МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение высшего образования
"САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ"

Кафедра № 44

УТВЕРЖДАЮ
Руководитель образовательной программы
старший преподаватель
(должность, уч. степень, звание)
Д.В. Куртяник
(инициалы, фамилия)
(подпись)
«16» февраля 2026 г

# РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Системное программное обеспечение»
(Наименование дисциплины)

| | |
|---|---|
| Код направления подготовки/ специальности | 09.03.01 |
| Наименование направления подготовки/ специальности | Информатика и вычислительная техника |
| Наименование направленности/ специализации | Компьютерные технологии, системы и сети |
| Форма обучения | заочная |
| Год приема | 2026 |

Санкт-Петербург – 2026

Лист согласования рабочей программы дисциплины

Программу составил (а)

| доц., к.т.н., доц. | «16» февраля 2026 г | А.Ю. Молчанов |
|---|---|---|
| (должность, уч. степень, звание) | (подпись, дата) | (инициалы, фамилия) |

Программа одобрена на заседании кафедры № 44
«16» февраля 2026 г, протокол № 5-25/26

Заведующий кафедрой № 44

| д.т.н.,проф. | «16» февраля 2026 г | М.Б. Сергеев |
|---|---|---|
| (уч. степень, звание) | (подпись, дата) | (инициалы, фамилия) |

Заместитель директора института №4 по методической работе

| доц.,к.т.н. | «16» февраля 2026 г | А.А. Фоменкова |
|---|---|---|
| (должность, уч. степень, звание) | (подпись, дата) | (инициалы, фамилия) |

Аннотация

Дисциплина «Системное программное обеспечение» входит в образовательную программу высшего образования – программу бакалавриата по направлению подготовки/ специальности 09.03.01 «Информатика и вычислительная техника» направленности/специализации «Компьютерные технологии, системы и сети». Дисциплина реализуется кафедрой «№44».

Дисциплина нацелена на формирование у выпускника следующих компетенций:

ПК-4 «Способен проектировать программное обеспечение»

Содержание дисциплины охватывает круг вопросов, связанных с теоретическим базисом, необходимым для создания и использования языков программирования высокого уровня (грамматики и распознаватели для формальных языков), созданием программных продуктов с помощью современных систем программирования, организацией взаимодействия между различными программными продуктами, а также технологиями разработки программного обеспечения для сетевых распределенных вычислительных систем с использованием современных операционных систем (ОС) и систем управления базами данных (СУБД).

Преподавание дисциплины предусматривает следующие формы организации учебного процесса: лекции, лабораторные работы, самостоятельная работа обучающихся.

Программой дисциплины предусмотрены следующие виды контроля: текущий контроль успеваемости, промежуточная аттестация в форме дифференцированного зачета (7 семестр), экзамена (8 семестр).

Общая трудоемкость освоения дисциплины составляет 6 зачетных единиц, 216 часов.

Язык обучения по дисциплине «русский»

1. Перечень планируемых результатов обучения по дисциплине

1.1. Цели преподавания дисциплины

Цель преподавания дисциплины – получение студентами необходимых знаний в области теоретических основ и получение практических навыков описания формальных языков, методов реализации трансляторов, принципов построения и функционирования операционных систем, технологий разработки программного обеспечения для распределенных вычислительных систем, технологий обеспечения взаимодействия программ в современных вычислительных системах.

1.2. Дисциплина входит в состав части, формируемой участниками образовательных отношений, образовательной программы высшего образования (далее – ОП ВО).

1.3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ОП ВО.

В результате изучения дисциплины обучающийся должен обладать следующими компетенциями или их частями. Компетенции и индикаторы их достижения приведены в таблице 1.

Таблица 1 – Перечень компетенций и индикаторов их достижения

| Категория (группа) компетенции | Код и наименование компетенции | Код и наименование индикатора достижения компетенции |
|---|---|---|
| Профессиональные компетенции | ПК-4 Способен проектировать программное обеспечение | ПК-4.З.1 знать принципы построения и виды архитектуры программного обеспечения<br>ПК-4.З.2 знать методы и средства проектирования программного обеспечения<br>ПК-4.У.1 уметь использовать существующие типовые решения и шаблоны проектирования программного обеспечения<br>ПК-4.В.1 владеть навыками проектирования структур данных, баз данных, программных интерфейсов |

2. Место дисциплины в структуре ОП

Дисциплина может базироваться на знаниях, ранее приобретенных обучающимися при изучении следующих дисциплин:

- «Математика. Математический анализ.»
- «Информатика.»
- «Основы программирования.»
- «Операционные системы.»

Знания, полученные при изучении материала данной дисциплины, имеют как самостоятельное значение, так и используются при изучении других дисциплин:

- «Интерфейсы периферийных устройств.»

3. Объем и трудоемкость дисциплины

Данные об общем объеме дисциплины, трудоемкости отдельных видов учебной работы по дисциплине (и распределение этой трудоемкости по семестрам) представлены в таблице 2.

Таблица 2 – Объем и трудоемкость дисциплины

| Вид учебной работы | Всего | Трудоемкость по семестрам | |
|---|---|---|---|
| | | №7 | №8 |
| 1 | 2 | 3 | 4 |
| ***Общая трудоемкость дисциплины***, | 6/ 216 | 2/ 72 | 4/ 144 |

| | | | |
|---|---|---|---|
| ЗЕ/ (час) | | | |
| ***Из них часов практической подготовки*** | 20 | 10 | 10 |
| ***Аудиторные занятия***, всего час. | 40 | 20 | 20 |
| в том числе: | | | |
| лекции (Л), (час) | 20 | 10 | 10 |
| практические/семинарские занятия (ПЗ), (час) | | | |
| лабораторные работы (ЛР), (час) | 20 | 10 | 10 |
| курсовой проект (работа) (КП, КР), (час) | | | |
| экзамен, (час) | 9 | | 9 |
| ***Самостоятельная работа***, всего (час) | 167 | 52 | 115 |
| ***Вид промежуточной аттестации*:** зачет, дифф. зачет, экзамен (Зачет, Дифф. зач, Экз.) | Дифф. зач., Экз., | Дифф. зач., | Экз., |

## 4. Содержание дисциплины

4.1. Распределение трудоемкости дисциплины по разделам и видам занятий. Разделы, темы дисциплины и их трудоемкость приведены в таблице 3.

Таблица 3 – Разделы, темы дисциплины, их трудоемкость

| Разделы, темы дисциплины | Лекции (час) | ПЗ (СЗ) (час) | ЛР (час) | КП (час) | СРС (час) |
|---|---|---|---|---|---|
| Семестр 7 | | | | | |
| Раздел 1. Общие принципы построения трансляторов. Теория формальных языков и грамматик.<br>Тема 1.1. Трансляторы и компиляторы. Основные понятия. Этапы компиляции.<br>Тема 1.2. Языки и грамматики. Формальное определение грамматик. Классификация языков и грамматик.<br>Тема 1.3. Распознаватели и преобразователи. Основные компоненты распознавателей. Классификация распознавателей.<br>Тема 1.4. Таблицы идентификаторов. Основные методы организации таблиц идентификаторов. | 2 | | 2 | | 10 |
| Раздел 2. Лексический анализ и регулярные языки.<br>Тема 2.1. Лексический анализ. Назначение и функции лексического анализа.<br>Тема 2.2. Регулярные языки и грамматики. Автоматные грамматики.<br>Тема 2.3. Конечные автоматы (КА). Использование КА в качестве лексического анализатора.<br>Тема 2.4. Регулярные множества и регулярные выражения.<br>Тема 2.5. Свойства регулярных языков. | 3 | | 4 | | 12 |
| Раздел 3. Синтаксический анализ и контекстно-свободные языки<br>Тема 3.1. Синтаксический анализ. Назначение и функции синтаксического анализа.<br>Тема 3.2. Контекстно-свободные языки и грамматики. | 3 | | 4 | | 16 |

| Разделы, темы дисциплины | Лекции (час) | ПЗ (СЗ) (час) | ЛР (час) | КП (час) | СРС (час) |
|---|---|---|---|---|---|
| Тема 3.3. Автоматы с магазинной памятью (МП-автоматы). Расширенные МП-автоматы.<br>Тема 3.4. Построение синтаксического распознавателя по заданной грамматике.<br>Тема 3.5. Линейные синтаксические распознаватели.<br>Тема 3.6. Свойства КС-языков.<br>Тема 3.7. Внутренние представления программ. | | | | | |
| Раздел 4. Особенности построения современных компиляторов<br>Тема 4.1. Проход. Понятие прохода. Однопроходные и многопроходные компиляторы.<br>Тема 4.2. Язык ассемблера. Особенности компиляторов с языка ассемблера.<br>Тема 4.3. Макрогенерация и макроподстановки. Условная компиляция.<br>Тема 4.3. Мобильность и переносимость программного обеспечения. Особенности компиляторов с языков Java и C#. | 2 | | 0 | | 14 |
| Итого в семестре: | 10 | | 10 | | 52 |
| Семестр 8 | | | | | |
| Раздел 5. Генерирование и оптимизация результирующего кода<br>Тема 5.1. Семантический анализ. Основные функции семантического анализа.<br>Тема 5.2. Распределение памяти. Организация компилятором таблиц RTTI.<br>Тема 5.3. Исключительные ситуации. Обработка исключительных ситуаций.<br>Тема 5.4. Синтаксически управляемый перевод. Генерация результирующего кода.<br>Тема 5.5. Оптимизация. Основные принципы и методы оптимизации. | 2 | | 4 | | 25 |
| Раздел 6. Современные системы программирования<br>Тема 6.1. Структура систем программирования.<br>Тема 6.2. История развития систем программирования.<br>Тема 6.3. Функции основных компонент в составе систем программирования. | 2 | | 0 | | 20 |
| Раздел 7. Технологии разработки программного обеспечения<br>Тема 7.1. Принципы организации распределенных вычислений.<br>Тема 7.2. Технология «файл-сервер».<br>Тема 7.3. Терминальный сервер.<br>Тема 7.4. Технология «клиент-сервер».<br>Тема 7.5. Трехуровневые и многоуровневые технологии разработки программного обеспечения.<br>Тема 7.6. Облачные вычисления и сервисы. | 3 | | 6 | | 28 |

| Разделы, темы дисциплины | Лекции (час) | ПЗ (СЗ) (час) | ЛР (час) | КП (час) | СРС (час) |
|---|---|---|---|---|---|
| Раздел 8. Разработка программного обеспечения для глобальных сетей<br>Тема 8.1. Особенности разработки программного обеспечения для глобальных сетей. Применение современных интерпретаторов.<br>Тема 8.2. Построение статических Web-страниц. Язык HTML.<br>Тема 8.3. Построение динамических Web-страниц на основе компилируемых и интерпретируемых языков.<br>Тема 8.4. Построение интерактивных Web-страниц. | 3 | | 0 | | 42 |
| Итого в семестре: | 10 | | 10 | | 115 |
| Итого | 20 | 0 | 20 | 0 | 167 |

Практическая подготовка заключается в непосредственном выполнении обучающимися определенных трудовых функций, связанных с будущей профессиональной деятельностью.

