Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
**«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
(ВлГУ)**

Институт информационных технологий и радиоэлектроники

УТВЕРЖДАЮ:
Директор института
_____ Галкин А.А.
«31» 08 2021 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Системное программное обеспечение

направление подготовки / специальность
09.03.01 Информатика и вычислительная техника

направленность (профиль) подготовки
Высокопроизводительные и распределенные вычисления

г. Владимир

2021 Год

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины «Системное программное обеспечение» является изучение основных принципов построения и использования системных программных продуктов.

Задачи:

1) обучение основам построения компиляторов, общим принципам программирования, представления основных структур программ и данных;

2) создание фундаментальной основы знаний, необходимой при проектировании программных продуктов для вычислительных систем;

3) изучение основных этапов и фаз построения компиляторов;

4) изучение синтаксических и семантических конструкций языков программирования высокого уровня;

5) освоение студентами основных приемов программирования типовых задач (списки, стеки, очереди, деревья);

6) изучение особенностей реализации обработки различных типов данных;

7) освоение студентами соответствия между операторами языков высокого уровня и командами языка ассемблера (фаза генерации кода).

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина «Системное программное обеспечение» относится к части, формируемой участниками образовательных отношений.

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОПОП (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине, в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции *(код, содержание индикатора*	Результаты обучения по дисциплине	
ПК-3 Способен разрабатывать компоненты системных программных продуктов	ПК-3.1 Знает основные концепции системного программирования; структуры данных, принципы построения трансляторов, основные этапы и фазы процесса компиляции ПК-3.2 Умеет осваивать методики использования программных средств для решения практических задач; обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности; разрабатывать компоненты системных программных продуктов ПК-3.3 Владеет современными инструментальными средствами и технологиями программирования	Знать основные структуры данных, принципы построения трансляторов Уметь применять программные средства для решения практических задач, разрабатывать компоненты системных программных продуктов Владеть современными инструментальными средствами и технологиями программирования	Тестовые вопросы Практико-ориентированное задание Курсовая работа

4. ОБЪЕМ И СТРУКТУРА ДИСЦИПЛИНЫ

Трудоемкость дисциплины составляет 9 зачетных единиц, 324 часа

Тематический план
форма обучения – очная

№ п/п	Наименование тем и/или разделов/тем дисциплины	Семестр	Неделя семестра	Контактная работа обучающихся с педагогическим работником				Самостоятельная работа	Формы текущего контроля успеваемости, форма промежуточной аттестации *(по семестрам)*
				Лекции	Практические занятия	Лабораторные работы	*в форме практической подготовки*		
1	Структуры данных. Списки, стеки, очереди, деревья	4	1-8	8		16		16	Рейтинг-контроль №1
2	Основные принципы построения трансляторов	4	9-10	2				4	
3	Формальные языки и формальные грамматики	4	11-14	4				40	Рейтинг-контроль №2
4	Лексический анализ. Таблица идентификаторов	4	15-18	4		2		12	Рейтинг-контроль №3
Всего за 4 семестр:				18		18		72	Экзамен
5	Синтаксический анализ.	5	1-6	6	6	6		40	Рейтинг-контроль №1
6	Восходящий и нисходящий анализ. МП-автоматы.	5	7-12	6	4	4		38	Рейтинг-контроль №2
7	Семантический анализ	5	13-14	2	2	2		24	
8	Генерация и оптимизация кода	5	15-18	4	6	6		24	Рейтинг-контроль №3
Всего за 5 семестр:				18	18	18		126	Зачет, курсовая работа
Наличие в дисциплине КП/КР									Курсовая работа
Итого по дисциплине				36	18	36		198	Экзамен, курсовая работа, зачет

Содержание лекционных занятий по дисциплине

Раздел 1. Структуры данных. Списки, стеки, очереди, деревья
Раздел 2. Основные принципы построения трансляторов
Раздел 3. Формальные языки и формальные грамматики.
Раздел 4. Лексический анализ. Таблица идентификаторов
Раздел 5. Синтаксический анализ
Раздел 6. Восходящий и нисходящий анализ. МП-автоматы
Раздел 7. Семантический анализ
Раздел 8. Генерация и оптимизация кода

Содержание лабораторных занятий по дисциплине (семестр 4)

1. Списки. Представление однородными массивами фиксированного размера.
2. Списки. Реализация с помощью динамических переменных.
3. Бинарные деревья. Итеративный алгоритм обработки.
4. Бинарные деревья. Рекурсивный алгоритм обработки.

