

НЕГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ ГУМАНИТАРНЫЙ
УНИВЕРСИТЕТ ПРОФСОЮЗОВ»

Кафедра Информатики и математики

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Структуры данных в предметной области

Основная профессиональная образовательная программа
высшего образования программы бакалавриата
по направлению

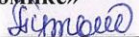
09.03.03 «Прикладная информатика»

Профиль подготовки «Прикладная информатика в экономике»

Квалификация:

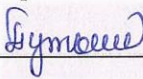
Бакалавр

Согласовано:
Руководитель ОПОП по направлению
09.03.03 – «Прикладная информатика»
Профиль «Прикладная информатика
в экономике»

 /Путькина Л.В.

Рассмотрена и утверждена на заседании кафедры

«01» июня 2020 г., протокол № 10

Зав. кафедрой  /Путькина Л.В.

Рекомендована решением
Методического совета

«15» июня 2020 г., протокол №15

Секретарь МС  /Волкова А.М.

Авторы-разработчики:

 /Мокрый В.Ю.

Санкт-Петербург

СТРУКТУРА

1. Цель и задачи освоения дисциплины
2. Место дисциплины в структуре ОПОП
3. Требования к результатам освоения дисциплины
4. Тематический план изучения дисциплины
5. Содержание разделов и тем дисциплины
6. План практических (подгрупповых) занятий
7. Образовательные технологии
8. План самостоятельной работы студентов
9. Контроль знаний по дисциплине
10. Учебно-методическое и информационное обеспечение дисциплины
11. Материально-техническое обеспечение дисциплины

Учебно-методическое обеспечение самостоятельной работы студентов

1. Методические рекомендации по организации самостоятельной работы студентов
2. Методические рекомендации по подготовке к практическим (подгрупповым) занятиям
3. Методические рекомендации по написанию контрольных работ
4. Методические рекомендации по написанию курсовой работы

Оценочные и методические материалы

1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы
2. Описание показателей и критериев оценивания компетенций, шкал оценивания
3. Типовые контрольные задания и методические материалы, процедуры оценивания знаний, умений и навыков

Глоссарий

Методические рекомендации для преподавателя по дисциплине

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

1.Цель и задачи освоения дисциплины:

Целью дисциплины «Структуры данных в предметной области» является изучение структур данных, которые используются при программировании решения задач из предметной области указанной специальности и освоение алгоритмов обработки данных (на примерах из профессиональной деятельности).

Основные задачи дисциплины:

- Дальнейшее развитие и углубление навыков алгоритмизации и программирования при решении прикладных задач предметной области.
- Знакомство с реализацией типовых алгоритмов, структур данных и их модификаций на выбранном рабочем языке программирования (C++).
- Изучение конструирования и использования сложных (динамических) структур данных на основе модели абстрактного типа данных (спецификация + представление + реализация).
- Ознакомление студентов с основными структурами данных, используемыми в современном программировании, применительно к специфике будущей профессиональной деятельности.
- Формирование базовых теоретических понятий, лежащих в основе процесса разработки алгоритмов и структур данных.
- Формирование навыков использования различных структур данных при разработке программ на выбранном языке программирования.
- Формирование представлений и знаний об основных классах алгоритмов (поиск, сортировка, алгоритмы на графах и т.п.), используемых в них структурах данных и общих схемах решения задач на их основе.
- Формирование представления об анализе сложности алгоритмов и программ.

2.Место дисциплины в структуре ОПОП

Междисциплинарные связи с обеспечиваемыми (последующими) дисциплинами

№ п/п	Наименование обеспечиваемых (последующих) дисциплин	№ № разделов данной дисциплины, необходимых для изучения обеспечиваемых дисциплин			
		1	2	3	4
1	Объектно-ориентированное программирование	+	+	+	+
2	Высокоуровневые методы информатики и программирования	+	+	+	+

3. Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование следующих компетенций с установленными к ним индикаторами:

Компетенции и индикаторы их достижения

Код и наименование компетенции	Код и наименование индикатора достижения компетенции
ПК-18 - Способен программировать приложения и создавать программные прототипы для решения прикладных задач	ПК-18.1. – Знать основные методы разработки алгоритмов и программ, структуры данных, используемые для представления типовых информационных объектов. ПК-18.2 – Уметь разрабатывать алгоритмы, используя изложенные в курсе общие схемы, методы и приёмы построения алгоритмов, выбирая подходящие структуры данных для представления информационных объектов ПК-18.3. – Владеть инструментарием программирования в системе программирования Microsoft Visual Studio.

4. Тематический план изучения дисциплины

См. приложение

5. Содержание разделов и тем дисциплины

РАЗДЕЛ 1 (Модуль 1). Введение

Тема 1. Введение. Предмет дисциплины и её задачи. Связь с другими дисциплинами учебного плана направления и специальности. Классификация структур данных. Простые типы данных. Составные (структурированные) типы данных. Данные и информация. Понятие данных статической и динамической структуры. Освоение интегрированной среды программирования Visual Studio. Работа с файлами.

РАЗДЕЛ 2 (Модуль 2). Базовые динамические структуры данных

Тема 2. Данные динамической структуры. Стек, очередь и дек как линейные списки (последовательности) с ограниченными наборами операций (доступа). Линейные и кольцевые списки. Представление и реализация списков на примере обслуживания заявок. Стек и очередь как абстрактные типы данных. Примеры алгоритмов, использующих стек и очередь.

Тема 3. Линейные динамические списки. Определение и функциональная спецификация линейных списков. Базовые операции над элементами списка (доступ, вставка, исключение, обход). Двусвязный линейный список. Сортировка списка. Выделение и освобождение динамической памяти. Примеры использования двусвязных динамических списков.

РАЗДЕЛ 3 (Модуль 3). Рекурсивные алгоритмы, поиск и сортировка

Тема 4. Рекурсия. Поиск и сортировка. Рекурсия. Понятие рекурсии. Пример использования рекурсии. Алгоритмы поиска и сортировки. Алгоритм двоичного поиска. Сортировка методом пузырька. Быстрая сортировка методом Хоара. Обзор других методов сортировки и поиска.

РАЗДЕЛ 4 (Модуль 4). Применение базовых динамических структур данных

Тема 5. Древоподобные структуры. Основная терминология. Бинарные (двоичные) деревья. Графическое представление древоподобной структуры (вложенные скобки, вложенные множества, ломаная последовательность, граф). Дерево как динамическая структура данных. Алгоритм построения бинарного дерева. Представление дерева в памяти компьютера. Использование деревьев для кодирования информации. Деревья бинарного поиска. Обходы бинарных деревьев: рекурсивные и не рекурсивные алгоритмы. Примеры практического применения бинарных деревьев.

Тема 6. Обзор применения структурированных типов данных. Обзор применения структурированных типов данных. Массивы и разреженные матрицы. Графы. Матрица инцидентности и смежности. Представление графов. Алгоритмы на графах: поиски вершин в графах, поиск в глубину, поиск в ширину. Алгоритм поиска на графах в ширину. Применение структур данных в решении прикладных задач.