4.2. Содержание разделов и тем лекционных занятий.

Содержание разделов и тем лекционных занятий приведено в таблице 4.

Таблица 4 – Содержание разделов и тем лекционного цикла

| Номер раздела | Название и содержание разделов и тем лекционных занятий |
|---|---|
| **1** | Раздел 1. Общие принципы построения трансляторов. Теория формальных языков и грамматик.<br>Тема 1.1. Трансляторы и компиляторы. Основные понятия. Этапы компиляции.<br>Определение транслятора, компилятора, интерпретатора. Различия между транслятором и компилятором, транслятором и интерпретатором. Понятие объектной программы и целевой вычислительной системы. Типовая структура компилятора. Этапы компиляции (этапы анализа и этапы синтеза), взаимосвязь между ними.<br>Тема 1.2. Языки и грамматики. Формальное определение грамматик. Классификация языков и грамматик.<br>Понятие цепочки символов (строки), основные операции над цепочками символов. Определение алфавита, цепочки символов над алфавитом, множества всех возможных цепочек символов над заданным алфавитом. Формальное определение языка, способы задания языков. Формальное определение грамматики. Грамматика в форме Бэкуса-Наура (БНФ), пример грамматики в БНФ. Другие способы определения грамматик, в том числе форма грамматики с метасимволами и графическая форма представления грамматик. Непосредственная выводимость цепочки символов из правил грамматики. Понятие вывода и цепочки вывода, сентенциальная форма грамматики. Дерево вывода, правосторонний и левосторонний выводы. Примеры цепочек вывода, дерева вывода. Определение языка, заданного грамматикой. Однозначные и неоднозначные грамматики, проблема однозначности. Примеры однозначных и неоднозначных грамматик, неоднозначность естественных языков. Классификация языков и грамматик по Хомскому, проблема преобразования. Назначение основных типов языков и их применение в вычислительной технике. |

| Номер раздела | Название и содержание разделов и тем лекционных занятий |
|---|---|
| | Тема 1.3. Распознаватели и преобразователи. Основные компоненты распознавателей. Классификация распознавателей.<br>Определение распознавателя. Основные компоненты распознавателей: считывающее устройство, устройство управления (УУ), внешняя память (ВП). Основные принципы работы распознавателя, начальная конфигурация и конечные конфигурации распознавателя. Классификация распознавателей по считывающему устройству, односторонние и двусторонние распознаватели. Классификация распознавателей по УУ, детерминированные и недетерминированные распознаватели. Классификация распознавателей по ВП, распознаватели с неограниченной внешней памятью (в том числе машина Тьюринга), распознаватели с ограниченной внешней памятью, распознаватели без внешней памяти. Соответствие между классификациями распознавателей и классификацией языков по Хомскому, типы распознавателей для основных типов языков. Определение преобразователя, отличие преобразователя от распознавателя. Назначение распознавателей и преобразователей, задача разбора и ее решение для основных типов языков.<br>Тема 1.4. Таблицы идентификаторов. Основные методы организации таблиц идентификаторов.<br>Назначение таблиц идентификаторов, информация, хранимая в таблицах идентификаторов. Основные действия, выполняемые компилятором над таблицами идентификаторов, требования к организации таблиц идентификаторов. Простейшие способы организации таблиц идентификаторов – логарифмический поиск в линейном списке и бинарное дерево – их основные преимущества и недостатки. Понятие о хеш-функции. Организация таблиц идентификаторов на основе хеш-функции с помощью хеш-адресации. Проблемы, связанные с организацией таблиц идентификаторов на основе хеш-адресации, понятие о коллизии. Методы борьбы с коллизиями. Организация таблиц идентификатором на основе рехеширования, основные преимущества и недостатки метода. Организация таблиц идентификаторов с помощью метода цепочек, основные преимущества и недостатки метода. Смешанные методы организации таблиц идентификаторов. |
| **2** | Тема 2.1. Лексический анализ. Назначение и функции лексического анализа.<br>Определение лексемы, основные типы лексем в языках программирования. Задачи лексического анализа, функции, выполняемые лексическим анализатором. Преимущества, предоставляемые лексическим анализатором, возможность организации компиляторов без лексического анализатора. Проблема определения границ лексем, взаимодействие лексического и синтаксического анализаторов – последовательный и параллельный методы взаимодействия. Принцип выбора лексемы максимальной длины в современных компиляторах. Таблица лексем и ее назначение, информация, хранимая в таблице лексем. Пример построения таблицы лексем.<br>Тема 2.2. Регулярные языки и грамматики. Автоматные грамматики.<br>Регулярные языки – простейший тип языков по классификации Хомского. Определение регулярной грамматики, леволинейные и |

| Номер раздела | Название и содержание разделов и тем лекционных занятий |
|---|---|
| | праволинейные грамматики. Задача разбора для регулярных языков и метод ее решения. Определение автоматной грамматики. Эквивалентность регулярных и автоматных грамматик, алгоритм построения автоматной грамматики на основе регулярной грамматики. Пример регулярной грамматики и ее преобразование к автоматному виду.<br>Тема 2.3. Конечные автоматы (КА). Использование КА в качестве лексического анализатора.<br>Определение КА. Способы задания КА: с помощью множеств и функции переходов, с помощью графа переходов. КА как распознаватель для регулярных языков, эквивалентность КА и регулярных языков. Детерминированные и недетерминированные КА, преобразование КА к детерминированному виду. Минимизация КА, множества эквивалентных состояний КА, алгоритм минимизации КА. Полное решение задачи разбора для регулярных языков, эквивалентность регулярных грамматик, автоматных грамматик и КА. Алгоритм построения КА по заданной автоматной грамматике. Пример автоматной грамматики, построение по ней КА, приведение КА к детерминированному виду и минимизация построенного КА.<br>Тема 2.4. Регулярные множества и регулярные выражения.<br>Определение регулярного множества и регулярного выражения. Свойства регулярных выражений. Уравнения с регулярными коэффициентами и системы уравнений с регулярными коэффициентами, методы решения систем уравнений с регулярными коэффициентами. Построение системы уравнений с регулярными коэффициентами на основе регулярной грамматики и ее решение. Автоматизированные методы построения лексических распознавателей, программа LEX и ее аналоги.<br>Тема 2.5. Свойства регулярных языков.<br>Способы определения регулярных языков, эквивалентность регулярных множеств, регулярных грамматик и КА. Свойства регулярных языков, решение проблем однозначности и проблем преобразования для регулярных грамматик. Лемма о разрастании для регулярных языков. |
| **3** | Тема 3.1. Синтаксический анализ. Назначение и функции синтаксического анализа.<br>Задачи и функции синтаксического анализатора, синтаксический анализатор – основная и самая сложная по вычислительным затратам часть компилятора. Этапы построения синтаксического анализатора, исходные данные, необходимые для выполнения синтаксического анализа. Восходящие и нисходящие синтаксические распознаватели.<br>Тема 3.2. Контекстно-свободные языки и грамматики.<br>Контекстно-свободные языки (КС-языки) – второй по сложности тип языков в классификации Хомского. Определение контекстно-свободной (КС) грамматики. Проблемы, связанные с КС-языками. Понятие о том, что языки программирования в строгом смысле слова не являются КС-языками, поэтому компилятор разбивает их анализ на две части: синтаксический и семантический анализ.<br>Тема 3.3. Автоматы с магазинной памятью (МП-автоматы). Расширенные МП-автоматы.<br>Определение МП-автомата, способы задания МП-автоматов, |

| Номер раздела | Название и содержание разделов и тем лекционных занятий |
|---|---|
| | принципы функционирования МП-автоматов. Детерминированные (ДМП) и недетерминированные МП-автоматы, невозможность построения ДМП-автомата, эквивалентного исходному МП-автомату в общем случае (невозможность приведения МП-автомата к детерминированному виду в общем случае). Определение детерминированных КС-языков (ДКС-языков), взаимосвязь ДКС-языков и ДМП-автоматов, взаимосвязь ДКС-языков с однозначными КС-грамматиками. Расширенные МП-автоматы – определение, отличия от обычных МП-автоматов. Эквивалентность МП-автоматов, расширенных МП-автоматов и КС-языков.<br>Тема 3.4. Построение синтаксического распознавателя по заданной грамматике.<br>Задача разбора и ее решение для КС-языков. Построение синтаксического распознавателя с подбором альтернатив на основе МП-автомата по заданной КС-грамматике, проблемы распознавателя с подбором альтернатив. Построение синтаксического распознавателя «сдвиг-свертка» на основе расширенного МП-автомата по заданной КС-грамматике, проблемы распознавателя «сдвиг-свертка». Преобразования КС-грамматик: условия преобразований, цели преобразований. Основные алгоритмы преобразования КС-грамматик: удаление бесплодных (бесполезных) символов, удаление недостижимых символов, устранение пустых правил, устранение цепных правил, устранение левой рекурсии. Синтаксические распознаватели с возвратом – распознаватель с подбором альтернатив и распознаватель типа «сдвиг-свертка»: построение, функционирование, основные преимущества и недостатки. Универсальные синтаксические распознаватели (табличные распознаватели) – алгоритм Кока-Янгера-Касами и алгоритм Эрли – их построение, функционирование, основные преимущества и недостатки.<br>Тема 3.5. Линейные синтаксические распознаватели.<br>Понятие о линейном синтаксическом распознавателе, основные преимущества линейных синтаксических распознавателей, отсутствие универсальности при применении линейных синтаксических распознавателей. Простейший линейный синтаксический распознаватель – распознаватель по методу рекурсивного спуска – его построение, свойства, ограничения на применение. Распознаватели на основе LL(k)-грамматик – основные свойства, принципы функционирования, ограничения. Построение простейшего распознавателя на основе LL(1)-грамматики, множества $FIRST_1$ и $FOLLOW_1$. Распознаватели на основе LR(k)-грамматик – основные свойства, принципы функционирования, ограничения. Часто используемые распознаватели на основе SLR(1)-грамматик и LALR(1)-грамматик. Распознаватели на основе грамматик предшествования – основные свойства, принципы функционирования, ограничения. Алгоритм распознавателя «сдвиг-свертка» для грамматик предшествования. Построение простейшего распознавателя на основе грамматики операторного предшествования, множества L(U) и R(U), $L^t(U)$ и $R^t(U)$, матрица предшествования и ее заполнение. Автоматизированные методы построения синтаксических распознавателей, программа YACC и ее аналоги.<br>Тема 3.6. Свойства КС-языков.<br>Способы определения КС-языков, эквивалентность КС-грамматик и |