Содержание лабораторных занятий по дисциплине (семестр 5)

1. Лексический анализ.
2. Синтаксический анализ.
3. Семантический анализ.
4. Генерация и оптимизация кода.

Содержание практических занятий по дисциплине (семестр 5)

1. Лексема. Таблица лексем. Преобразование программы на ЯВУ в список лексем.
2. Лексический анализ. Сканер на основе детерминированного автомата.
3. Сканер на основе инженерной импровизации.
4. Диаграммы Вирта. БНФ. РБНФ.
5. Грамматика арифметических выражений. Нисходящий и восходящий синтаксический анализ. Дерево разбора.
6. Внутреннее представление программы: обратная польская запись, семантическое дерево (дерево операций).
7. Грамматики с левосторонней и правосторонней рекурсией. Процедура синтаксического анализа.
8. Процедура генерации кода. Вычисление значения выражения по дереву операций.

5. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

5.1. Текущий контроль успеваемости

(семестр 4)

Вопросы рейтинг-контроля №1

1. Перечислите критерии классификации структур данных
2. Перечислите операции для работы со списками
3. Опишите структуры данных, которые можно построить на основе связного списка. Назовите преимущества и недостатки таких структур.
4. Перечислите условия (или предметные области) оптимального использования массивов. Докажите свою точку зрения.
5. Перечислите условия (или предметные области) оптимального использования связных списков. Докажите свою точку зрения.

Вопросы рейтинг-контроля №2

1. Перечислите операции, которые можно выполнять со строками.
2. Что такое "семантика языка"?
3. Перечислите методы задания языков.
4. Докажите, верно или нет следующее тождество:

$$(\alpha^2\beta^2)^R = (\beta^R\alpha^R)^2$$

5. Докажите, верно или нет следующее тождество:

$$(\alpha^2\beta^2)^R = (\beta^R)^2(\alpha^R)^2$$

6. Почему метод перечисления всех строк языка очень редко применяется на практике? В каких случаях этот метод может применяться?

Вопросы рейтинг-контроля №3

1. Транслятор, компилятор, интерпретатор. Особенности и отличия.
2. Общая схема работы компилятора. Фазы компиляции.

3. Компилятор можно построить без лексического анализатора (ЛА). Почему большинство современных компиляторов все-таки используют фазу ЛА?

4. Какую роль выполняет ЛА в процессе компиляции? Как могут быть связаны между собой лексический и синтаксический анализы?

5. Что такое таблица идентификаторов (ТИ)? Для чего она предназначена? Какие существуют способы организации ТИ?

6. Какие цели преследуются при организации таблиц идентификаторов (ТИ)? Исходя из каких характеристик оценивается эффективность того или иного метода организации ТИ?

7. Чем различаются таблица лексем и ТИ?

8. Что такое хеш-функция и хеш-адресация? Для чего они используются?

9. Что такое коллизия? Почему она происходит при использовании хеш-функции? В чём заключаются преимущества и недостатки метода цепочек?

10. Как могут быть скомбинированы различные методы организации ТИ?

(семестр 5)

Вопросы рейтинг-контроля №1

1. Фаза синтаксического анализа. Задачи синтаксического анализатора (СА).
2. Виды СА.
3. Дайте характеристику восходящего СА.
4. Дайте характеристику нисходящего СА.
5. Для каких грамматик применим синтаксический анализ методом рекурсивного спуска? Опишите этот метод.
6. Для каких грамматик применим синтаксический анализ методом «сдвиг-свёртка»? Опишите этот метод.
7. Обработка синтаксических ошибок. Виды ошибок.
8. Стратегии восстановления после ошибок.
9. Способы внутреннего представления входной программы.

Вопросы рейтинг-контроля №2

1. Фаза семантического анализа.
2. Дана LL(1)-грамматика

G'({+,–,/,*,a,b},{S,R,T,F,E},**P'**,S) с правилами:

P': S → TR
R → λ | +TR | -TR
T → EF
F → λ | *EF | /EF
E → (S) | a | b

Сгенерировать множество FIRST для этой грамматики. Показать все шаги алгоритма.

3. Дана LL(1)-грамматика

G'({+,–,/,*,a,b},{S,R,T,F,E},**P'**,S) с правилами:

P': S → TR
R → λ | +TR | -TR
T → EF
F → λ | *EF | /EF
E → (S) | a | b

Сгенерировать множество FOLLOW для этой грамматики. Показать все шаги алгоритма.

4. Выполнить разбор строки

вариант 1: a – b + (a * b)
вариант 2: a * b – (a + b)
вариант 3: a / b + (a – b)

для грамматики, используя множества FIRST и FOLLOW.