6. План практических (подгрупповых) занятий

№ п/п	Наименование темы дисциплины	Наименование и содержание практических (подгрупповых) занятий, литература для подготовки к занятиям	Формируемые компетенции	Формы контроля усвоения знаний
1.	Введение	Предмет дисциплины и её задачи. Связь с другими дисциплинами учебного плана направления и специальности. Классификация структур данных. Простые типы данных. Составные (структурированные) типы данных. Данные и информация. Понятие данных статической и динамической структуры. Освоение интегрированной среды программирования Visual C++. Работа с файлами. Задание. Выполнение лабораторной работы №1 «Освоение интегрированной среды Visual C++/Visual Studio» в ЭУМК по дисциплине. Литература: 1 – 3.	ПК-18	Опрос, проверка заданий для самостоятельного выполнения, зачёт.
2.	Данные динамической структуры	Структуры данных фиксированного и переменного размера. Стек, очередь и дек как линейные списки (последовательности) с ограниченными наборами операций (доступа). Линейные и кольцевые списки. Представление и реализация списков на примере обслуживания заявок. Стек и очередь как абстрактные типы данных. Примеры алгоритмов, использующих стек и очередь. Задания. 1. Выполнение работы №2 из ЭУМК СПбГУП по дисциплине «Структуры данных в предметной области» по теме «Обработка линейных	ПК-18	Опрос, проверка заданий для самостоятельного выполнения, зачёт.

		динамических списков». 2. Выполнение исследовательской работы на тему «Примеры алгоритмов, использующих стек и очередь». Литература: 1 – 3.		
3.	Линейные динамические списки	Линейные динамические списки. Определение и функциональная спецификация линейных списков. Базовые операции над элементами списка (доступ, вставка, исключение, обход). Двусвязный линейный список. Сортировка списка. Выделение и освобождение динамической памяти. Примеры использования двусвязных динамических списков (на примере учета вкладчиков банка). Задания. 1. Выполнение работы №2 из ЭУМК СПбГУП по дисциплине «Структуры данных в предметной области» по теме «Обработка линейных динамических списков». 2. Исследовательская работа на тему «Практическое применение линейных динамических списков». Литература: 1 – 3.	ПК-18	Опрос, проверка заданий для самостоятельного выполнения, зачёт.
4.	Рекурсия. Поиск и сортировка	Рекурсия. Понятие рекурсии. Пример использования рекурсии. Алгоритмы поиска и сортировки. Алгоритм двоичного поиска. Сортировка методом пузырька. Быстрая сортировка методом Хоара. Обзор других методов сортировки и поиска. Задания. 1. Выполнение практических заданий, предложенных преподавателем. 2. Исследовательская работа на тему «Обзор других методов сортировки и поиска». Литература: 1 – 3.	ПК-18	Опрос, проверка заданий для самостоятельного выполнения, зачёт.
5.	Древовидные структуры	Древовидные структуры. Основная терминология. Бинарные (двоичные) деревья. Графическое представление древовидной структуры (вложенные скобки, вложенные множества, ломаная последовательность, граф). Дерево как динамическая структура данных. Алгоритм построения бинарного дерева. Представление	ПК-18	Опрос, проверка заданий для самостоятельного выполнения, зачёт.

		<p>дерева в памяти компьютера. Использование деревьев для кодирования информации.</p> <p>Деревья бинарного поиска. Обходы бинарных деревьев: рекурсивные и не рекурсивные алгоритмы. Примеры практического применения бинарных деревьев.</p> <p>Задания.</p> <ol style="list-style-type: none"> 1. Выполнение практических заданий, предложенных преподавателем. 2. Выполнение работы №3 из ЭУМК СПбГУП по дисциплине «Структуры данных в предметной области» по теме: «Алгоритмы построения и обхода бинарных деревьев». 3. Исследовательская работа на тему «Примеры использования бинарных деревьев в прикладных задачах». <p>Литература: 1 – 3.</p>		
6	Обзор применения структурированных типов данных	<p>Обзор применения структурированных типов данных. Массивы и разреженные матрицы.</p> <p>Графы. Матрица инцидентности и смежности. Представление графов.</p> <p>Алгоритмы на графах: поиски вершин в графах, поиск в глубину, поиск в ширину.</p> <p>Алгоритм поиска на графах в ширину</p> <p>Применение структур данных в решении прикладных задач.</p> <p>Задания.</p> <ol style="list-style-type: none"> 1. Выполнение практических заданий, предложенных преподавателем. 2. Представление отчёта о результатах исследования по теме «Примеры применения структурированных типов данных при решении прикладных задач». <p>Литература: 1 – 3.</p>	ПК-18	Опрос, проверка заданий для самостоятельного выполнения, зачёт.

7. Образовательные технологии

При проведении учебных занятий по дисциплине для успешного освоения применяются различные образовательные технологии, которые обеспечивают развитие навыков командной работы, межличностной коммуникации, принятия решений, лидерских качеств.

Методы / Формы	Лекции (Л)	Подгрупповые занятия (П)
Диалого-дискуссионное обсуждение проблем	+	+
Работа в команде		+
Игра		+
Поисковый метод	+	+
Проектный метод		+
Исследовательский метод	+	+
Выступление в роли обучающего		+

8. План самостоятельной работы студентов

№ п/п	Содержание самостоятельной работы студентов	Формируемые компетенции	Форма отчетности студента
1	Изучение литературы и научных работ по темам дисциплины	ПК-18	Представление развёрнутого списка в процессе выполнения исследовательской работы
2	Подготовка к практическому занятию, изучение материала, предложенного преподавателем.	ПК-18	Файлы с заданиями
3	Выполнение заданий практических работ из практикума, размещённого в ЭУМК: http://www.gup.ru/newumk/manuals/EFHome/CompDept/Struc_Dan_UMK/index.html	ПК-18	Файлы с заданиями
4	Работа над лекционным материалом	ПК-18	Отчёт о выполнении работы. Представленные конспекты лекций.
5	Подготовка к защите исследовательской работы	ПК-18	Отчёт о выполнении работы.
6	Подготовка к контрольной работе (выполнение заданий повышенной сложности, предложенных преподавателем)	ПК-18	Отчёт о выполнении работы, представляемый на итоговом занятии по дисциплине
7	Работа над лекционным материалом. Подготовка к зачёту.	ПК-18	Зачёт

9. Контроль знаний по дисциплине

По дисциплине предусмотрен текущий контроль и промежуточная аттестация.

Текущий контроль успеваемости студента – одна из составляющих оценки качества усвоения образовательных программ. Текущий контроль проводится в течение семестра по мере изучения тем дисциплины.

Промежуточная аттестация проводится по окончании изучения дисциплины в виде **зачёта**. Вопросы к промежуточной аттестации сформулированы в **Оценочных и методических материалах**.