| Номер раздела | Название и содержание разделов и тем лекционных занятий |
|---|---|
| | МП-автоматов. Свойства КС-языков, неразрешимость проблем однозначности и проблем преобразования для КС-грамматик в общем случае. Классы КС-языков и КС-грамматик, решение проблем однозначности и проблем преобразования для определенных классов КС-грамматик. Лемма о разрастании для КС-языков.<br>Тема 3.7. Внутренние представления программ.<br>Понятие о внутреннем представлении программы, информация, хранимая во внутреннем представлении программы, использование компилятором нескольких вариантов внутреннего представления программы по мере необходимости. Машинно-зависимые и машинно-независимые способы внутреннего представления программ. Основные машинно-независимые способы внутреннего представления программ: дерево операций (и его взаимосвязь с синтаксическим деревом), трехадресный код (тетрады), двухадресный код (триады), постфиксная (или обратная польская) запись операций. Основные преимущества и недостатки способов внутреннего представления программ. Вычисления выражений в форме постфиксной записи операций. |
| **4** | Тема 4.1. Проход. Понятие прохода. Однопроходные и многопроходные компиляторы.<br>Понятие прохода. Последовательность выполнения проходов компилятором, взаимосвязь между последовательно выполняемыми проходами. Количество проходов компилятора, однопроходные и многопроходные компиляторы. Влияние количества проходов на вычислительную сложность компилятора, зависимость эффективности результирующей программы от количества проходов.<br>Тема 4.2. Язык ассемблера. Особенности компиляторов с языка ассемблера.<br>Определение языка ассемблера (мнемокода). Области применения языков ассемблера в современных вычислительных системах. Компилятор с языка ассемблера как простейший случай компилятора. Типовая грамматика языка ассемблера. Особенности компиляторов с языка ассемблера.<br>Тема 4.3. Макрогенерация и макроподстановки. Условная компиляция.<br>Понятие о макрогенерации, основные цели выполнения макрогенерации. Макроопределения («макросы») и их структура, выполнение макроподстановок на основе макроопределений. Понятие о макропроцессоре (препроцессоре) как о текстовом трансляторе с макроязыка. Примеры выполнения макроподстановок без параметров и с параметрами. Возможные проблемы при выполнении макроподстановок, сравнение выполнения макроподстановок с вызовом функций. Организация условной компиляции на основе макрогенерации, задачи, решаемые с помощью условной компиляции.<br>Тема 4.3. Мобильность и переносимость программного обеспечения. Особенности компиляторов с языков Java и C#.<br>Понятие мобильности и переносимости программного обеспечения. Основные сложности обеспечения мобильности кода. Основные способы обеспечения переносимости программного кода: стандартизация (соответствие стандартам переносимости), условная |

| Номер раздела | Название и содержание разделов и тем лекционных занятий |
|---|---|
| | компиляция (с помощью макроподстановок), использование интерпретируемых языков. Переносимость кода с помощью сочетания компиляции и интерпретации: компилятор – промежуточный код – интерпретатор – выполнение кода. Особенности компиляторов с языков Java и C#, ориентированных на сочетание компиляции и интерпретации кода. |
| **5** | Тема 5.1. Семантический анализ. Основные функции семантического анализа.<br>Семантика и синтаксис языка, зависимость семантики от синтаксиса. Семантический анализ, основные задачи семантического анализа, исходные данные, необходимые для выполнения семантического анализа. Функции семантического анализа: проверка семантических ограничений входного языка, дополнение внутреннего представления программы неявными вызовами процедур и функций, проверка общих семантических ограничений языков программирования. Ошибки и предупреждения, порождаемые семантическим анализатором. Примеры ошибочных ситуаций и ситуаций, вызывающих предупреждения. Ограничения на возможности семантического анализа в компиляторе.<br>Тема 5.2. Распределение памяти. Организация компилятором таблиц RTTI.<br>Основные принципы распределения памяти компилятором, статическая и динамическая, локальная и глобальная области памяти. Вычисления объема памяти для простейших типов данных, зависимость объема памяти от целевой вычислительной системы. Формулы вычисления объема памяти для сложных структур данных: массивов, структур (записей), союзов (записей с вариантами), классов и объектов. Кратность адресации в вычислительных системах, выравнивание адресов ячеек памяти в зависимости от кратности адресации, влияние кратности на производительность результирующей программы и на объем требуемой оперативной памяти. Дисплей памяти процедуры (функции), стековая организация дисплея памяти процедуры (функции), преимущества и недостатки стековой организации дисплея памяти процедуры (функции). Особенности стековой организации дисплея памяти процедуры (функции) в различных вычислительных системах, соглашения о вызовах, два типа основных соглашений о вызовах: соглашение о вызовах языка Си и соглашение о вызовах языка Паскаль. Организация компилятором таблиц информации о типах данных (Run-Time Type Information – RTTI) в объектно-ориентированных языках программирования, информация, хранимая в таблицах RTTI и ее взаимосвязь с классами (объектами) в объектно-ориентированных языках. Проблемы использования таблиц RTTI при работе с библиотеками классов (объектов).<br>Тема 5.3. Исключительные ситуации. Обработка исключительных ситуаций.<br>Понятие об исключительных ситуациях, роль исключительных ситуаций в объектно-ориентированных языках программирования. Необходимость поддержки обработки исключительных ситуаций со стороны операционной системы (ОС) на целевой вычислительной системе. Порождение компилятором кода для обработки исключительных ситуаций, явный и неявный код обработки |

| Номер раздела | Название и содержание разделов и тем лекционных занятий |
|---|---|
| | исключительных ситуаций. Возможные уровни обработки исключительных ситуаций – на уровне кода приложения, на уровне библиотеки системы программирования, на уровне ОС. Основные типы исключительных ситуаций – аппаратные, порождаемые ОС, порождаемые системой программирования, пользовательские. Новые возможности и ограничения при обработке исключительных ситуаций, вычислительные затраты на обработку исключительных ситуаций.<br>Тема 5.4. Синтаксически управляемый перевод. Генерация результирующего кода.<br>Проблемы, возникающие при порождении результирующего кода компилятором. Понятие синтаксически управляемого перевода (СУ-перевода), зависимость семантики кода от его синтаксиса. Использование метода СУ-перевода для генерации результирующего кода компилятором. Ограничения на применение метода СУ-перевода. Проблемы эффективности результирующего кода, возникающие из-за применения метода СУ-перевода.<br>Тема 5.5. Оптимизация. Основные принципы и методы оптимизации.<br>Причины необходимости применения оптимизации при порождении результирующего кода компилятором. Определение оптимизации, основные принципы выполнения оптимизации, условие отсутствия влияния оптимизации на семантику результирующей программы. Критерии эффективности результирующей программы: скорость выполнения и потребность в вычислительных ресурсах (в основном – в оперативной памяти), зависимость оптимизации от критериев эффективности. Машинно-зависимая и машинно-независимая оптимизация, сочетание оптимизации со способами внутреннего представления программы и с процессом генерации результирующего кода. Основные машинно-независимые методы оптимизации (для линейных участков кода): свертка объектного кода, исключение лишних операций, оптимизация арифметических выражений, оптимизация логических выражений, оптимизация вызовов процедур и функций. |
| **6** | Тема 6.1. Структура систем программирования.<br>Определение системы программирования. Основные составные части системы программирования: текстовый редактор, компилятор, компоновщик, библиотеки, загрузчик, отладчик, компилятор ресурсов, редактор ресурсов. Структурная схема системы программирования, последовательность обработки данных в системе программирования.<br>Тема 6.2. История развития систем программирования.<br>Начальный этап развития систем программирования: выполнение всей последовательности обработки данных в ручном режиме, высокая трудоемкость разработки, высокая вероятность возникновения ошибок в процессе обработки данных. Первый этап развития систем программирования: появление языка описания процесса обработки данных (Makefile), выполнение процесса обработки данных на основе интерпретатора Makefile, передача функций загрузчика в операционную систему (ОС). Второй этап развития систем программирования: появление интегрированных сред разработки, выполнение процесса обработки данных в одной среде (в окне текстового редактора). Первоначальное внедрение интегрированных сред в качестве обучающих систем |

| Номер раздела | Название и содержание разделов и тем лекционных занятий |
|---|---|
| | программирования, их дальнейшее развитие и выход на рынок профессиональных средств разработки. Противостояние систем на основе Makefile и интегрированных сред на рынке профессиональных средств разработки, поглощение систем на основе Makefile системами на основе интегрированных сред. Третий этап развития систем программирования: появление графического интерфейса пользователя (GUI – Graphical User Interface), включение в состав систем программирования средств создания и обработки ресурсов пользовательского интерфейса. Взаимодействие систем программирования и средств автоматизированного проектирования программного обеспечения (CASE – Computer Aided Software Engineering).<br>Тема 6.3. Функции основных компонент в составе систем программирования.<br>Функции текстовых редакторов в составе системы программирования. Функции компилятора с входного языка. Назначение компоновщика (редактора связей), основные функции компоновщика. Библиотеки: состав библиотек по назначению, типы библиотек – статические и динамические библиотеки. Преимущества и недостатки статических и динамических библиотек, технологии дальнейшего развития динамических библиотек. Функции загрузчика, настраивающий и динамический загрузчики, взаимодействие систем программирования с ОС целевой вычислительной системы. Функции отладчика, взаимодействие отладчика с другими компонентами системы программирования, формирование и использование отладочной информации в результирующей программе. Ресурсы пользовательского интерфейса: определение ресурсов пользовательского интерфейса, состав, формирование и использование ресурсов пользовательского интерфейса. Функции компилятора ресурсов пользовательского интерфейса. Функции редактора ресурсов пользовательского интерфейса, взаимодействие редактора ресурсов пользовательского интерфейса с внешними источниками данных. |
| **7** | Тема 7.1. Принципы организации распределенных вычислений.<br>Результирующая программа как выполняемое приложение. Возможные схемы организации распределенных вычислений, многообразие способов организации распределенных вычислений. Организация распределенных вычислений с разбиением приложений по уровням. Четыре основных уровня при построении приложений: уровень файловых операций, уровень операций со структурами данных, уровень бизнес-логики, уровень интерфейса с пользователем.<br>Тема 7.2. Технология «файл-сервер».<br>Выделение файловых операций на отдельный компьютер, понятие файлового сервера. Общее описание технологии «файл-сервер». Поддержка требований к файловым серверам на уровне современных ОС. Главное преимущество технологии «файл-сервер»: отсутствие каких-либо дополнительных требований к разработке приложений. Недостатки технологии «файл-сервер» и методы борьбы с ними.<br>Тема 7.3. Терминальный сервер.<br>Выделение средств по организации взаимодействия с пользователем |