G'({+,–,/,*,a,b},{S,R,T,F,E},**P'**,S) с правилами:

P': S → TR

R → λ | +TR | -TR

T → EF

F → λ | *EF | /EF

E → (S) | a | b

Вопросы рейтинг-контроля №3

1. Фаза генерации кода. Общие принципы генерации кода.
2. Суть и принципы синтаксически управляемого (СУ)-перевода.
3. Приведите пример генерация кода по дереву операций.
4. Цели фазы оптимизации кода.
5. Машинно-независимые и машинно-зависимые методы оптимизации.
6. Перечислите возможные операции по оптимизации линейных участков программы.

5.2. Промежуточная аттестация

Вопросы к экзамену (4 семестр**)**

1. Классификация структур данных. Признаки классификации.
2. Линейные списки: операции и формы представления.
3. Стеки: операции и формы представления.
4. Очереди: операции и формы представления.
5. Линейные списки специального вида и операции с ними.
6. Многомерные массивы переменного размера: графы.
7. Многомерные массивы переменного размера: структуры.
8. Многомерные массивы переменного размера: деревья.
9. Деревья: формы представления.
10. Бинарные деревья и процедуры сканирования бинарных деревьев.
11. Представление деревьев в системе «отец-сын-брат»
12. Формы записи выражений. Обратная польская запись.
13. Применение стека для управления памятью в ЯВУ.
14. Рекурсия. Примеры использования рекурсии.
15. Применение связанных списков для управления памятью.
16. Основные принципы построения компиляторов. Определения.
17. Транслятор, компилятор, интерпретатор – отличия и особенности.
18. Фазы компиляции.
19. Этап лексического анализа.
20. Этап синтаксического анализа.
21. Таблица идентификаторов.
22. Этап семантического анализа.
23. Формальные языки. Основные определения.
24. Формальные грамматики. Основные определения.
25. Классификация грамматик Хомского.
26. Регулярные грамматики для констант и идентификаторов.
27. Форма Бекуса-Наура (БНФ).
28. Расширенная форма Бэкуса-Наура (РБНФ).

Вопросы на зачет *(семестр 5)*

1. Построение лексического анализатора. Таблицы идентификаторов.
2. Построения синтаксического анализатора.
3. Построение генератора кода.
4. Лексема. Таблица лексем. Преобразование программы на ЯВУ в список лексем.
5. Лексический анализ. Сканер на основе детерминированного автомата.
6. Сканер на основе инженерной импровизации.
7. Диаграммы Вирта. БНФ. РБНФ.
8. Грамматика арифметических выражений. Нисходящий и восходящий синтаксический анализ. Дерево разбора.
9. Внутреннее представление программы: обратная польская запись, семантическое дерево (дерево операций).
10. Грамматики с левосторонней и правосторонней рекурсией. Процедура синтаксического анализа.

11. Процедура генерации кода. Вычисление значения выражения по дереву операций.

5.3. Самостоятельная работа обучающегося.

Самостоятельная работа заключается в изучении содержания тем курса по конспектам, учебникам и дополнительной литературе, подготовке к лабораторным и практическим занятиям, выполнении заданий для самостоятельной работы, оформлении отчетов по лабораторным работам, подготовке к промежуточной аттестации.

Задания для самостоятельной работы студентов (семестр 4)

1. Ознакомиться с основными принципами построения трансляторов.
2. Рассмотреть различные структуры данных.
3. Разобраться в использовании бинарных деревьев.
4. Изучить теорию и практику лексического анализа.
5. Изучить вопрос использования таблицы лексем.

Задания для самостоятельной работы студентов (семестр 5)

1. Познакомиться со структурой современной системы программирования
2. Изучить принципы функционирования систем программирования
3. Понять основные принципы построения трансляторов.
4. Разобраться в вопросе проектирования компилятора
5. Выполнить внешний проект
6. Выполнить проект архитектуры
7. Разработать синтаксический анализатор
8. Изучить трансляторы с языка ассемблера
9. Изучить средства ассемблера
10. Спроектировать генератор кода

Темы заданий на курсовую работу

Типовое задание на курсовую работу предполагает разработку системы компиляции. Входным языком является подмножество языка Pascal, определяемое индивидуальным вариантом.

В ходе выполнения курсовой работы должен сформироваться проект. Проект – система документов, определяющая процесс создания какого-либо продукта (его бумажная или электронная модель). Здесь за основу принят структурный подход с его методом пошаговой детализации, что вполне целесообразно в применении к программным системам (ПС) малого и даже среднего размера.