10. Учебно-методическое и информационное обеспечение дисциплины:

а) основная литература:

1. Иванова Г. С. Программирование [Электронный ресурс] : учебник / Г. С. Иванова. - М. : КноРус, 2019. - Режим доступа: <http://www.book.ru/book/931234>

2. Хлебников, А. А. Информационные технологии [Электронный ресурс]: учебник / А. А. Хлебников. – М: КноРус, 2018. – Режим доступа: <https://www.book.ru/book/927689>

б) дополнительная литература:

1. Иванова, Г. С. Технология программирования: учебное пособие / Г. С. Иванова. – М. КноРус, 2016. – Режим доступа: <https://www.book.ru/book/918438>

2. Мельников П.П. Компьютерные технологии в экономике [Текст]: учебное пособие / П. П. Мельников. – М.: КноРус, 2015. – Режим доступа: <https://www.book.ru/book/915920>

3. Методы оптимальных решений (Экономико-математические методы и модели) [Электронный ресурс] : учебное пособие / коллектив авторов ; под ред. С. И. Макарова. - М. : КноРус, 2019. - Режим доступа: <http://www.book.ru/book/929988>

в) Периодические издания

журнал «Вестник Томского государственного педагогического университета» [Электронный ресурс]. Режим доступа: <https://vestnik.tspu.edu.ru/>

г) Лицензионное программное обеспечение

1. Офисный пакет приложений Microsoft Office 2010, в том числе программы MS Word, MS Excel, MS Access, MS PowerPoint.

2. Программа MS Visio 2010.

3. Система программирования Microsoft Visual Studio.

д) Современные профессиональные базы данных и информационные справочные системы

4. Официальный сайт СПбГУП: <http://www.gup.ru/>

5. Электронно-библиотечная система СПбГУП <http://library.gup.ru>

6. Системы поддержки самостоятельной работы СПбГУП: <http://edu.gup.ru/>

7. Справочная правовая система «КонсультантПлюс» (версия ПРОФ), установленная в Университете

8. Российское образование <http://www.edu.ru/>

9. Единое окно доступа к образовательным ресурсам <http://window.edu.ru/>

10. Электронно-библиотечная система <http://e.lanbook.com/>

11. Материально-техническое обеспечение дисциплины

Аудиторный фонд с демонстрационным оборудованием и техническими средствами обучения, учебно-наглядные пособия и методические ресурсы кафедры, фонды Научной библиотеки.

Изучение дисциплины инвалидами и обучающимися с ограниченными возможностями здоровья осуществляется с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья обучающихся.

УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

1. Методические рекомендации по организации самостоятельной работы студентов

Самостоятельная работа в высшем учебном заведении является важной организационной формой индивидуального изучения студентами программного материала. Эти слова особенно актуальны в наше время, когда в педагогике высококвалифицированных специалистов широко используется дистанционное обучение, предполагающее значительную самостоятельную работу студента на основе рекомендаций преподавателя.

2. Методические рекомендации по подготовке к практическим (подгрупповым) занятиям

Подгрупповые занятия – важная форма учебного процесса. Они способствуют закреплению и углублению знаний, полученных студентами на лекциях и в результате самостоятельной работы над научной и учебной литературой и нормативными источниками. Они призваны развивать самостоятельность мышления, умение делать выводы, связывать теоретические положения с практикой, формировать профессиональное правовое сознание будущих бакалавров. На занятиях вырабатываются необходимые каждому выпускнику навыки и умения публично выступать, логика доказывания, культура профессиональной речи. Кроме того, подгрупповые занятия – это средство контроля преподавателей за самостоятельной работой студентов, они непосредственно влияют на уровень подготовки к итоговым формам отчетности – зачётам и экзаменам. В выступлении на семинарском занятии должны содержаться следующие элементы:

- четкое формулирование соответствующего теоретического положения в виде развернутого определения;
- приведение и раскрытие основных черт, признаков, значения и роли изучаемого явления или доказательства определенного теоретического положения;
- подкрепление теоретических положений конкретными фактами.

Для качественного и эффективного изучения дисциплины необходимо овладение навыками работы с книгой, воспитание в себе стремления и привычки получать новые знания из научной и иной специальной литературы. Без этих качеств не может быть настоящего специалиста ни в одной области деятельности.

Читать и изучать, следует, прежде всего, то, что рекомендуется к каждой теме программой, планом семинарских занятий, перечнем рекомендуемой литературы.

Когда студент приступает к самостоятельной работе, то он должен проявить инициативу в поиске специальных источников. Многие новейшие научные положения появляются, прежде всего, в статьях, опубликованных в журналах.

Надо иметь в виду, что в каждом последнем номере издаваемых журналов публикуется библиография всех статей, напечатанных за год, это облегчает поиск нужных научных публикаций.

Работа с научной литературой, в конечном счете, должна привести к выработке у студента умения самостоятельно размышлять о предмете и объекте изучения, которое должно проявляться:

- в ясном и отчетливом понимании основных понятий и суждений, содержащихся в публикации, разработке доказательств, подтверждающих истинность тех или иных положений;

- в понимании студентами обоснованности и целесообразности, приводимых в книге и статье примеров, поясняющих доказательства и выводы автора. При этом будет уместно, если студент самостоятельно приведет дополнительные примеры к этим выводам;
- в отделении основных положений от дополнительных, второстепенных сведений;
- в способности студента критически разобраться в содержании публикации, определить свое отношение к ней в целом, дать ей общую оценку, характеристику.

3. Методические рекомендации по написанию контрольных работ

Важнейшей формой учебной отчётности студента является **контрольная работа**.

Выполнение контрольной работы является промежуточной формой отчётности по изучаемой дисциплине и преследует цель лишь оценить способность студента к самостоятельному поиску источников, формированию содержания и его письменного изложения по указанной проблеме. Это важная составляющая изучения дисциплины, а также эффективная форма контроля знаний. При заочном обучении она выступает как обязательная, основная форма самостоятельной работы. В контрольной работе (в соответствии с учебным планом) студент обязан самостоятельно глубоко разобраться в изучаемых проблемах, усвоить суть темы, уяснить её содержание и только затем письменно представить свою отчётную работу.

Выполнение контрольной работы является одним из условий допуска студента к сдаче экзамена. Работа должна соответствовать установленным требованиям, то есть в ней должны быть раскрыты все проблемы, определённые темой. Для этого студент обязан самостоятельно проанализировать первоисточники и дать исчерпывающие ответы на вопросы темы. Контрольная работа — серьёзное учебное задание, и чтобы написать её как следует, необходимо использовать те первоисточники и учебные пособия, которые позволяют полнее разобраться в проблеме. Студент должен регулярно работать в университетской и городской библиотеке, вдумчиво конспектировать лекции преподавателей.

При написании контрольной работы следует обращать особое внимание на грамотное использование терминологии. При употреблении впервые тех или иных терминов и понятий следует давать их определения либо в самом тексте, либо в сносках.

Приступая к контрольной работе, требуется сначала ознакомиться с имеющейся литературой по теме, изучить первоисточники и составить план. Здесь, в отличие от курсовой работы, план предполагает рассмотрение одной, причём довольно широкой, проблемы, и он может состоять из двух-трёх вопросов. Минимальное количество первоисточников, привлекаемых для написания курсовой работы — пять наименований.