| Номер раздела | Название и содержание разделов и тем лекционных занятий |
|---|---|
| | на отдельный компьютер, понятие терминального сервера. Общее описание терминального сервера. Протоколы, используемые для организации терминальных серверов: протокол XWindow и протокол RDP (Remote Desktop Protocol). Особенности работы приложений на терминальном сервере. Требования к ресурсам терминального сервера.<br>Тема 7.4. Технология «клиент-сервер».<br>Выделение операций со структурами данных на отдельный компьютер, понятие сервера данных. Общее описание технологии «клиент-сервер», технология «клиент-сервер» как следующий шаг в развитии технологии «файл-сервер». Общие требования к серверу данных, определяющее требование по структуризации данных. Организация сервера данных с помощью СУБД (Система Управления Базами Данных). Требования к средствам разработки клиентских приложений по технологии «клиент-сервер». Взаимодействие клиентских приложений с сервером данных: организация взаимодействия с сервером данных, основные технологии взаимодействия, язык структурированных запросов SQL (Structured Query Language). Основные преимущества технологии «клиент-сервер» по сравнению с технологией «файл-сервер». Основные недостатки технологии «клиент-сервер» и методы борьбы с ними: перенос части бизнес-логики на сервер данных, использование терминальных серверов для функционирования клиентских приложений. Ограничения возможностей технологии «клиент-сервер».<br>Тема 7.5. Трехуровневые и многоуровневые технологии разработки программного обеспечения.<br>Вынесение операций бизнес логики на отдельный компьютер, понятие сервера приложений. Общее описание трехуровневой технологии, трехуровневая технология как следующий шаг в развитии технологии «клиент-сервер». Общие требования к серверу приложений, взаимодействия сервера приложений и сервера данных. Взаимодействие сервера приложений и клиентских приложений, сложности в организации взаимодействия, технологии взаимодействия CORBA и COM/DCOM. Требования к средствам разработки клиентских приложений по трехуровневой технологии, понятие об интерфейсах взаимодействия (интерфейсах сервера приложений), организация интерфейсов взаимодействия, сравнение интерфейсов с таблицей RTTI и динамическими библиотеками. Шаблоны для построения серверов приложений, предлагаемые системами программирования. Возможность разбиения сервера приложений на несколько серверов, переход от трехуровневой к многоуровневой (multi-tier) технологии разработки. Основные преимущества трехуровневой и многоуровневой технологий по сравнению с технологиями «клиент-сервер» и «файл-сервер». Основные сложности разработки приложений по трехуровневой и многоуровневой технологиям. Современное состояние рынка средств разработки.<br>Тема 7.6. Облачные вычисления и сервисы.<br>Понятие об облачных вычислениях и сервисах, требования, предъявляемые к облачным вычислениям и сервисам. Облачные вычисления и сервисы, построенные по принципам HaaS (Hardware as a Service) и SaaS (Software as a Service) – основные понятия и возможности. Организация облачных сервисов на основе основных |

| Номер раздела | Название и содержание разделов и тем лекционных занятий |
|---|---|
| | технологий разработки приложений: «файл-сервер», «клиент-сервер» и многоуровневая технология. |
| **8** | Тема 8.1. Особенности разработки программного обеспечения для глобальных сетей. Применение современных интерпретаторов.<br>Главная особенность глобальных сетей – многообразие целевых вычислительных систем, отсутствие ограничений на структуру целевой вычислительной системы. Следствия по разработке программ для глобальных сетей: невозможность использования компиляторов, необходимость использования интерпретаторов. Глобальные сети – основная область применения современных интерпретаторов и интерпретируемых языков. Типовая структура программы навигации по сети Интернет (browser): сетевая часть, поддерживающая обмен данными по сети, и интерпретатор поступивших данных.<br>Тема 8.2. Построение статических Web-страниц. Язык HTML.<br>Статическая Web-страница – хранилище неизменяемых данных, организация статических Web-страниц, как сетевого хранилища структурированных файлов. Организация доступа к Web-страницам: протокол доступа HTTP (Hyper-Text Transfer Protocol), язык описания страниц HTML (Hyper-Text Markup Language). Любая программа навигации содержит интерпретатор HTML. Основной недостаток статических Web-страниц – предоставляемая информация не зависит от запросов и условий, задаваемых пользователями.<br>Тема 8.3. Построение динамических Web-страниц на основе компилируемых и интерпретируемых языков.<br>Принципы организации динамических Web-страниц: формирование текста HTML страницы с учетом данных и условий, полученных в запросе пользователя. Использование различных технологий разработки приложений (технология «клиент-сервер» и многоуровневая технология) для организации динамических Web-страниц. Сочетание динамических Web-страниц и технологии «клиент-сервер» как простейшее средство построения сервера приложений. Возможности взаимодействия сервера Интернет с приложениями, технологии CGI (Common-Gateway Interface) и ISAPI (Internet Server Application Program Interface), преимущества и недостатки каждой технологии. Построение взаимодействующих с сервером Интернет приложений с помощью компилятора или с помощью интерпретатора, преимущества и недостатки каждого способа. Ограничения по интерактивности динамических Web-страниц.<br>Тема 8.4. Построение интерактивных Web-страниц.<br>Требования к интерактивности Web-страниц и ограничения по организации интерактивности на стороне сервера. Принципы построения интерактивных Web-страниц: организация выполнения кода на стороне клиента. Способы организации выполнения кода на стороне клиента: скрипты (scripts) и скриптовые языки, интерпретаторы и интерпретируемые языки, встраиваемые компоненты (технология ActiveX и подобные ей). Ограничения по выполнению кода на стороне клиента. Вопросы безопасности выполнения кода на стороне клиента. Общая схема организации Web-страниц с учетом динамичности и интерактивности. |

4.3. Практические (семинарские) занятия

Темы практических занятий и их трудоемкость приведены в таблице 5.

Таблица 5 – Практические занятия и их трудоемкость

| № п/п | Темы практических занятий | Формы практических занятий | Трудоемкость, (час) | Из них практической подготовки, (час) | № раздела дисциплины |
|---|---|---|---|---|---|
| Учебным планом не предусмотрено | | | | | |
| | | | | | |
| Всего | | | | | |

4.4. Лабораторные занятия

Темы лабораторных занятий и их трудоемкость приведены в таблице 6.

Таблица 6 – Лабораторные занятия и их трудоемкость

| № п/п | Наименование лабораторных работ | Трудоемкость, (час) | Из них практической подготовки, (час) | № раздела дисциплины |
|---|---|---|---|---|
| Семестр 7 | | | | |
| 1 | Вводное занятие | 1 | 1 | 1 |
| 2 | Работа с таблицами идентификаторов | 2 | 2 | 1 |
| 3 | Построение распознавателя для регулярной грамматики | 2 | 2 | 2 |
| 4 | Построение лексического анализатора | 2 | 2 | 2 |
| 5 | Построение синтаксического анализатора | 2 | 2 | 3 |
| 6 | Построение простейшего дерева вывода | 1 | 1 | 3 |
| Семестр 8 | | | | |
| 7 | Вводное занятие | 1 | 1 | 5 |
| 8 | Генерация объектного кода | 1 | 1 | 5 |
| 9 | Оптимизация объектного кода | 1 | 1 | 5 |
| 10 | Построение распознавателя для анализа описаний типов данных и переменных | 1 | 1 | 5 |
| 11 | Определение объема памяти для заданных структур данных | 2 | 2 | 5 |
| 12 | Разработка простейшего приложения на основе технологии «клиент-сервер» | 4 | 4 | 7 |
| Всего | | 20 | 20 | |

4.5. Выполнение курсового проекта/ курсовой работы

Учебным планом не предусмотрено

4.6. Самостоятельная работа обучающихся

Виды самостоятельной работы и ее трудоемкость приведены в таблице 7.

Таблица 7 – Виды самостоятельной работы и ее трудоемкость

| Вид самостоятельной работы | Всего, час | Семестр 7, час | Семестр 8, час |
|---|---|---|---|
| 1 | 2 | 3 | 4 |
| Изучение теоретического материала дисциплины (ТО) | 92 | 30 | 52 |
| Курсовое проектирование (КП, КР) | | | |
| Расчетно-графические задания (РГЗ) | | | |
| Выполнение реферата (Р) | | | |
| Подготовка к текущему контролю успеваемости (ТКУ) | 48 | 8 | 20 |
| Домашнее задание (ДЗ) | | | |
| Контрольные работы заочников (КРЗ) | 30 | 4 | 20 |
| Подготовка к промежуточной аттестации (ПА) | 47 | 10 | 23 |
| Всего: | 207 | 52 | 115 |

## 5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)

Учебно-методические материалы для самостоятельной работы обучающихся указаны в п.п. разделов 6-11.

## 6. Перечень печатных и электронных учебных изданий

Перечень печатных и электронных учебных изданий приведен в таблице 8.

Таблица 8– Перечень печатных и электронных учебных изданий

| Шифр/ URL адрес | Библиографическая ссылка | Количество экземпляров в библиотеке (кроме электронных экземпляров) |
|---|---|---|
| 004.1<br>М76 | Разработка распределенных и облачных приложений /А.Ю. Молчанов; С.-Петерб. гос. ун-т аэрокосм. приборостроения. - Санкт-Петербург: Изд-во ГУАП, 2024. - 100 с. | 20 |
| 004<br>М76 | Разработка корпоративных программных решений на платформе .NET : учебное пособие / А.Ю. Молчанов, А.В. Аксенов; С.-Петерб. гос. ун-т аэрокосм. приборостроения. - Санкт-Петербург: Изд-во ГУАП, 2022. - 140 с. | 5 |
| 004.7<br>К95 | Интерфейсы взаимодействия приложений и распределенные вычисления /Н.В. Кучин, А.Ю. Молчанов – СПб.: Изд-во ГУАП, 2020 – 125 с. | 40 |
| 004.7<br>К95 | Многоуровневые системы и облачные вычисления /Н.В. Кучин, А.Ю. Молчанов – СПб.: Изд-во ГУАП, 2018 – 113 с. | 40 |

| Шифр/ URL адрес | Библиографическая ссылка | Количество экземпляров в библиотеке (кроме электронных экземпляров) |
|---|---|---|
| 004.4 М76 | Системное программное обеспечение: лабораторный практикум /А. Ю. Молчанов; С.-Петерб. гос. ун-т аэрокосм. приборостроения. - СПб.: Изд-во ГУАП, 2014. – 131 с. | 35 |
| 004.4(075) М75 | Системное программное обеспечение: учебник / А. Ю. Молчанов. - СПб.: ПИТЕР, 2010. - 395 с | 49 |
| 004.4(075) М75 | Системное программное обеспечение: лабораторный практикум / А. Ю. Молчанов. - СПб.: ПИТЕР, 2005. - 283 с. | 38 |
| 004.4 Г67 | Системное программное обеспечение: учебник / А. В. Гордеев, А. Ю. Молчанов. - СПб.: ПИТЕР, 2002. - 734 с. | 61 |

## 7. Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет»

Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины приведен в таблице 9.