Варианты индивидуальных заданий для курсового проектирования (характеризует набор конструкций входного языка):

1. Массив, FOR, IF, BREAKE
2. Массив, WHILE, сравнение.
3. Массив, REPEAT, сравнение.
4. Метка, IF, сравнение, GOTO.
5. Константа, CASE, GOTO, метка.
6. Процедура, EXIT, передача данных по указателю, константа.
7. Функция, передача данных по указателю, константа.
8. IF, сравнение, OR, AND.
9. IF, константа, XOR, NAND.

Фонд оценочных материалов (ФОМ) для проведения аттестации уровня сформированности компетенций обучающихся по дисциплине оформляется отдельным документом.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1. Книгообеспеченность

Наименование литературы: автор, название, вид издания, издательство	Год издания	КНИГООБЕСПЕЧЕННОСТЬ
		Наличие в электронном каталоге ЭБС
Основная литература		
1. Язык C#. Базовый курс [Электронный ресурс] : учеб. пособие / В.В. Подбельский. - 2-е изд., перераб. и доп. - М. : Финансы и статистика,.	2015	http://www.studentlibrary.ru/book/ISBN9785279035342.html
2. Электронное издание на основе: Разработка Паскаль-компилятора [Электронный ресурс] / Л.А. Залогова. - 4-е изд. (эл.).- М. : Лаборатория знаний,	2017	https://www.studentlibrary.ru/ru/book/ISBN9785001014508.html
3. Построение компиляторов / Никлаус Вирт - Москва : ДМК Пресс, 2010. - ISBN 978-5-94074-585-3. - Текст : электронный	2010	https://www.studentlibrary.ru/ru/book/ISBN9785940745853.html
Дополнительная литература		
1. Объектно-ориентированное программирование в C++: лекции и упражнения [Электронный ресурс] : Учебное пособие для вузов / Ашарина И.В. - М. : Горячая линия - Телеком,.	2012	http://www.studentlibrary.ru/book/ISBN9785991270014.html
2. Системное и прикладное программное обеспечение[Электронный ресурс] : Учебное пособие / Иванова Н.Ю., Маняхина В.Г. - М. : Прометей,.	2011	http://www.studentlibrary.ru/book/ISBN9785426300781.html
3. Основы программирования в ядре операционной системы GNU/Linux [Электронный ресурс] : Учеб. пособие / В.А. Крищенко, Н.Ю. Рязанова. - М. : Издательство МГТУ им. Н. Э. Баумана,.	2010	http://www.studentlibrary.ru/book/bauman_0335.html
4. Теория и реализация языков программирования [Электронный ресурс] / Серебряков В.А. - М.: ФИЗМАТЛИТ,.	2012	http://www.studentlibrary.ru/book/ISBN9785922114172.html

6.2. Периодические издания

Журналы (https://elibrary.ru/):

1. Вестник компьютерных и информационных технологий

2. Вычислительные технологии

6.3. Интернет-ресурсы

http://www.studentlibrary.ru

http://library.vlsu.ru/

http://citforum.ru/

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Для реализации данной дисциплины имеются специальные помещения для проведения занятий *лекционного типа, занятий практического/лабораторного типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы*. Практические/лабораторные работы проводятся в аудиториях 424-2, 401-2, 412-2 и 416-2.

Рабочую программу составил ст. преподаватель каф. ВТиСУ Трофимов М.А.

Рецензент
(представитель работодателя) _____ Генеральный директор ООО "Диаграмма" Протягов И.В.

Программа рассмотрена и одобрена на заседании кафедры ВТ и СУ
Протокол № 1 от 31 августа 2021 года
Заведующий кафедрой Ланцов В.Н._____

Рабочая программа рассмотрена и одобрена
на заседании учебно-методической комиссии направления 09.03.01 информатика и вычислительная техника
Протокол № 1 от 31 августа 2021 года
Председатель комиссии Ланцов В.Н. зав. каф. ВТиСУ_____

ЛИСТ ПЕРЕУТВЕРЖДЕНИЯ
РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ

Рабочая программа одобрена на 2022 / 2023 учебный года

Протокол заседания кафедры № 1 от 29.08.22 года

Заведующий кафедрой Куликов К.В.

Рабочая программа одобрена на 20____ / 20____ учебный года

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой_____

Рабочая программа одобрена на 20____ / 20____ учебный года

Протокол заседания кафедры № _____ от _____ года

Заведующий кафедрой_____