Как правило, контрольные работы по дисциплине сугубо индивидуальны, то есть их тематика персонифицирована. Однако в отдельных случаях темы контрольных работ могут быть адресованы и сразу нескольким, и группе в целом. Таким приёмом преподаватель выявляет степень усвоения какой-то важной учебной проблемы и определяет необходимость проведения дополнительных занятий по какой-либо теме. В настоящее время широко используется методика компьютерного тестирования знаний студентов по дисциплинам, в результате чего появляется возможность быстро проверять знания по наиболее важным темам и объективно оценивать их. Эта форма также может выступать как вид контрольной работы.

В качестве контрольной работы широко применяется самостоятельное изучение монографического исследования по конкретной, крайне важной проблеме, требующей глубокого рассмотрения. Этот вид работы предполагает не простое знакомство с определённым монографическим исследованием, а детальное его изучение. Для этого студенту важно знать некоторые правила работы с первоисточником, которым для него будет являться монография. Следует выяснить фамилию автора, его имя и отчество, учёную степень и звание, а также что побудило его взяться за изучение данной проблемы;

обратить внимание на основные вопросы монографии и их разрешение автором, уметь раскрывать их в ходе собеседования с преподавателем.

Студенту следует письменно (предельно кратко) очертить те вопросы (полностью или частично), которые поставлены автором в монографическом исследовании; при изложении их следует указывать страницы источника.

Задания для написания контрольных работ (для заочной формы обучения)

1. Понятие данных и структуры данных. Классификация типов данных.
2. Составные статические структуры данных.
3. Основные понятия о данных динамической структуры.
4. Стеки: назначение, определение, описание.
5. Очереди: определение, организация.
6. Дек: определение, структура, состояние.
7. Примеры алгоритмов, использующих стек и очередь.
8. Односвязные линейные списки: определение и функциональная спецификация.
9. Базовые операции над элементами списка.
10. Двусвязный линейный список.
11. Выделение и освобождение динамической памяти.
12. Формирование упорядоченного списка.
13. Рекурсия: понятие и примеры использования данных при применении рекурсии.
14. Использование данных в алгоритмах поиска и сортировки (основные понятия).
15. Алгоритм двоичного поиска.
16. Сортировка методом пузырька.
17. Быстрая сортировка методом Хоара.
18. Обзор методов сортировки и поиска.
19. Древовидные структуры: назначение, основные понятия и терминология.
20. Двоичные деревья.
21. Графическое представление древовидной структуры.
22. Дерево как динамическая структура данных.
23. Алгоритм построения двоичного дерева.
24. Деревья двоичного поиска.
25. Алгоритмы обхода двоичного дерева (основные понятия и обзор).
26. Кодирование информации с использованием двоичных деревьев.
27. Рекурсивный алгоритм обхода.
28. Не рекурсивные алгоритмы обхода.
29. Алгоритм поиска по дереву.
30. Дерево сортировки.
31. Удаление из дерева.

Принципы выбора темы работы

Студенты при написании контрольной работы могут выбрать любую из предложенных тем на свое усмотрение.

4. Методические рекомендации по написанию курсовой работы

Курсовая работа учебным планом не предусмотрена.

ОЦЕНОЧНЫЕ И МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ

Оценочные и методические материалы включают в себя:

- перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы;
- показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания;
- типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы;
- методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

№ п\п	Контролируемые темы дисциплины	Код формируемой компетенции	Код и наименование индикатора достижения	Наименование оценочного средства
1.	Введение	ПК-18	<p>ПК-18.1. – Знать основные методы разработки алгоритмов и программ, структуры данных, используемые для представления типовых информационных объектов.</p> <p>ПК-18.2 – Уметь разрабатывать алгоритмы, используя изложенные в курсе общие схемы, методы и приёмы построения алгоритмов, выбирая подходящие структуры данных для представления информационных объектов</p> <p>ПК-18.3. – Владеть инструментарием программирования в системе программирования Microsoft Visual Studio</p>	<p>Опрос</p> <p>Задания для самостоятельной работы</p> <p>Задания для самостоятельной работы</p>

2.	Данные динамической структуры	ПК-18	<p>ПК-18.1. – Знать основные методы разработки алгоритмов и программ, структуры данных, используемые для представления типовых информационных объектов.</p> <p>ПК-18.2 – Уметь разрабатывать алгоритмы, используя изложенные в курсе общие схемы, методы и приёмы построения алгоритмов, выбирая подходящие структуры данных для представления информационных объектов</p> <p>ПК-18.3. – Владеть инструментарием программирования в системе программирования Microsoft Visual Studio</p>	<p>Опрос</p> <p>Задания для самостоятельной работы</p> <p>Задания для самостоятельной работы</p>
3.	Линейные динамические списки	ПК-18	<p>ПК-18.1. – Знать основные методы разработки алгоритмов и программ, структуры данных, используемые для представления типовых информационных объектов.</p> <p>ПК-18.2 – Уметь разрабатывать алгоритмы, используя изложенные в курсе общие схемы, методы и приёмы построения алгоритмов, выбирая</p>	<p>Опрос</p> <p>Задания для самостоятельной работы</p>

			<p>подходящие структуры данных для представления информационных объектов</p> <p>ПК-18.3. – Владеть инструментарием программирования в системе программирования Microsoft Visual Studio</p>	
4.	Рекурсия. Поиск и сортировка	ПК-18	<p>ПК-18.1. – Знать основные методы разработки алгоритмов и программ, структуры данных, используемые для представления типовых информационных объектов.</p> <p>ПК-18.2 – Уметь разрабатывать алгоритмы, используя изложенные в курсе общие схемы, методы и приёмы построения алгоритмов, выбирая подходящие структуры данных для представления информационных объектов</p> <p>ПК-18.3. – Владеть инструментарием программирования в системе программирования Microsoft Visual Studio</p>	<p>Опрос</p> <p>Задания для самостоятельной работы</p> <p>Задания для самостоятельной работы</p>
5.	Древовидные структуры	ПК-18	<p>ПК-18.1. – Знать основные методы разработки алгоритмов и программ, структуры данных, используемые для представления</p>	Опрос

			<p>типовых информационных объектов. ПК-18.2 – Уметь разрабатывать алгоритмы, используя изложенные на лекциях общие схемы, методы и приёмы построения алгоритмов, выбирая подходящие структуры данных для представления информационных объектов ПК-18.3. – Владеть инструментарием программирования в системе программирования Microsoft Visual Studio</p>	<p>Задания для самостоятельной работы</p> <p>Задания для самостоятельной работы</p>
6.	Обзор применения структурированных типов данных	ПК-18	<p>ПК-18.1. – Знать основные методы разработки алгоритмов и программ, структуры данных, используемые для представления типовых информационных объектов. ПК-18.2 – Уметь разрабатывать алгоритмы, используя изложенные в курсе общие схемы, методы и приёмы построения алгоритмов, выбирая подходящие структуры данных для представления информационных объектов ПК-18.3. – Владеть инструментарием программирования в</p>	<p>Опрос</p> <p>Задания для самостоятельной работы</p> <p>Задания для самостоятельной работы</p>

			системе программирования Microsoft Visual Studio	
Результат достижения планируемых результатов изучения дисциплины				Зачет

2. Описание показателей и критериев оценивания компетенций, шкал оценивания

Критерии оценивания (текущий контроль)

1. Оценка «отлично» выставляется студенту, если студент имеет глубокие знания учебного материала по теме практического задания, в логической последовательности излагает материал; смог ответить на все уточняющие и дополнительные вопросы;
2. Оценка «хорошо» выставляется, если студент показал знание учебного материала, смог ответить почти полностью на все заданные дополнительные и уточняющие вопросы;
3. Оценка «удовлетворительно» выставляется, если студент в целом освоил материал; однако, ответил не на все уточняющие и дополнительные вопросы;
4. Оценка «неудовлетворительно» выставляется студенту, если он имеет существенные пробелы в знаниях основного учебного материала по теме практического задания, который полностью не раскрыл содержание вопросов, не смог ответить на уточняющие и дополнительные вопросы.