Таблица 9 – Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет»

| URL адрес | Наименование |
|---|---|
| | Не требуется |

## 8. Перечень информационных технологий

8.1. Перечень программного обеспечения, используемого при осуществлении образовательного процесса по дисциплине.

Перечень используемого программного обеспечения представлен в таблице 10.

Таблица 10– Перечень программного обеспечения

| № п/п | Наименование |
|---|---|
| | Не предусмотрено |

8.2. Перечень информационно-справочных систем,используемых при осуществлении образовательного процесса по дисциплине

Перечень используемых информационно-справочных систем представлен в таблице 11.

Таблица 11– Перечень информационно-справочных систем

| № п/п | Наименование |
|---|---|
| | Не предусмотрено |

## 9. Материально-техническая база

Состав материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине, представлен в таблице12.

Таблица 12 – Состав материально-технической базы

| № п/п | Наименование составной части материально-технической базы | Номер аудитории (при необходимости) |
|---|---|---|
| 1 | Лекционная аудитория | 32-04 |
| 2 | Вычислительная лаборатория | 52-09 |

## 10. Оценочные средства для проведения промежуточной аттестации

10.1. Состав оценочных средствдля проведения промежуточной аттестации обучающихся по дисциплине приведен в таблице 13.

Таблица 13 – Состав оценочных средств для проведения промежуточной аттестации

| Вид промежуточной аттестации | Перечень оценочных средств |
|---|---|
| Экзамен | Список вопросов к экзамену;<br>Экзаменационные билеты*;<br>Тесты. |
| Дифференцированный зачёт | Список вопросов;<br>Тесты. |

Примечание: *экзаменационные билеты формируются на основе вопросов и задач таблицы 15.

10.2. В качестве критериев оценки уровня сформированности (освоения) компетенций обучающимися применяется 5-балльная шкала оценки сформированности компетенций, которая приведена в таблице 14. В течение семестра может использоваться 100-балльная шкала модульно-рейтинговой системы Университета, правила использования которой, установлены соответствующим локальным нормативным актом ГУАП.

Таблица 14 –Критерии оценки уровня сформированности компетенций

| Оценка компетенции<br>5-балльная шкала | Характеристика сформированных компетенций |
|---|---|
| «отлично»<br>«зачтено» | Обучающийся:<br>– глубоко и всесторонне усвоил программный материал;<br>– уверенно, логично, последовательно и грамотно его излагает;<br>– опираясь на знания основной и дополнительной литературы, тесно связывает усвоенные научные положения с практической деятельностью направления;<br>– умело обосновывает и аргументирует выдвигаемые им идеи;<br>– делает выводы и обобщения;<br>– свободно владеет системой специализированных понятий.<br>– правильно выполнил от 90% до 100% тестовых заданий**. |
| «хорошо»<br>«зачтено» | Обучающийся:<br>– твердо усвоил программный материал, грамотно и по существу излагает его, опираясь на знания основной литературы;<br>– не допускает существенных неточностей;<br>– увязывает усвоенные знания с практической деятельностью направления;<br>– аргументирует научные положения;<br>– делает выводы и обобщения;<br>– владеет системой специализированных понятий.<br>– правильно выполнил от 70% до 89% тестовых заданий**. |

| Оценка компетенции<br>5-балльная шкала | Характеристика сформированных компетенций |
|---|---|
| «удовлетворительно»<br>«зачтено» | – обучающийся усвоил только основной программный материал, по существу излагает его, опираясь на знания только основной литературы;<br>– допускает несущественные ошибки и неточности;<br>– испытывает затруднения в практическом применении знаний направления;<br>– слабо аргументирует научные положения;<br>– затрудняется в формулировании выводов и обобщений;<br>– частично владеет системой специализированных понятий.<br>– правильно выполнил от 51% до 69% тестовых заданий**. |
| «неудовлетворительно»<br>«не зачтено» | – обучающийся не усвоил значительной части программного материала;<br>– допускает существенные ошибки и неточности при рассмотрении проблем в конкретном направлении;<br>– испытывает трудности в практическом применении знаний;<br>– не может аргументировать научные положения;<br>– не формулирует выводов и обобщений.<br>– правильно выполнил менее 51% тестовых заданий**. |

10.3. Типовые контрольные задания или иные материалы.

Вопросы (задачи) для экзамена представлены в таблице 15.

Таблица 15 – Вопросы (задачи) для экзамена

| № п/п | Перечень вопросов (задач) для экзамена | Код индикатора |
|---|---|---|
| 1. | Трансляторы, компиляторы и интерпретаторы. Структура компилятора, основные этапы компиляции. | ПК-4.З.1 |
| 2. | Определение алфавита, цепочки символов, языка. Операции над цепочками символов. Способы задания языков. | ПК-4.З.1 |
| 3. | Грамматики. Определение грамматики. Форма Бэкуса-Наура. Другие формы представления грамматик. | ПК-4.З.1 |
| 4. | Выполнение классификации языков и грамматик по Хомскому. Проблема преобразования грамматик. | ПК-4.У.1 |
| 5. | Цепочки вывода. Сентенциальная форма. Правосторонний и левосторонний выводы. Дерево вывода. Однозначные и неоднозначные грамматики. | ПК-4.З.1 |
| 6. | Распознаватели. Общая схема распознавателя. Классификация распознавателей. | ПК-4.З.1 |
| 7. | Конечные автоматы (КА). Определение КА. Детерминированные и недетерминированные КА. | ПК-4.З.1 |
| 8. | Регулярные множества и регулярные выражения, их свойства. Решение уравнений с регулярными коэффициентами. | ПК-4.У.1 |
| 9. | Связь регулярных множеств, регулярных грамматик и КА. Построение автоматной грамматики по заданной регулярной грамматике. | ПК-4.У.1 |
| 10. | Связь регулярных множеств, регулярных грамматик и КА. Построение КА по заданной автоматной грамматике. | ПК-4.З.1 |
| 11. | Автоматы с магазинной памятью (МП-автоматы). Расширенные МП-автоматы. Детерминированные МП-автоматы и ДКС-языки. | ПК-4.З.1 |
| 12. | Построение МП-автомата на основе КС-грамматики. МП-автомат с подбором альтернатив. | ПК-4.У.1 |
| 13. | Построение расширенного МП-автомата на основе КС-грамматики. | ПК-4.У.1 |

| № п/п | Перечень вопросов (задач) для экзамена | Код индикатора |
|---|---|---|
| | Расширенный МП-автомат на основе алгоритма «сдвиг-свертка». | |
| 14. | Преобразование КС-грамматик. Алгоритмы разбора с возвратами и их характеристики. Табличные распознаватели и их характеристики. | ПК-4.У.1 |
| 15. | Свойства регулярных языков и грамматик. Свойства КС-языков и КС-грамматик. | ПК-4.З.1 |
| 16. | Распознавание цепочек КС-языков методом рекурсивного спуска. | ПК-4.В.1 |
| 17. | LL(k)-грамматики. Определение, свойства, принципы построения распознавателей. | ПК-4.З.1 |
| 18. | LR(k)-грамматики. Определение, свойства, принципы построения распознавателей. SLR(1) и LALR(1)-грамматики. | ПК-4.З.1 |
| 19. | Грамматики предшествования. Алгоритм «сдвиг-свертка» для грамматик предшествования. | ПК-4.З.1 |
| 20. | Грамматики предшествования. Построение отношений предшествования для грамматик операторного предшествования. | ПК-4.У.1 |
| 21. | Автоматизированные методы построения распознавателей. Построение распознавателей с помощью программ LEX и YACC. | ПК-4.В.1 |
| 22. | Сканеры (лексические анализаторы). Назначение и функции лексического анализатора. Способы взаимосвязи синтаксического и лексического анализаторов. | ПК-4.З.1 |
| 23. | Таблицы символов. Организация таблиц символов в виде линейных списков и по методу бинарного дерева. | ПК-4.В.1 |
| 24. | Таблицы символов. Организация таблиц символов с помощью хэш-функций на основе рехэширования и на основе метода цепочек. | ПК-4.В.1 |
| 25. | Семантический анализ. Основные действия, выполняемые на этапе семантического анализа. | ПК-4.З.1 |
| 26. | Внутреннее представление программы. Методы внутреннего представления программ. | ПК-4.З.1 |
| 27. | Обратная польская запись. Вычисление выражений в форме обратной польской записи. | ПК-4.У.1 |
| 28. | Распределение памяти. Динамическая и статическая память, глобальная и локальная память. | ПК-4.З.1 |
| 29. | Распределение памяти. Вычисление объема памяти под простые и сложные структуры данных. | ПК-4.У.1 |
| 30. | Распределение памяти. Стековая организация дисплея памяти, соглашения о вызовах. | ПК-4.З.1 |
| 31. | Таблицы RTTI – структура и назначение. Организация компилятором таблиц RTTI. | ПК-4.З.1 |
| 32. | Генерация кода. Назначение генерации кода. Принцип СУ-перевода. | ПК-4.З.1 |
| 33. | Оптимизация программ. Назначение оптимизации, основные принципы и методы оптимизации. | ПК-4.З.2 |
| 34. | Оптимизация программ. Оптимизация линейных участков: свертка объектного кода, исключение лишних операций. | ПК-4.З.2 |
| 35. | Оптимизация программ. Особенности оптимизации логических и арифметических выражений, вызовов процедур и функций. | ПК-4.З.2 |
| 36. | Понятие прохода. Однопроходные и многопроходные компиляторы. | ПК-4.З.1 |
| 37. | Язык ассемблера. Особенности построения компиляторов с языка ассемблера. | ПК-4.З.1 |
| 38. | Макроопределения и макрогенерация. Условная компиляция. | ПК-4.З.1 |
| 39. | Особенности построения интерпретаторов. Области применения | ПК-4.З.1 |

| № п/п | Перечень вопросов (задач) для экзамена | Код индикатора |
|---|---|---|
| | современных интерпретаторов. | |
| 40. | Современные системы программирования. Структура и составные части систем программирования. | ПК-4.З.2 |
| 41. | Современные системы программирования. Функции текстовых редакторов в системах программирования. | ПК-4.З.2 |
| 42. | Современные системы программирования. Функции компоновщиков и загрузчиков в системах программирования. | ПК-4.З.2 |
| 43. | Современные системы программирования. Организация библиотек процедур и функций, статические и динамические библиотеки. | ПК-4.З.2 |
| 44. | Современные системы программирования. Ресурсы пользовательского интерфейса: создание, обработка и использование ресурсов. | ПК-4.З.2 |
| 45. | Мобильность и переносимость кода. Принципы разработки переносимого кода. Особенности разработки на языках Java и C#. | ПК-4.З.2 |
| 46. | Концепция организации распределенных вычислений. Технология «файл-сервер». Недостатки и преимущества технологии «файл-сервер». | ПК-4.З.2 |
| 47. | Концепция организации распределенных вычислений. Технология «клиент-сервер». Недостатки и преимущества технологии «клиент-сервер». | ПК-4.З.2 |
| 48. | Построение приложений по трехуровневой и многоуровневой технологии. Недостатки и преимущества многоуровневых технологий. | ПК-4.В.1 |
| 49. | Особенности разработки программного обеспечения для сети Интернет. Разработка статических Web-страниц. | ПК-4.В.1 |
| 50. | Разработка динамических Web-страниц на основе CGI, ISAPI и интерпретаторов. Разработка интерактивных Web-страниц. | ПК-4.В.1 |

Вопросы (задачи) для зачета / дифф. зачета представлены в таблице 16.

