Е. Алексеев, В.А. Таланов. - 2-е изд., исправ. - М. : Национальный Открытый Университет «ИНТУИТ», 2016. - 248 с. : схем., ил. - (Основы информационных технологий). - Библиогр. в кн. - ISBN 5-9556-0066-3 ; То же [Электронный ресурс]. - URL: [http://biblioclub.ru/index.php?page=book\\_red&id=428782&sr=1](http://biblioclub.ru/index.php?page=book_red&id=428782&sr=1)
2. Структуры и алгоритмы обработки данных: Учебное пособие / В.Д. Колдаев. - М.: ИЦ РИОР: НИЦ ИНФРА-М, 2014. - 296 с.: 60х90 1/16. - (Высшее образование: Бакалавриат). (переплет) ISBN 978-5-369-01264-2, 500 экз. <http://znanium.com/catalog.php?bookinfo=418290>
3. Алгоритмы и структуры данных: Учебник / Белов В.В., Чистякова В.И. - М.: КУРС, НИЦ ИНФРА-М, 2016. - 240 с.: 60х90 1/16. - (Бакалавриат) (Переплёт 7БЦ) ISBN 978-5-906818-25-6 . <http://znanium.com/catalog.php?bookinfo=551224>

#### Дополнительная литература

1. Зыков, С. В. Программирование : учебник и практикум для академического бакалавриата / С. В. Зыков. — М. : Издательство Юрайт, 2017. — 320 с. — (Серия : Бакалавр. Академический курс). — ISBN 978-5-534-02444-9. <https://biblio-online.ru/book/122D27F3-13E4-4095-8946-C619F0FCC5C3>
2. Структуры данных и проектирование программ [Электронный ресурс] / Р. Л. Круз ; пер. с англ. — 2-е изд. (эл.). — М. : БИНОМ. Лаборатория знаний, 2014. — 765 с. : ил. — (Программисту). ISBN 978-5-9963-1308-2. <http://znanium.com/catalog.php?bookinfo=543549>

### 10.2 Периодические издания

1. Вестник Международного университета природы, общества и человека "Дубна" / гл. ред. Е.Н. Черемисина. – Дубна: Международный университет природы, общества и человека "Дубна".- (Аспекты разработки программного обеспечения). – Журнал.
2. Программные продукты и системы: научно-практическое издание. / гл. ред. С.В. Емельянов. – Тверь: МНИИПУ. – Журнал. – Международное научно-практическое приложение к журналу "Проблемы теории и практики управления".

Сетевое научное издание «Системный анализ в науке и образовании» (свидетельство о регистрации Эл № ФС77-51141 от 14 сентября 2012 г.).

### 10.3 Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

#### Электронно-библиотечные системы и базы данных научного содержания

1. Электронно-библиотечная система «Университетская библиотека онлайн»: [www.biblioclub.ru](http://www.biblioclub.ru)
2. Электронно-библиотечная система «ЮРАЙТ» <https://biblio-online.ru>

3. Научная электронная библиотека (НЭБ): <http://elibrary.ru/defaultx.asp>
4. Электронно-библиотечная система «Znanium» <http://znanium.com/>
5. БД российских журналов East View : <http://dlib.eastview.com>
6. Базы данных компании EBSCO Publishing: <http://search.ebscohost.com/>
7. БД российских научных журналов на Elibrary.ru (РУНЭБ): <http://elibrary.ru/defaultx.asp>.
8. <http://www.scopus.com/home.url>

### **Профессиональные ресурсы сети «Интернет»**

1. Национальный открытый университет «Интуит» <http://www.intuit.ru> .
2. Кузниченко М.А. Динамические структуры данных. Учебное пособие. Орск, Изд-во Орского гуманитарного технологического ин-та 2011 стр. 102 ISBN: 978-5-8424-0551-0 ББК: 3973.26-018я73-1
3. Костюкова Н. И. Комбинаторные алгоритмы для программистов УДК: 519.1(07) ББК: 32.973 М.: Национальный Открытый Университет «ИНТУИТ», 2016 Объем: 217 2-е изд./, исправ/
4. Мейер Б. Инструменты, алгоритмы и структуры данных Издательство: Национальный Открытый Университет «ИНТУИТ», 2016
5. Алексеев В. Е., Таланов В. А. Структуры данных. Модели вычислений Издательство: Национальный Открытый Университет «ИНТУИТ», 2016
6. Сообщество аналитиков: <http://www.uml2.ru/>.
7. Материалы IT-портала Центра информационных технологий <http://www.citforum.ru>.

#### **10.4 Перечень информационных технологий, используемых при осуществлении образовательного процесса, включая программное обеспечение, информационные справочные системы (при необходимости)**

Используется лицензионное программное обеспечение:

- VisualStudio 2013 (свободно распространяемое ПО)
- MS Power Point 2013
- MS Word Office2013
- Браузер Интернет

### **11. Описание материально-технической базы**

Для проведения лекционных занятий требуется специализированная аудитория, оснащенная компьютером, проектором, электронной доской.

Для проведения семинарских занятий требуется специализированный компьютерный класс, подключенный к сети Интернет и обеспечением доступа в электронную информационно-образовательную среду университета. На компьютерах необходимо наличие среды программирования поддерживающую C#.

### **12 Язык преподавания**

Русский язык