Критерии оценивания (зачёт).

Знания, умения, навыки и компетенции студентов оцениваются следующими оценками: «зачтено», «не зачтено».

- «зачтено» - студент хорошо и прочно усвоил весь программный материал, исчерпывающе, последовательно, грамотно и логически стройно его излагает, увязывает с практикой, свободно справляется с решением ситуационных задач и тестовыми заданиями, правильно обосновывает принятие решений, умеет самостоятельно обобщать программный материал, не допуская ошибок, знает дополнительную литературу по изучаемой дисциплине.

- «не зачтено» - студент не знает значительной части основного программного материала, в ответах допускает существенные ошибки, не владеет умениями и навыками в выполнении тестовых заданий и решении задач, не способен ответить на дополнительные вопросы.

1. Типовые контрольные задания и методические материалы, процедуры оценивания знаний, умений и навыков

ТЕКУЩИЙ КОНТРОЛЬ

Дискуссионные столы и кейс-задачи в программе не предусмотрены. Текущий контроль осуществляется по результатам выполнения заданий практикума по дисциплине «Структуры данных в предметной области» и заданий для самостоятельной работы.

ПАСПОРТ ТЕСТОВЫХ ЗАДАНИЙ

1. Общее количество тестовых заданий в базе – 100.
2. Ограничение времени выполнения теста (в минутах) – одна попытка, 35 минут.
3. Автоматическое перемешивание вопросов в тесте: - да (нет).
4. Случайный порядок ответов в тестовом задании: - да (нет).
5. Критерии оценки результатов тестирования:
 - Неудовлетворительно – 0 – 55% правильных ответов.
 - Удовлетворительно - 55 – 75% правильных ответов.
 - Хорошо – 75 - 90% правильных ответов

- Отлично – 90% и более правильных ответов

Пример тестовых заданий для текущего контроля представлен ниже:

1. Сколько байтов будет выделено компилятором для хранения следующей структуры:

```
struct d
{
    char c[3];
    int i;
```

- 2;
 - 3;
 - 4;
 - 5;
 - 6;
 - 7.
2. Каким образом можно определить массив указателей на строки:
- char keyword(nkeys);
 - char keyword(*nkeys);
 - char *keyword[nkeys];
 - char keyword.nkeys.
3. Укажите правильный формат записи указателя на структуру date:
- struct date p=d;
 - struct date *p=&d;
 - struct date p=*d;
 - struct date p= =d.
4. Если в программе записано struct node * next, то это означает, что
- задана структура;
 - используется структурная переменная;
 - используется указатель на структуру;
 - используется указатель на массив.
5. С использованием какой команды можно осуществить доступ к элементу структуры:
- st.name = “Петров”;
 - struct node * next ;} st1_node;
 - st1.id = st2.id;
 - struct node { int data; struct node * next; } st1_node;

6. Дано следующее описание переменных:

```
int *p,*q;
char *r;
```

Какие из следующих операторов являются неправильными:

- p=q;
 - q=r;
 - p=NULL;
 - r=NULL;
 - q=*p;
 - *p=NULL.
7. Приведите значение переменной r после выполнения следующего фрагмента

программы:

```
int main ( )
{
int t=2, b=7, r=3;
if (t>b)
{
if (b < r) r=b;
}
else r=t;
return (0);
}
```

- 2;
- 3;
- 4;
- 5;
- 6.

8. Какое значение получит переменная i после выполнения оператора case 4:

```
int i=2;
switch (i)
{
case 1: i += 2;
case 2: i *= 3;
case 0: i /= 2;
case 4: i -= 5;
default;
```

- переменная i примет значение 4, а затем значение -1;
- переменная i примет значение 5, а затем значение -3;
- переменная i примет значение 3, а затем значение -2;
- переменная i примет значение 6, а затем значение -5.

9. Приведите значение переменной r после выполнения следующего фрагмента программы:

```
int main ( )
{
int t=2, b=7, r=3;
if (t>b)
if (b < r) r=b;
else r=t;
return (0);
}
```

- 2;
- 3;
- 4;
- 5;
- 6.

10. Какой оператор отвечает за выбор команды в операторе switch:

- default;
- case;

- break;
- goto;
- return.

11. Чтобы создать простую программу в интегрированной среде из списка типов файлов необходимо выбрать следующий тип файлов:

- Binary File;
- Active Server Page;
- C++ Source File;
- Bitmap File.

12. Какую комбинацию клавиш нужно нажать для компиляции исходного кода программы:

- Shift+F3;
- Ctrl+Alt+Delete;
- Ctrl+F5;
- Alt+F5.

ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ

Вопросы для подготовки к промежуточной аттестации по дисциплине (зачету)

1. Понятие данных и структуры данных.
2. Данные статической и динамической структуры.
3. Простые типы данных.
4. Составные статические структуры данных.
5. Операции над структурами данных: доступ, поиск, вставка, удаление.
6. Разреженные матрицы: основные понятия.
7. Схемы хранения ленточных матриц.
8. Записи, таблицы, строки, множества, их представление в памяти ПК.
9. Основные понятия о данных динамической структуры.
10. Стеки: назначение, определение, описание.
11. Очереди: определение, организация.
12. Дек: определение, структура, состояния.
13. Примеры алгоритмов, использующих стек и очередь.
14. Односвязные линейные списки: определение, использование.
15. Базовые операции над элементами списка.
16. Двусвязный линейный список.
17. Выделение и освобождение динамической памяти.
18. Формирование упорядоченного списка.
19. Рекурсия: понятие и примеры использования.
20. Алгоритмы поиска и сортировки: основные понятия.
21. Алгоритм двоичного поиска.
22. Сортировка методом пузырька.
23. Быстрая сортировка методом Хоара.
24. Обзор методов сортировки и поиска.
25. Древовидные структуры: назначение, основные понятия и терминология.
26. Двоичные деревья.
27. Графическое представление древовидной структуры.
28. Дерево как динамическая структура данных.
29. Алгоритм построения двоичного дерева.
30. Деревья двоичного поиска.
31. Алгоритмы обхода двоичного дерева (основные понятия и обзор).
32. Рекурсивный алгоритм обхода.
33. Не рекурсивные алгоритмы обхода.
34. Алгоритм поиска по дереву.
35. Дерево сортировки.
36. Удаление из дерева.
37. Графы: основные понятия и определения.
38. Матрицы инцидентности и смежности.
39. Алгоритмы на графах: поиски вершин в графах, поиск в глубину.
40. Алгоритм поиска на графах в ширину.