Таблица 16 – Вопросы (задачи) для зачета / дифф. Зачета

| № п/п | Перечень вопросов (задач) для зачета / дифф. зачета | Код индикатора |
|---|---|---|
| 1. | Трансляторы, компиляторы и интерпретаторы. Структура компилятора, основные этапы компиляции. | ПК-4.З.1 |
| 2. | Определение алфавита, цепочки символов, языка. Операции над цепочками символов. Способы задания языков. | ПК-4.З.1 |
| 3. | Грамматики. Определение грамматики. Форма Бэкуса-Наура. Другие формы представления грамматик. | ПК-4.З.1 |
| 4. | Выполнение классификации языков и грамматик по Хомскому. Проблема преобразования грамматик. | ПК-4.У.1 |
| 5. | Цепочки вывода. Сентенциальная форма. Правосторонний и левосторонний выводы. Дерево вывода. Однозначные и неоднозначные грамматики. | ПК-4.З.1 |
| 6. | Распознаватели. Общая схема распознавателя. Классификация распознавателей. | ПК-4.З.1 |
| 7. | Конечные автоматы (КА). Определение КА. Детерминированные и недетерминированные КА. | ПК-4.З.1 |
| 8. | Регулярные множества и регулярные выражения, их свойства. Решение уравнений с регулярными коэффициентами. | ПК-4.У.1 |

| № п/п | Перечень вопросов (задач) для зачета / дифф. зачета | Код индикатора |
|---|---|---|
| 9. | Связь регулярных множеств, регулярных грамматик и КА. Построение автоматной грамматики по заданной регулярной грамматике. | ПК-4.У.1 |
| 10. | Связь регулярных множеств, регулярных грамматик и КА. Построение КА по заданной автоматной грамматике. | ПК-4.З.1 |
| 11. | Автоматы с магазинной памятью (МП-автоматы). Расширенные МП-автоматы. Детерминированные МП-автоматы и ДКС-языки. | ПК-4.З.1 |
| 12. | Построение МП-автомата на основе КС-грамматики. МП-автомат с подбором альтернатив. | ПК-4.У.1 |
| 13. | Построение расширенного МП-автомата на основе КС-грамматики. Расширенный МП-автомат на основе алгоритма «сдвиг-свертка». | ПК-4.У.1 |
| 14. | Преобразование КС-грамматик. Алгоритмы разбора с возвратами и их характеристики. Табличные распознаватели и их характеристики. | ПК-4.У.1 |
| 15. | Свойства регулярных языков и грамматик. Свойства КС-языков и КС-грамматик. | ПК-4.З.1 |
| 16. | Распознавание цепочек КС-языков методом рекурсивного спуска. | ПК-4.В.1 |
| 17. | LL(k)-грамматики. Определение, свойства, принципы построения распознавателей. | ПК-4.З.1 |
| 18. | LR(k)-грамматики. Определение, свойства, принципы построения распознавателей. SLR(1) и LALR(1)-грамматики. | ПК-4.З.1 |
| 19. | Грамматики предшествования. Алгоритм «сдвиг-свертка» для грамматик предшествования. | ПК-4.З.1 |
| 20. | Грамматики предшествования. Построение отношений предшествования для грамматик операторного предшествования. | ПК-4.У.1 |
| 21. | Автоматизированные методы построения распознавателей. Построение распознавателей с помощью программ LEX и YACC. | ПК-4.В.1 |
| 22. | Сканеры (лексические анализаторы). Назначение и функции лексического анализатора. Способы взаимосвязи синтаксического и лексического анализаторов. | ПК-4.З.1 |
| 23. | Таблицы символов. Организация таблиц символов в виде линейных списков и по методу бинарного дерева. | ПК-4.В.1 |
| 24. | Таблицы символов. Организация таблиц символов с помощью хэш-функций на основе рехэширования и на основе метода цепочек. | ПК-4.В.1 |
| 25. | Семантический анализ. Основные действия, выполняемые на этапе семантического анализа. | ПК-4.З.1 |

Перечень тем для выполнения курсового проекта/ курсовой работы представлены в таблице 17.

Таблица 17 – Перечень тем для выполнения курсового проекта / курсовой работы

| № п/п | Примерный перечень тем для выполнения курсового проекта/ курсовой работы |
|---|---|
| | Учебным планом не предусмотрено |

Вопросы для проведения промежуточной аттестации в виде тестирования представлены в таблице 18.

Таблица 18 – Примерный перечень вопросов для тестов

| № п/п | Примерный перечень вопросов для тестов | Код индикатора |
|---|---|---|
| 1. | Задание с выбором одного верного ответа из предложенных<br>*Инструкция: Прочитайте текст и выберите один правильный ответ.*<br>Выберите термин, наиболее точно соответствующий определению «программа, которая переводит исходную программу, написанную на входном языке, в эквивалентную ей результирующую программу на выходном языке»:<br>а) Компилятор.<br>б) Транслятор.<br>в) Интерпретатор.<br>г) Компоновщик. | ПК-4.З.1<br>ПК-4.В.1 |
| 2. | Задание закрытого типа на установление последовательности<br>*Инструкция: Прочитайте текст и установите последовательность.*<br>Расположите этапы компиляции в порядке их выполнения:<br>а) Синтаксический анализ.<br>б) Лексический анализ.<br>в) Генерирование и оптимизация кода.<br>г) Подготовка к генерированию кода.<br>д) Семантический анализ.<br>Запишите соответствующую последовательность букв слева направо:<br>\| \| \| \| \| \| | ПК-4.З.1<br>ПК-4.У.1 |
| 3. | Задание комбинированного типа с выбором нескольких верных ответов из предложенных<br>*Инструкция: Прочитайте текст и выберите несколько правильных ответов.*<br>Укажите, какие типы лексем используются в языках программирования:<br>а) Ключевые слова.<br>б) Пробелы.<br>в) Комментарии.<br>г) Идентификаторы<br>д) Константы.<br>е) Операторы.<br>ж) Разделители. | ПК-4.З.1<br>ПК-4.В.1 |
| 4. | Задание открытого типа<br>*Инструкция: Прочитайте текст и запишите ответ.*<br>Дана грамматика в форме Бэкуса-Наура:<br>G ({0, 1, +, -, '.'}, {S, N, D, Z}, **P**, S)<br>**P**: $S \rightarrow D \mid N$<br>$N \rightarrow D. \mid N0 \mid N1$<br>$D \rightarrow Z0 \mid Z1 \mid D0 \mid D1$<br>$Z \rightarrow + \mid - \mid \lambda$<br>Определите и запишите тип этой грамматики по классификации Хомского. | ПК-4.З.1<br>ПК-4.В.1 |

<table>
<tr><th>№ п/п</th><th>Примерный перечень вопросов для тестов</th><th>Код индикатора</th></tr>
<tr><td>5.</td><td>Задание открытого типа<br>Инструкция: Прочитайте текст и запишите ответ.<br>Дана грамматика в форме Бэкуса-Наура:<br>G ({a, n, d, o, r, b, (, )}, {S, T, F}, P, S)<br>P: S → S and T | T<br>T → T or F | F<br>F → b | (S)<br>Определите и запишите тип этой грамматики по классификации Хомского.</td><td>ПК-4.З.1<br>ПК-4.В.1</td></tr>
<tr><td>6.</td><td>Задание закрытого типа на установление соответствия<br>Инструкция: Прочитайте текст и установите соответствие. К каждой позиции, данной в левом столбце, подберите соответствующую позицию в правом столбце.<br>Укажите тип распознавателя, используемый для каждого типа языков по классификации Хомского:
<table>
<tr><th>Тип языков</th><th>Тип распознавателя</th></tr>
<tr><td>А. Фразовые</td><td>1. Конечный автомат (КА)</td></tr>
<tr><td>Б. Контекстно-зависимые</td><td>2. Двусторонний недетерминированный автомат с неограниченной внешней памятью</td></tr>
<tr><td>В. Контекстно-свободные</td><td>3. Двусторонний недетерминированный автомат с линейно ограниченной внешней памятью</td></tr>
<tr><td>Г. Регулярные</td><td>4. Автомат с магазинной памятью (МП-автомат)</td></tr>
</table>
Запишите выбранные цифры под соответствующими буквами:
<table>
<tr><td>А.</td><td>Б.</td><td>В.</td><td>Г.</td></tr>
<tr><td></td><td></td><td></td><td></td></tr>
</table></td><td>ПК-4.З.1<br>ПК-4.У.1</td></tr>
<tr><td>7.</td><td>Задание закрытого типа на установление последовательности<br>Инструкция: Прочитайте текст и установите последовательность.<br>Расположите типы грамматик по классификации Хомского в порядке возрастания их сложности:<br>а) Регулярные.<br>б) Контекстно-свободные.<br>в) Фразовые.<br>г) Контекстно-зависимые.<br>Запишите соответствующую последовательность букв слева направо:
<table>
<tr><td></td><td></td><td></td><td></td></tr>
</table></td><td>ПК-4.У.1</td></tr>
</table>

| № п/п | Примерный перечень вопросов для тестов | Код индикатора |
|---|---|---|
| 8. | Задание комбинированного типа с выбором нескольких верных ответов из предложенных<br>*Инструкция: Прочитайте текст и выберите несколько правильных ответов.*<br>Укажите, какие из перечисленных ниже цепочки символов соответствуют регулярному языку, заданному регулярным выражением $a^*ba^*$:<br>а) *aaba*.<br>б) *baaa*.<br>в) *abba*.<br>*г) aaa*<br>д) *aaab*.<br>е) *aaabdaaa*. | ПК-4.З.1 |
| 9. | Задание закрытого типа на установление последовательности<br>*Инструкция: Прочитайте текст и установите последовательность.*<br>Расположите типы LR-грамматик в порядке возрастания сложности распознавателей для них:<br>а) LR(1).<br>б) LR(0).<br>в) LALR(1).<br>г) SLR(1).<br>Запишите соответствующую последовательность букв слева направо:<br>\| \| \| \| \| | ПК-4.У.1 |
| 10. | Задание открытого типа<br>*Инструкция: Прочитайте текст и запишите ответ.*<br>Дана последовательность операций в форме триад:<br>1. + (6, 4)<br>2. * (10, 4)<br>3. := (A, ^1)<br>4. + (A, ^2)<br>5. * (^1, ^4)<br>6. := (A, ^5)<br>Определите и запишите, какое значение в результате выполнения этой последовательности будет присвоено переменной A. | ПК-4.В.1 |
| 11. | Задание открытого типа<br>*Инструкция: Прочитайте текст и запишите ответ.*<br>Дана последовательность операций в форме обратной постфиксной записи:<br>6 4 + 10 / 20 2 / +<br>Посчитайте и запишите результат вычисления этой последовательности операций. | ПК-4.В.1 |