ГЛОССАРИЙ

1. **Абстрактные типы данных** – теоретическая концепция, в которой пытаются полностью освободиться от явного задания представления данных и заменить его на порождение представления по свойствам операций, требуемым для нового типа данных.

2. **Граф** – пара (V, E) , где V – непустое множество объектов некоторой природы, называемых вершинами графа, а E – подмножество двухэлементных подмножеств множества V , называемых ребрами графа.

3. **Данные** – информация, представленная в виде, пригодном для обработки автоматическими средствами при участии человека.

4. **Данные динамической структуры (динамические структуры данных)** – совокупность переменных, количество, размерность или характер взаимосвязей между которыми меняется во время работы программ. Динамические структуры данных базируются на указателях и динамических переменных.

5. **Данные статической структуры** – совокупность фиксированного количества переменных постоянной размерности с неизменным характером связей между ними.

6. **Двоичное (бинарное) дерево** – упорядоченное дерево, каждая вершина которого имеет не более двух поддеревьев.

7. **Двоичное дерево поиска (ДДП)** – дерево, все вершины которого упорядочены, каждая вершина имеет не более двух потомков, и все вершины, кроме корня, имеют родителя. Вершины, не имеющие потомков, называются листьями.

8. **Дек** – линейный список, в котором все добавления и удаления осуществляются на обоих концах списка.

9. **Дерево** – связный граф без циклов.

10. **Динамическая память (куча)** – область памяти компьютера, рассматривая как массив байтов (например, в Turbo Pascal) и занимающая объем приблизительно 300000 байт, позволяющая обрабатывать такие структуры данных, как абстрактные типы данных.

11. **Запись** – структура, связывающая элементы различных типов в один объект. Элементы в записи называются полями.

12. **Кольцевой список** – список, в котором не содержится пустых указателей, а последний элемент ссылается на первый.

13. **Ленточная матрица** – разреженная матрица с ленточной структурой ненулевых элементов. Для ленточных матриц вводят понятие ширины ленты.

14. **Линейный список** – динамическая структура данных, каждый элемент которой посредством указателя связывается со следующим элементом. Типовые операции: добавление нового узла, удаление узла, объединение узлов, разбиение списка на два или более, копирование списка, сортировка элементов списка, поиск элемента в списке.

15. **Матрица инцидентности графа (V, E)** – матрица с $|V|$ строками, соответствующих вершинам, и $|E|$ столбцами, соответствующих ребрам (или дугам графа).

16. **Матрица смежности неориентированного графа** – это матрица $B = [B_{ij}]$, где $B_{ij}=1$, если существует ребро, идущее из вершины i в вершину j .

17. **Матрица смежности ориентированного графа** – это матрица $B = [B_{ij}]$, где $B_{ij}=1$, если существует дуга, идущая из вершины i в вершину j .

18. **Метод сортировки Хоара** – другие названия этого алгоритма звучат как «быстрая сортировка» или «быстрая рекурсивная сортировка».

19. **Множество** – структура данных, определяемая как $\text{type } T = \text{set of } T_0$ (в языке Pascal). Значениями переменной x типа T являются множества элементов типа T_0 .

20. **Оператор** – единица выполнения программы. Программа состоит из последовательности операторов. Выделяют два типа операторов: составной оператор (последовательность операторов, заключенная в фигурные скобки) и блок (совокупность описаний и операторов, заключенных в фигурные скобки).

21. **Очередь** – структура данных, сохраняющая элементы в списке и обеспечивающая доступ к данным только в двух концах списка. Элементы удаляются из очереди в том же порядке, в котором они добавляются (FIFO – first in/first out).

22. **Простые типы данных (для языка С)** – целое, символьное, вещественное числа, число с плавающей точкой, типы перечисления, логический тип, указатель.

23. **Разреженный массив, или разреженная матрица**, – это массив, в котором не все элементы используются, имеются в наличии или нужны в данный момент. Разреженные массивы полезны в тех случаях, когда выполняются два условия: размер массива, который требуется приложению, достаточно большой, и когда не все элементы массива используются.

24. **Рекурсия** – фундаментальное понятие в математике и компьютерных науках. В языках программирования рекурсивной программой называется программа, которая обращается сама к себе. Рекурсивная программа не может вызывать себя до бесконечности, следовательно, вторая важная особенность рекурсивной программы – наличие условия завершения, позволяющее программе прекратить вызывать себя.

25. **Связный список** – базовая динамическая структура данных, состоящая из узлов, каждый из которых содержит как собственно данные, так и одну или две ссылки («связки») на следующий и/или предыдущий узел списка. Односвязный список – ссылка в каждом узле указывает на следующий узел в списке. Двусвязный список – ссылки в каждом узле указывают на предыдущий и последующий узлы списка.

26. **Сортировка** – алгоритмический процесс перестановки объектов данной последовательности в определенном заданном порядке. Цель сортировки – облегчение поиска элементов в отсортированной последовательности.

27. **Сортировка включения (вставками)** – метод сортировки, в котором из неупорядоченной последовательности поочередно выбирается каждый элемент, сравнивается с предыдущим (уже упорядоченным) элементом последовательности и помещается на соответствующее место в последнем.

28. **Сортировка выбором** – метод сортировки, при котором в неупорядоченной последовательности выбирается минимальный элемент, который исключается из дальнейшей обработки, а оставшаяся последовательность принимается за исходную. Процесс повторяется до тех пор, пока все элементы не будут выбраны.

29. **Сортировка обменом (метод пузырька)** – метод сортировки, при котором все соседние элементы массива попарно сравниваются друг с другом и меняются местами в том случае, если предшествующий элемент больше последующего.

30. **Составные статические структуры данных** – массивы, записи, структуры, файлы, абстрактные типы данных и классы, стеки и очереди, параметризованные типы данных, списки, деревья, классы.

31. **Стек** – линейный список, в котором все добавления производятся на одном конце списка, а удаления на его противоположном конце.

32. **Строки** – фундаментальный тип данных, предназначенный для ввода, обработки и вывода слов в алфавитах.

33. **Структура данных** – совокупность элементов данных, между которыми существуют некоторые отношения, причем элементами данных могут быть простые данные и структуры данных.

34. **Структура данных** – способ организации данных, применяющийся для решения задачи.

35. **Тип данных** – множество значений, набор операций, которые можно применять к таким значениям, и, возможно, способ реализации хранения значений и выполнения операций.

36. **Указатель** – переменная, значением которой является целое число, являющееся адресом некоторого программного объекта.

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ПРЕПОДАВАТЕЛЯ ПО ДИСЦИПЛИНЕ

Целью дисциплины «Структуры данных в предметной области» является изучение структур данных, которые используются при программировании решения задач из предметной области указанной специальности и освоение алгоритмов обработки данных (на примерах из профессиональной деятельности).

Дисциплина изучает основные структуры данных и алгоритмы, а также способы их использования при программировании на языке C++.

Форма промежуточной аттестации знаний – **зачёт**.

Методические принципы и приёмы построения учебной дисциплины сетевая экономика. Ключевым методическим способом подачи учебного материала по дисциплине «Структуры данных в предметной области» является лекция.

Лекционное занятие – это систематическое, последовательное, устное изложение лектором учебного материала. Занятие «лекция» носит, прежде всего, обзорный характер, охватывая весь круг выносимых на изучение учебных вопросов. При проведении такого типа занятий очень важно живое слово лектора, его педагогическое мастерство как педагога, который даёт студентам информационную базу. Лекции являются важной формой передачи преподавателем студентам общетеоретических знаний.

Лекции, как правило, читаются не по всем, а по наиболее сложным темам курса, не дублируют учебники, а содержат новейшие научные данные и примеры, которых может не быть в учебных пособиях. Для лучшего усвоения материала на лекционных занятиях целесообразно предварительно перед лекцией ознакомиться с положениями лекционной темы в конспекте лекций, содержащемся в данном учебно-методическом пособии либо в рекомендуемых учебниках.

Подгрупповые (лабораторные) занятия – другая важная форма учебного процесса. Они способствуют закреплению и углублению знаний, полученных студентами на лекциях и в результате самостоятельной работы над научной и учебной литературой и нормативными источниками. Они призваны развивать самостоятельность мышления, умение делать выводы, связывать теоретические положения с практикой, формировать профессиональное сознание. На занятиях вырабатываются необходимые каждому бакалавру навыки и умения публично выступать, логика доказывания, культура профессиональной речи. Кроме того, семинары – это средство контроля преподавателей за самостоятельной работой студентов, они непосредственно влияют на уровень подготовки к итоговым формам отчётности – зачётам и экзаменам. В выступлении на семинарском занятии должны содержаться следующие элементы:

– чёткое формулирование соответствующего теоретического положения в виде развёрнутого определения;

– приведение и раскрытие основных черт, признаков, значения и роли изучаемого явления или доказательства определённого теоретического положения;

– подкрепление теоретических положений конкретными фактами.

Для качественного и эффективного изучения актуальных проблем теории необходимо овладение навыками работы с книгой, воспитание в себе стремления и привычки получать новые знания из научной и иной специальной литературы. Без этих качеств не может быть настоящего специалиста ни в одной области деятельности.

Читать и изучать, следует, прежде всего, то, что рекомендуется к каждой теме программой, планом семинарских занятий, перечнем рекомендуемой литературы.

Когда студент приступает к самостоятельной работе, то он должен проявить инициативу в поиске специальных источников. Надо иметь в виду, что в каждом последнем номере издаваемых журналов публикуется библиография всех статей, напечатанных за год, это облегчает поиск нужных научных публикаций.

Работа с научной литературой, в конечном счёте, должна привести к выработке у бакалавра умения самостоятельно размышлять о предмете и объекте изучения, которое должно проявляться:

- в ясном и отчётливом понимании основных понятий и суждений, содержащихся в публикации, разработке доказательств, подтверждающих истинность тех или иных положений;

- в понимании студентами обоснованности и целесообразности, приводимых в книге и статье примеров, поясняющих доказательства и выводы автора. При этом будет уместно, если бакалавр самостоятельно приведёт дополнительные примеры к этим выводам;

- в отделении основных положений от дополнительных, второстепенных сведений;

- в способности студента критически разобраться в содержании публикации, определить своё отношение к ней в целом, дать ей общую оценку, характеристику.

Другим важнейшим методическим приёмом в учебном процессе является самостоятельная работа студента.

Самостоятельная работа в высшем учебном заведении, является важной организационной формой индивидуального изучения студентами программного материала.

В современных условиях дидактическое значение самостоятельной подготовки неизмеримо возрастает, а её цели состоят в том, чтобы:

- повысить ответственность самих обучаемых за свою профессиональную подготовку, сформировать в себе личностные и профессионально-деловые качества;

- научить студентов самостоятельно приобретать знания, формировать навыки и умения, необходимы для профессиональной деятельности;

- развивать в себе самостоятельность в организации, планировании и выполнении заданий, определяемых учебным планом и указаниями преподавателя.

Достигнуть этих целей в ходе самостоятельной работы при изучении дисциплины возможно только при хорошей личной организации своего учебного труда, умении использовать все резервы имеющегося времени и подчинить их профессиональной подготовке.

Самостоятельная работа как метод обучения включает:

- изучение и конспектирование обязательной литературы в соответствии с программой дисциплины;

- ознакомление с литературой, рекомендованной в качестве дополнительной;

- выполнение заданий лабораторных работ (разработка программ на языке программирования С++, реализующих рассматриваемые алгоритмы обработки информации) и подготовку отчётов о ходе их выполнения;

- изучение и осмысление специальной терминологии и понятий предметной области дисциплины;

- изучение и отработка нормативных актов, комментариев к ним, проведение сравнительного анализа с предыдущим;

- сбор материала и написание контрольных, конкурсных и дипломных работ;

- изучение указанной литературы для подготовки к зачёту.

Основными компонентами содержания данного вида работы являются:

- творческое изучение учебных пособий и научной литературы;

- разработка программ и анализ результативности их выполнения;

- подготовка отчётов по результатам выполнения лабораторных работ;

- выполнение исследовательских заданий по предложенным темам;

- умелое конспектирование;

- участие в различных формах учебного процесса, научных конференциях, в работе кружков и т. д.;

- получение консультаций у преподавателя по отдельным проблемам курса;
- получение информации и опыта о работе профессионалов в процессе производственно-учебной практики;
- знакомство со специальной литературой при формировании своей личной библиотеки и др.

Данный комплекс рекомендаций позволяет студентам овладеть многими важными приёмами самостоятельной работы и успешно использовать их при подготовке контрольных по дисциплине.

Важнейшей формой учебной отчётности студента является **контрольная работа**. Выполнение контрольной работы является промежуточной формой отчётности по изучаемой дисциплине и преследует цель лишь оценить способность студента к самостоятельному поиску источников, формированию содержания и его письменного изложения по указанной проблеме. Это важная составляющая изучения дисциплины, а также эффективная форма контроля знаний. При заочном обучении она выступает как обязательная, основная форма самостоятельной работы. В контрольной работе (в соответствии с учебным планом) студент обязан самостоятельно глубоко разобраться в изучаемых проблемах, усвоить суть темы, уяснить её содержание и только затем письменно представить свою отчётную работу.

Выполнение контрольной работы является одним из условий допуска студента к сдаче зачёта. Работа должна соответствовать установленным требованиям, то есть в ней должны быть раскрыты все проблемы, определённые темой. Для этого студент обязан самостоятельно проанализировать первоисточники и дать исчерпывающие ответы на вопросы темы. Контрольная работа – серьёзное учебное задание, и чтобы написать её как следует, необходимо использовать те первоисточники и учебные пособия, которые позволяют полнее разобраться в проблеме. Бакалавр должен регулярно работать в университетской и городской библиотеке, вдумчиво конспектировать лекции преподавателей.