| № п/п | Примерный перечень вопросов для тестов | Код индикатора |
|---|---|---|
| 12. | Задание открытого типа<br>*Инструкция: Прочитайте текст и запишите ответ.*<br>Дана последовательность операций на языке Паскаль:<br>A := A*B + C/D;<br>B := A*B + C/D;<br>C := A*B + C/D;<br>D := A*B + C/D;<br>Укажите количество лишних операций в этой последовательности. | ПК-4.В.1 |
| 13. | Задание открытого типа<br>*Инструкция: Прочитайте текст и запишите ответ.*<br>Дана последовательность операций на языке Паскаль:<br>A := 4*5 + 10/2;<br>B := A*4 - A/25;<br>Посчитайте, какое количество операций останется в этой последовательности после выполнения свертки объектного кода (с учетом предопределенных значений переменных). | ПК-4.В.1 |
| 14. | Задание открытого типа<br>*Инструкция: Прочитайте текст и запишите ответ.*<br>Дано описание типов на языке Паскаль:<br>`type`<br>`TArray = packed array [1..10] of byte;`<br>`TRecord = packed record a : TArray; i : integer;`<br>`end;`<br>`var`<br>`a, b, c : byte;`<br>`a1 : TArray;`<br>`r1, r2 : TRecord;`<br>Посчитайте и запишите, какое количество памяти (в байтах) будет выделено под все перечисленные переменные без учета кратности распределения памяти, если размер типа «`byte`» равен 1 байт, а размер типа «`integer`» - 4 байта. | ПК-4.В.1 |
| 15. | Задание открытого типа<br>*Инструкция: Прочитайте текст и запишите ответ.*<br>Дано описание типов на языке Паскаль:<br>`type`<br>`TArray = packed array [1..10] of byte;`<br>`TRecord = packed record a : TArray; i : integer;`<br>`end;`<br>`var`<br>`a, b, c : byte;`<br>`a1 : TArray;`<br>`r1, r2 : TRecord;`<br>Посчитайте и запишите, какое количество памяти (в байтах) будет выделено под все перечисленные переменные при кратности распределения памяти равной 4 байта, если размер типа «`byte`» равен 1 байт, а размер типа «`integer`» - 4 байта. | ПК-4.В.1 |

<table>
<tr><th>№ п/п</th><th>Примерный перечень вопросов для тестов</th><th>Код индикатора</th></tr>
<tr><td>16.</td><td>Задание закрытого типа на установление последовательности<br>Инструкция: Прочитайте текст и установите последовательность.<br>Расположите составляющие части системы программирования в порядке выполнения ими обработки данных:<br>а) Компоновщик.<br>б) Текстовый редактор.<br>в) Загрузчик.<br>г) Компилятор.<br>Запишите соответствующую последовательность букв слева направо:<br><table><tr><td></td><td></td><td></td><td></td></tr></table></td><td>ПК-4.З.2<br>ПК-4.В.1</td></tr>
<tr><td>17.</td><td>Задание закрытого типа на установление соответствия<br>Инструкция: Прочитайте текст и установите соответствие. К каждой позиции, данной в левом столбце, подберите соответствующую позицию в правом столбце.<br>Укажите, какие входные данные обрабатываются каждым из указанных компонентов системы программирования и операционной системы:<br><table><tr><th>Входные данные</th><th>Обработчик</th></tr><tr><td>А. Описание ресурсов пользовательского интерфейса</td><td>1. Компоновщик</td></tr><tr><td>Б. Объектный код</td><td>2. Компилятор ресурсов</td></tr><tr><td>В. Исходный код</td><td>3. Настраивающий загрузчик</td></tr><tr><td>Г. Исполняемый код</td><td>4. Компилятор входного языка</td></tr></table>Запишите выбранные цифры под соответствующими буквами:<br><table><tr><td>А.</td><td>Б.</td><td>В.</td><td>Г.</td></tr><tr><td></td><td></td><td></td><td></td></tr></table></td><td>ПК-4.З.2<br>ПК-4.В.1</td></tr>
<tr><td>18.</td><td>Задание открытого типа<br>Инструкция: Прочитайте текст и запишите ответ.<br>Укажите, какое количество серверных компонент используется в приложении, построенном на основе двухуровневой архитектуры организации распределенных вычислений «клиент-сервер», развернутом с применением терминального доступа.</td><td>ПК-4.З.2<br>ПК-4.У.1</td></tr>
<tr><td>19.</td><td>Задание с выбором одного верного ответа из предложенных<br>Инструкция: Прочитайте текст и выберите один правильный ответ.<br>Выберите термин, наиболее точно соответствующий определению «программный код, загружаемый во время выполнения по мере необходимости»:<br>а) Объектный код.<br>б) Статические библиотеки.<br>в) Динамические библиотеки.<br>г) Ресурсы пользовательского интерфейса.</td><td>ПК-4.З.2</td></tr>
<tr><td>20.</td><td>Задание с выбором одного верного ответа из предложенных<br>Инструкция: Прочитайте текст и выберите один правильный</td><td>ПК-4.У.1<br>ПК-4.З.2</td></tr>
</table>

| № п/п | Примерный перечень вопросов для тестов | Код индикатора |
|---|---|---|
| | *ответ.*<br>Выберите термин, наиболее точно соответствующий определению «данные, используемые приложением для организации взаимодействия с пользователем, но не влияющие непосредственно на логику выполнения приложения»:<br>а) Объектный код.<br>б) Статические библиотеки.<br>в) Динамические библиотеки.<br>г) Ресурсы пользовательского интерфейса. | |
| 21. | Задание комбинированного типа с выбором нескольких верных ответов из предложенных<br>*Инструкция: Прочитайте текст и выберите несколько правильных ответов.*<br>Укажите, какие данные обрабатывает компоновщик:<br>а) Исходный код.<br>б) Объектный код.<br>в) Динамические библиотеки.<br>г) Статические библиотеки.<br>д) Результирующий код. | ПК-4.У.1<br>ПК-4.З.2 |
| 22. | Задание комбинированного типа с выбором нескольких верных ответов из предложенных<br>*Инструкция: Прочитайте текст и выберите несколько правильных ответов.*<br>Укажите, какие компоненты входят в состав приложений, построенных на основе комплексной двухуровневой архитектуры организации распределенных вычислений «клиент-сервер»:<br>а) Клиентская часть.<br>б) Файловый сервер.<br>в) Сервер данных.<br>г) Терминальный сервер.<br>д) Сервер приложений. | ПК-4.З.2 |
| 23. | Задание комбинированного типа с выбором нескольких верных ответов из предложенных<br>*Инструкция: Прочитайте текст и выберите несколько правильных ответов.*<br>Укажите, какие компоненты входят в состав приложений, построенных на основе комплексной трехуровневой архитектуры организации распределенных вычислений:<br>а) Клиентская часть.<br>б) Файловый сервер.<br>в) Сервер данных.<br>г) Терминальный сервер.<br>д) Сервер приложений. | ПК-4.З.2 |

Примечание: система оценивания тестовых заданий.

| Оценка тестовых заданий балльная шкала | Характеристика заданий |
|---|---|
| Полное совпадение с верным ответом оценивается 1 баллом / неверный ответ или его отсутствие – 0 баллов. | 1 тип)<br>Задание с выбором одного верного ответа из нескольких предложенных считается верным, если правильно указан выбранный вариант |
| Полное совпадение с верным ответом оценивается 1 баллом, если допущены ошибки или ответ отсутствует 0 баллов. | 2 тип)<br>Задание с выбором нескольких верных ответов из предложенных считается верным, если указаны все правильные варианты и не указан ни один неправильный вариант |
| Полное совпадение с верным ответом оценивается 1 баллом, если допущены ошибки или ответ отсутствует 0 баллов. | 3 тип)<br>Задание закрытого типа на установление соответствия считается верным, если установлены все соответствия (позиции из одного столбца верно сопоставлены с позициями другого столбца) |
| Полное совпадение с верным ответом оценивается 1 баллом, если допущены ошибки или ответ отсутствует 0 баллов | 4 тип)<br>Задание закрытого типа на установление последовательности считается верным, если правильно указана вся последовательность пунктов. |
| Правильный ответ за задание оценивается в 3 балла, если допущена неточность \ ответ правильный, но не полный - 1 балл, если ответ неправильный \ ответ отсутствует – 0 баллов». | 5 тип)<br>Задание открытого типа считается верным, если ответ совпадает с эталонным по содержанию и полноте. |

Перечень тем контрольных работ по дисциплине обучающихся заочной формы обучения, представлены в таблице 19.

Таблица 19 – Перечень контрольных работ

| № п/п | Перечень контрольных работ |
|---|---|
| 1 | Построение КА по заданной регулярной грамматике |
| 2 | Заполнение матриц предшествования для заданной КС-грамматики |
| 3 | Построение дерева вывода для заданной цепочки символов языка, определенного с помощью КС-грамматики |
| 4 | Вычисление значений заданных выражений в форме триад |
| 5 | Вычисление значений заданных выражений в форме обратной постфиксной записи |
| 6 | Оптимизация заданной последовательности триад методом свертки объектного кода |
| 7 | Оптимизация заданной последовательности триад методом исключения лишних операций |
| 8 | Рассчитать объем оперативной памяти, необходимой для хранения заданных структур данных (с учетом кратности и без учета кратности) |

10.4. Методические материалы, определяющие процедуры оценивания индикаторов, характеризующих этапы формирования компетенций, содержатся в локальных нормативных актах ГУАП, регламентирующих порядок и процедуру проведения текущего контроля успеваемости и промежуточной аттестации обучающихся ГУАП.

11. Методические указания для обучающихся по освоению дисциплины

11.1. Методические указания для обучающихся по освоению лекционного материала.

Основное назначение лекционного материала – логически стройное, системное, глубокое и ясное изложение учебного материала. Назначение современной лекции в рамках дисциплины не в том, чтобы получить всю информацию по теме, а в освоении фундаментальных проблем дисциплины, методов научного познания, новейших достижений научной мысли. В учебном процессе лекция выполняет методологическую, организационную и информационную функции. Лекция раскрывает понятийный аппарат конкретной области знания, её проблемы, дает цельное представление о дисциплине, показывает взаимосвязь с другими дисциплинами.