При написании контрольной работы следует обращать особое внимание на грамотное использование специальной терминологии. При употреблении впервые тех или иных терминов и понятий следует давать их определения либо в самом тексте, либо в сносках.

Приступая к контрольной работе, требуется сначала ознакомиться с имеющейся литературой по теме, изучить первоисточники и составить план. Здесь, в отличие от курсовой работы, план предполагает рассмотрение одной, причём довольно широкой, проблемы, и он может состоять из двух-трёх вопросов. Минимальное количество первоисточников, привлекаемых для написания контрольной работы – пять наименований.

Контрольные работы могут выступать как дополнительные (вспомогательные) учебные формы отчётности студента, которые осуществляются в ходе семинарских (практических) занятий (в конце) и проводятся максимум в течение 10-15 минут. Преподаватель может заранее объявить о предстоящей работе и предложить примерный перечень тем, то есть сориентировать студентов на работу по более широкому кругу вопросов. Таким образом, бакалаврам даётся возможность лишний раз обратиться к учебному материалу и более качественно подготовиться к выполнению контрольной работы.

Как правило, контрольные работы по дисциплине сугубо индивидуальны, то есть их тематика персонифицирована. Однако в отдельных случаях темы контрольных работ могут быть адресованы и сразу нескольким бакалаврам, и группе в целом. Таким приёмом преподаватель выявляет степень усвоения какой-то важной учебной проблемы и определяет необходимость проведения дополнительных занятий по какой-либо теме.

В качестве контрольной работы широко применяется самостоятельное изучение монографического исследования по конкретной, крайне важной проблеме, требующей

глубокого рассмотрения. Этот вид работы предполагает не простое знакомство с определённым монографическим исследованием, а детальное его изучение. Для этого студенту важно знать некоторые правила работы с первоисточником, которым для него будет являться монография. Следует выяснить фамилию автора, его имя и отчество, учёную степень и звание, а также что побудило его взяться за изучение данной проблемы; обратить внимание на основные вопросы монографии и их разрешение автором, уметь раскрывать их в ходе собеседования с преподавателем.

Студенту следует письменно (предельно кратко) очертить те вопросы (полностью или частично), которые поставлены автором в монографическом исследовании; при изложении их следует указывать страницы источника.

Важными в методическом плане на семинарских занятиях являются проводимые **тестовые опросы** и решение задач, которые содействуют превращению знаний в глубокие убеждения, дают простор для развития творческо-эмоциональной сферы, позволяют сделать выводы об эффективности занятий с учащимися, что в итоге повышает интерес к овладению знаниями.

Только сочетая дидактически и органически все методические способы и приёмы в их диалектическом единстве и взаимосвязи, мы можем добиться должного уяснения учебного материала со стороны студентов.

Методические рекомендации для преподавателей

Тема занятия	Виды учебных занятий	Способы учебной деятельности	Методы обучения, формы педагогического общения	Средства обучения	Формы контроля
1	2	3	4	5	6
Введение	Лекция.	Коллективный	Лекция, рассказ объяснительно-иллюстративный, монолог, внешний диалог, дискуссия.	Ноутбук, проектор, компьютеры с установленным программным обеспечением, презентация, электронный курс по дисциплине	Аттестация в компьютерном классе (по университетскому графику), проверка заданий для самостоятельного выполнения, зачёт.
Данные динамической структуры	Лекция, практическое занятие	Коллективный	Лекция, рассказ объяснительно-иллюстративный, монолог, внешний диалог частично-поисковый, дискуссия.	Ноутбук, проектор, компьютеры с установленным программным обеспечением, презентация, электронный курс по дисциплине	Аттестация в компьютерном классе (по университетскому графику), проверка заданий для самостоятельного выполнения, зачёт.
Линейные динамические списки	Лекция, практическое занятие	Коллективный	Лекция, рассказ объяснительно-иллюстративный, монолог, внешний диалог частично-поисковый, дискуссия.	Ноутбук, проектор, компьютеры с установленным программным обеспечением, презентация, электронный курс по дисциплине	Аттестация в компьютерном классе (по университетскому графику), проверка заданий для самостоятельного выполнения, зачёт.
Рекурсия. Поиск и сортировка	Лекция, практическое занятие	Коллективный	Лекция, рассказ объяснительно-иллюстративный, монолог, внешний диалог частично-поисковый, дискуссия.	Ноутбук, проектор, компьютеры с установленным программным обеспечением, презентация, электронный курс по дисциплине	Аттестация в компьютерном классе (по университетскому графику), проверка заданий для самостоятельного выполнения, зачёт.
Древовидные структуры	Лекция, практическое занятие	Коллективный	Лекция, рассказ объяснительно-иллюстративный, монолог, внешний диалог частично-поисковый, дискуссия.	Ноутбук, проектор, компьютеры с установленным программным обеспечением, презентация, электронный курс по дисциплине	Аттестация в компьютерном классе (по университетскому графику), проверка заданий для самостоятельного выполнения, зачёт.
Обзор применения структурированных типов данных	Лекция, практическое занятие	Коллективный	Лекция, рассказ объяснительно-иллюстративный, монолог, внешний диалог частично-поисковый, дискуссия.	Ноутбук, проектор, компьютеры с установленным программным обеспечением, презентация, электронный курс по дисциплине	Аттестация в компьютерном классе (по университетскому графику), проверка заданий для самостоятельного выполнения, зачёт.

**Тематический план изучения дисциплины
«Структуры данных в предметной области»**

Годы набора с 2020, форма обучения очная

Наименование разделов и тем	Всего	Трудоемкость по дисциплине				Формируемые компетенции	
		Контактная работа	в т.ч.				СР
			Лекции	Подгр/Лаб.	Пр/Сем		
Введение	22	8	2	6	0	14	ПК-18
Данные динамической структуры	20	10	4	6	0	10	ПК-18
Линейные динамические списки	20	10	4	6	0	10	ПК-18
Рекурсия. Поиск и сортировка	19	10	4	6	0	9	ПК-18
Древовидные структуры	14	8	2	6	0	6	ПК-18
Обзор применения структурированных типов данных	13	8	2	6	0	5	ПК-18
Зачёт	+						
Итого по дисциплине	108	54	18	36	0	54	
Зачетных единиц	3						

**Тематический план изучения дисциплины
«Структуры данных в предметной области»**

Годы набора с 2020 форма обучения заочная

Наименование разделов и тем	Всего	Трудоемкость по дисциплине					Формируемые компетенции
		Контактная работа	в т.ч.			СР	
			лекции	Подгр/ Лаб.	Пр/ Сем		
Введение	6	2	2	0	0	4	ПК-18
Данные динамической структуры	12	4	2	2	0	8	ПК-18
Линейные динамические списки	34	2	0	2	0	32	ПК-18
Рекурсия. Поиск и сортировка	20	0	0	0	0	20	ПК-18
Древовидные структуры	22	2	0	2	0	20	ПК-18
Обзор применения структурированных типов данных	10	0	0	0	0	10	ПК-18
Контроль	4						ПК-18
Итого по дисциплине	108	10	4	6	0	94	
Зачетных единиц	3						
Контрольная работа	+						