Планируемые результаты при освоении обучающимися лекционного материала:

– получение современных, целостных, взаимосвязанных знаний, уровень которых определяется целевой установкой к каждой конкретной теме;

– получение опыта творческой работы совместно с преподавателем;

– развитие профессионально-деловых качеств, любви к предмету и самостоятельного творческого мышления.

– появление необходимого интереса, необходимого для самостоятельной работы;

– получение знаний о современном уровне развития науки и техники и о прогнозе их развития на ближайшие годы;

– научиться методически обрабатывать материал (выделять главные мысли и положения, приходить к конкретным выводам, повторять их в различных формулировках);

– получение точного понимания всех необходимых терминов и понятий.

Лекционный материал может сопровождаться демонстрацией слайдов и использованием раздаточного материала при проведении коротких дискуссий об особенностях применения отдельных тематик по дисциплине.

Лекционный материал может сопровождаться демонстрацией слайдов и использованием раздаточного материала при проведении коротких дискуссий об особенностях применения отдельных тематик по дисциплине.

Структура предоставления лекционного материала:

– Изучение теоретических вопросов, связанных с рассматриваемой темой

– Описание программных средств, методов и алгоритмов, применяемых при разработке системного программного обеспечения

– Демонстрация примеров реализации описанных ранее алгоритмов и использования программных средств

– Обобщение изложенного материала

– Ответы на возникающие вопросы по теме лекции.

Конспекты лекций и указания по их использованию размещаются в личном кабинете преподавателя.

11.2. Методические указания для обучающихся по участию в семинарах

Не предусмотрено

11.3. Методические указания для обучающихся по прохождению практических занятий

Не предусмотрено

11.4. Методические указания для обучающихся по выполнению лабораторных работ

В ходе выполнения лабораторных работ обучающийся должен углубить и закрепить знания, практические навыки, овладеть современной методикой и техникой эксперимента в соответствии с квалификационной характеристикой обучающегося. Выполнение лабораторных работ состоит из экспериментально-практической, расчетно-аналитической частей и контрольных мероприятий.

Выполнение лабораторных работ обучающимся является неотъемлемой частью изучения дисциплины, определяемой учебным планом, и относится к средствам, обеспечивающим решение следующих основных задач обучающегося:

– приобретение навыков исследования процессов, явлений и объектов, изучаемых в рамках данной дисциплины;

– закрепление, развитие и детализация теоретических знаний, полученных на лекциях;

– получение новой информации по изучаемой дисциплине;

– приобретение навыков самостоятельной работы с лабораторным оборудованием и приборами.

Задание и требования к проведению лабораторных работ

Задания на выполнение лабораторных работ и методические указания по их выполнению размещаются в личном кабинете преподавателя.

Вариант задания по каждой лабораторной работе обучающийся получает в соответствии с номером в списке группы. Перед началом выполнения лабораторной работы обучающемуся следует внимательно ознакомиться с методическими указаниями по ее выполнению. В соответствии с заданием обучающийся должен подготовить необходимые данные, получить от преподавателя допуск к выполнению лабораторной работы, разработать и отладить программный продукт, в результате выполнения разработанного программного продукта получить требуемые результаты, продемонстрировать их преподавателю, оформить и защитить отчет по лабораторной работе.

Лабораторные работы должны выполняться в вычислительной лаборатории или обучающимися самостоятельно на личных компьютерах. Результатом выполнения каждой лабораторной работы должна быть компьютерная программа, разработанная в соответствии с заданием.

По результату выполнения каждой лабораторной работы обучающийся готовит отчет, который публикуется в личном кабинете и проверяется преподавателем. После проверки отчета результат выполнения каждой лабораторной работы предъявляется преподавателю, который оценивает соответствие функционала разработанной программы заданию на выполнение лабораторной работы.

Структура и форма отчета о лабораторной работе

Отчет о лабораторной работе должен включать в себя:

– титульный лист;
– полную формулировку задания в соответствии с полученным вариантом;
– теоретические сведения, используемые при выполнении лабораторной работы;
– результаты выполненных предварительных расчетов;
– краткое описание использованных средств разработки;
– полученные результаты и выводы.

Требования к оформлению отчета о лабораторной работе

По каждой лабораторной работе выполняется отдельный отчет. В соответствии с методическими указаниями, допускается составлять единый отчет для 2 лабораторных работ, объединенных одной темой.

Титульный лист оформляется в соответствии с шаблоном (образцом) приведенным на сайте ГУАП (www.guap.ru) в разделе «Сектор нормативной документации». Текстовые и графические материалы оформляются в соответствии с действующими ГОСТами и требованиями, приведенными на сайте ГУАП (www.guap.ru) в разделе «Сектор нормативной документации».

Методические указания по прохождению лабораторных работ:

1. Системное программное обеспечение: лабораторный практикум / А. Ю. Молчанов; С.-Петерб. гос. ун-т аэрокосм. приборостроения. - СПб.: Изд-во ГУАП, 2014. - 131 с.: рис., табл.
2. Системное программное обеспечение: методические указания к выполнению лабораторных работ / С.-Петерб. гос. ун-т аэрокосм. приборостроения; сост. Т. Л. Прокофьева. - СПб. : Изд-во ГУАП, 2012. - 67 с.: рис.

11.5. Методические указания для обучающихся по прохождению курсового проектирования/выполнения курсовой работы

Не предусмотрено

11.6. Методические указания для обучающихся по прохождению самостоятельной работы

В ходе выполнения самостоятельной работы, обучающийся выполняет работу по заданию и при методическом руководстве преподавателя, но без его непосредственного участия.

Для обучающихся по заочной форме обучения, самостоятельная работа может включать в себя контрольную работу.

В процессе выполнения самостоятельной работы, у обучающегося формируется целесообразное планирование рабочего времени, которое позволяет им развивать умения и навыки в усвоении и систематизации приобретаемых знаний, обеспечивает высокий уровень успеваемости в период обучения, помогает получить навыки повышения профессионального уровня.

Методическими материалами, направляющими самостоятельную работу обучающихся являются:

- учебно-методический материал по дисциплине;
- методические указания по выполнению контрольных работ (для обучающихся по заочной форме обучения).

11.7. Методические указания для обучающихся по прохождению текущего контроля успеваемости.

Текущий контроль успеваемости предусматривает контроль качества знаний обучающихся, осуществляемого в течение семестра с целью оценивания хода освоения дисциплины.

Текущий контроль успеваемости проводится в форме устной беседы и выполнения тестов после предоставления обучающимся отчетов по следующим лабораторным работам:

- лабораторной работы «Работа с таблицами идентификаторов» в семестре 7;
- лабораторных работ «Построение распознавателя для регулярной грамматики» и «Построение лексического анализатора» в семестре 7;

- лабораторных работ «Построение синтаксического анализатора» и «Построение простейшего дерева вывода» в семестре 7;
- лабораторных работ «Генерация объектного кода» и «Оптимизация объектного кода» в семестре 8;
- лабораторных работ «Построение распознавателя для анализа описаний типов данных и переменных» и «Определение объема памяти для заданных структур данных» в семестре 8;
- лабораторной работы «Разработка простейшего приложения на основе технологии "клиент-сервер"» в семестре 8.

Итого предусмотрено 3 этапа текущего контроля успеваемости в семестре 6 и 3 этапа текущего контроля успеваемости в семестре 7.

Результаты промежуточного контроля успеваемости фиксируются в электронном журнале в личном кабинете преподавателя вместе с отчетами о выполнении лабораторных работ в виде баллов, полученных за выполнение каждой лабораторной работы. Сумма баллов, набранная обучающимся за семестр, принимается во внимание при проведении промежуточной аттестации.

Максимальные баллы по результатам контроля успеваемости распределяются следующим образом:

| Этапы промежуточного контроля успеваемости | Семестр 7 | Семестр 8 |
|---|---|---|
| 1 | 20 | 30 |
| 2 | 35 | 30 |
| 3 | 45 | 40 |
| Итого | 100 | 100 |

11.8. Методические указания для обучающихся по прохождению промежуточной аттестации.

Промежуточная аттестация обучающихся предусматривает оценивание промежуточных и окончательных результатов обучения по дисциплине. Она включает в себя:

– экзамен – форма оценки знаний, полученных обучающимся в процессе изучения всей дисциплины или ее части, навыков самостоятельной работы, способности применять их для решения практических задач. Экзамен, как правило, проводится в период экзаменационной сессии и завершается аттестационной оценкой «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

– дифференцированный зачет – это форма оценки знаний, полученных обучающимся при изучении дисциплины, при выполнении курсовых проектов, курсовых работ, научно-исследовательских работ и прохождении практик с аттестационной оценкой «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Система оценок при проведении промежуточной аттестации осуществляется в соответствии с требованиями Положений «О текущем контроле успеваемости и промежуточной аттестации студентов ГУАП, обучающихся по программы высшего образования» и «О модульно-рейтинговой системе оценки качества учебной работы студентов в ГУАП».

В случае невыполнения или неполного выполнения требований текущего контроля успеваемости предусмотрены следующие ограничения по результатам промежуточной аттестации:

- В случае выполнения и успешной сдачи менее 4, но более 1 лабораторных работ из предусмотренных планом на семестр 7 обучающийся при успешном прохождении промежуточной аттестации в форме дифференцированного зачета не может получить аттестационную оценку выше «хорошо».

- В случае выполнения и успешной сдачи менее 2 лабораторных работ из предусмотренных планом на семестр 7 обучающийся при успешном прохождении промежуточной аттестации в форме дифференцированного зачета не может получить аттестационную оценку выше «удовлетворительно».
- В случае выполнения и успешной сдачи менее 4, но более 1 лабораторных работ из предусмотренных планом на семестр 8 обучающийся при успешном прохождении промежуточной аттестации в форме экзамена не может получить аттестационную оценку выше «хорошо».
- В случае выполнения и неуспешной сдачи менее 2 лабораторных работ из предусмотренных планом на семестр 8 обучающийся при успешном прохождении промежуточной аттестации в форме дифференцированного зачета не может получить аттестационную оценку выше «удовлетворительно».

В случае успешного выполнения всех лабораторных работ в заданные сроки и в соответствии с планом обучения, а также при условии успешного прохождения всех этапов текущего контроля успеваемости, обучающийся имеет возможность получить оценку по промежуточной аттестации на основании суммы баллов, набранных по результатам прохождения всех этапов текущего контроля успеваемости.

Оценка определяется следующим образом:

| Сумма набранных баллов | Оценка |
|---|---|
| От 55 до 69 включительно | «удовлетворительно» |
| От 70 до 84 включительно | «хорошо» |
| От 85 до 100 включительно | «отлично» |

Лист внесения изменений в рабочую программу дисциплины

| Дата внесения изменений и дополнений. Подпись внесшего изменения | Содержание изменений и дополнений | Дата и № протокола заседания кафедры | Подпись зав. кафедрой |
|---|---|---|---|
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |