

Документ подписан простой электронной подписью Информация о владельце: ФИО: Таскаев Сергей Валерьевич Должность: Ректор Дата подписания: 04.06.2025 12:35:17 Уникальный программный ключ: 04c19ed8bf108f7b6cb77a486b9a8788b8722727	МИНОВЕРНАУКИ РОССИИ Федеральное государственное бюджетное образовательное учреждение высшего образования «Челябинский государственный университет» (ФГБОУ ВО «ЧелГУ»)	Рабочая программа дисциплины "Системное программирование" по направлению подготовки (специальности) 10.05.01 "Компьютерная безопасность" направленности (профилю) специализация N 6 "Информационно-аналитическая и техническая экспертиза компьютерных систем" ФГБОУ ВО «ЧелГУ»	стр. 1
--	--	---	--------

Рабочая программа дисциплины (модуля)* Системное программирование

Направление подготовки (специальность)

10.05.01 Компьютерная безопасность

Направленность (профиль)

специализация N 6 "Информационно-аналитическая и техническая экспертиза компьютерных систем"

Присваиваемая квалификация (степень)

специалист по защите информации

Форма обучения

очная

Год(ы) набора 2025

*Рабочая программа дисциплины (модуля) адаптирована для инклюзивного обучения инвалидов и лиц с ограниченными возможностями здоровья

Челябинск 2025 г.



Содержание

1. Цели освоения дисциплины
2. Место дисциплины в структуре ОПОП
3. Компетенции обучающегося, формируемые в результате освоения дисциплины (модуля)
4. Объем дисциплины (модуля)
5. Структура и содержание дисциплины (модуля)
6. Фонд оценочных средств
 - 6.1. Перечень видов оценочных средств
 - 6.2. Типовые контрольные задания и иные материалы для текущей аттестации
 - 6.3. Типовые контрольные вопросы и задания для промежуточной аттестации
 - 6.4. Критерии оценивания
7. Учебно-методическое и информационное обеспечение дисциплины (модуля)
 - 7.1. Рекомендуемая литература
 - 7.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"
 - 7.3. Перечень информационных технологий
8. Материально-техническое обеспечение дисциплины (модуля)
9. Методические указания для обучающихся по освоению дисциплины (модуля)
10. Специальные условия освоения дисциплины обучающимися с инвалидностью и ограниченными возможностями здоровья



1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Приобретение специализированных знаний в области системного программирования для современных процессоров и операционных систем.

Результаты обучения по дисциплине направлены на достижение индикаторов:

ОПК-7.1 Знает общие принципы построения, области и особенности применения языков программирования высокого и низкого уровня; знает язык программирования высокого и низкого уровня; знает общие сведения о методах проектирования, документирования, разработки, тестирования и отладки программного обеспечения.

ОПК-7.2 Умеет работать с интегрированной средой разработки программного обеспечения; умеет разрабатывать и реализовывать на языке высокого и низкого уровня алгоритмы решения типовых профессиональных задач; умеет применять известные методы программирования и возможности базового языка программирования для решения типовых профессиональных задач.

ОПК-7.3 Владеет навыками разработки алгоритмов решения типовых профессиональных задач; владеет навыками разработки, документирования, тестирования и отладки программ.

ОПК-13.1 Знает общие принципы построения и использования современных языков программирования высокого и низкого уровня; современные технологии программирования; показатели качества программного обеспечения;

ОПК-13.2 Умеет формализовать поставленную задачу, работать с интегрированными средами разработки программного обеспечения; разрабатывать эффективные алгоритмы и программы.

ОПК-13.3 Владеет навыками разработки программных модулей, реализующих задачи, связанные с обеспечением безопасности операционных систем распространенных семейств; навыками разработки алгоритмов для решения типовых профессиональных задач.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Цикл (раздел) ОПОП: Б1.О.14

2.1 Требования к предварительной подготовке обучающегося:

Обучающиеся должны владеть знаниями и навыками в области программирования и операционных систем, получаемыми в рамках следующих дисциплин:

Информатика

Языки программирования

Языки Ассемблера

Операционные системы

2.2 Дисциплины и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:

Данная дисциплина является предшествующей для следующих дисциплин:

Защита программ и данных

Защита в операционных системах

Исследование вредоносного программного обеспечения

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

ОПК-7: Способен создавать программы на языках высокого и низкого уровня, применять методы и инструментальные средства программирования для решения профессиональных задач, осуществлять обоснованный выбор инструментария программирования и способов организации программ;

Знать:

– архитектуру и программный интерфейс современных операционных систем.

Уметь:

– создавать прикладное и системное программное обеспечение для современных операционных систем.

Владеть:



Рабочая программа дисциплины "Системное программирование" по направлению подготовки (специальности) 10.05.01 "Компьютерная безопасность" направленности (профилю) специализация N 6 "Информационно-аналитическая и техническая экспертиза компьютерных систем" ФГБОУ ВО «ЧелГУ»	стр. 4
---	--------

– навыками реализации программного обеспечения любой сложности с использованием высокоуровневых и низкоуровневых языков программирования.

ОПК-13: Способен разрабатывать компоненты программных и программно-аппаратных средств защиты информации в компьютерных системах и проводить анализ их безопасности;

Знать:

– системную и программную архитектуру современных процессоров.

Уметь:

– создавать пользовательские программы;
– создавать код режима ядра.

Владеть:

– навыками использования инструментальных средств создания кода под современные операционные системы.

В результате освоения дисциплины обучающийся должен

3.1 Знать:	
3.1.1	– системную и программную архитектуру современных процессоров и операционных систем;
3.1.2	– программные интерфейсы современных операционных систем;
3.1.3	– методы создания программного кода режима ядра и режима пользователя.
3.2 Уметь:	
3.2.1	– создавать программный код с использованием интерфейсов различных уровней для современных операционных систем.
3.3 Владеть:	
3.3.1	– навыками создания программного обеспечения любой сложности под современные операционные системы.

4. ОБЪЕМ ДИСЦИПЛИНЫ (МОДУЛЯ)

Общая трудоемкость	3 ЗЕТ
Часов по учебному плану : 108 в том числе : аудиторные занятия : 68 самостоятельная работа : 33,1 : контактная работа: 74,9 ИКР: 6,9	Виды контроля в семестрах: зачеты 6

5. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Литература
	Раздел 1. Системное программирование для процессоров архитектур IA-32, x64, ARM, MIPS			
1.1	Создание загрузочного кода. /Лек/	6	4	Л1.1 Л1.2Л2.1 Л2.4 Л2.5Л3.3 Л3.4 Э1 Э2
1.2	Создание загрузочного кода /Лаб/	6	4	Л1.1 Л1.2Л2.1 Л2.4 Л2.5Л3.3 Л3.4 Э1 Э2
1.3	Создание загрузочного кода /Ср/	6	4	Л1.1 Л1.2Л2.1 Л2.4 Л2.5Л3.3 Л3.4 Э1 Э2



Рабочая программа дисциплины "Системное программирование" по направлению подготовки (специальности) 10.05.01 "Компьютерная безопасность" направленности (профилю) специализация N 6 "Информационно-аналитическая и техническая экспертиза компьютерных систем" ФГБОУ ВО «ЧелГУ»				стр. 5
1.4	Реализация функциональности операционной системы. /Лек/	6	4	Л1.1 Л1.2Л2.1 Л2.4 Л2.5Л3.2 Л3.3 Л3.4 Э1 Э2
1.5	Реализация функциональности операционной системы. /Лаб/	6	4	Л1.1 Л1.2Л2.1 Л2.4 Л2.5Л3.2 Л3.3 Л3.4 Э1 Э2
1.6	Реализация функциональности операционной системы. /Ср/	6	4	Л1.1 Л1.2Л2.1 Л2.4 Л2.5Л3.2 Л3.3 Л3.4 Э1 Э2
Раздел 2. Системное программирование в Windows NT				
2.1	Программная среда операционной системы. /Лек/	6	4	Л1.1Л2.1 Л2.2 Л2.3 Л2.4 Л2.5Л3.1 Л3.2 Э1 Э2
2.2	Программная среда операционной системы. /Лаб/	6	4	Л1.1Л2.1 Л2.2 Л2.3 Л2.4 Л2.5Л3.1 Л3.2 Э1 Э2
2.3	Программная среда операционной системы. /Ср/	6	6,1	Л1.1Л2.1 Л2.2 Л2.3 Л2.4 Л2.5Л3.1 Л3.2 Э1 Э2
2.4	Архитектура ядра операционной системы. Создание кода режима ядра. /Лек/	6	4	Л1.1Л2.1 Л2.2 Л2.3 Л2.4 Л2.5Л3.1 Л3.2 Э1 Э2
2.5	Архитектура ядра операционной системы. Создание кода режима ядра. /Лаб/	6	4	Л1.1Л2.1 Л2.2 Л2.3 Л2.4 Л2.5Л3.1 Л3.2 Э1 Э2
2.6	Архитектура ядра операционной системы. Создание кода режима ядра. /Ср/	6	4	Л1.1Л2.1 Л2.2 Л2.3 Л2.4 Л2.5Л3.1 Л3.2 Э1 Э2
2.7	Руткиты в операционных системах Windows NT. /Лек/	6	4	Л1.1Л2.1 Л2.2 Л2.3 Л2.4 Л2.5Л3.1 Л3.2 Э1 Э2
2.8	Руткиты в операционных системах Windows NT. /Лаб/	6	4	Л1.1Л2.1 Л2.2 Л2.3 Л2.4 Л2.5Л3.1 Л3.2 Э1 Э2
2.9	Руткиты в операционных системах Windows NT. /Ср/	6	4	Л1.1Л2.1 Л2.2 Л2.3 Л2.4 Л2.5Л3.1 Л3.2 Э1 Э2
Раздел 3. Системное программирование в UNIX				
3.1	Программная среда операционной системы. /Лек/	6	6	Л1.1 Л1.2Л2.1 Л2.4 Л2.5Л3.2 Л3.4 Э1



Рабочая программа дисциплины "Системное программирование" по направлению подготовки (специальности) 10.05.01 "Компьютерная безопасность" направленности (профилю) специализация N 6 "Информационно-аналитическая и техническая экспертиза компьютерных систем" ФГБОУ ВО «ЧелГУ»				стр. 6
3.2	Программная среда операционной системы. /Лаб/	6	6	Л1.1 Л1.2Л2.1 Л2.4 Л2.5Л3.2 Л3.4 Э1 Э2
3.3	Программная среда операционной системы. /Ср/	6	4	Л1.1 Л1.2Л2.1 Л2.4 Л2.5Л3.2 Л3.4 Э1 Э2
3.4	Архитектура ядра операционной системы. Создание кода режима ядра. /Лек/	6	4	Л1.1 Л1.2Л2.1 Л2.4 Л2.5Л3.2 Л3.4 Э1 Э2
3.5	Архитектура ядра операционной системы. Создание кода режима ядра. /Лаб/	6	4	Л1.1 Л1.2Л2.1 Л2.4 Л2.5Л3.2 Л3.4 Э1 Э2
3.6	Архитектура ядра операционной системы. Создание кода режима ядра. /Ср/	6	4	Л1.1 Л1.2Л2.1 Л2.4 Л2.5Л3.2 Л3.4 Э1 Э2
3.7	Руткиты в операционных системах UNIX. /Лек/	6	4	Л1.1 Л1.2Л2.1 Л2.4 Л2.5Л3.2 Л3.4 Э1 Э2
3.8	Руткиты в операционных системах UNIX. /Лаб/	6	4	Л1.1 Л1.2Л2.1 Л2.4 Л2.5Л3.2 Л3.4 Э1 Э2
3.9	Руткиты в операционных системах UNIX. /Ср/	6	3	Л1.1 Л1.2Л2.1 Л2.4 Л2.5Л3.2 Л3.4 Э1 Э2
Раздел 4. Иная контактная работа				
4.1	Иная контактная работа: индивидуальные консультации, текущий контроль. /ИКР/	6	6,9	

6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

6.1. Перечень видов оценочных средств

Зачетные задания.
Домашние задания.
Аудиторные задания.
Вопросы на зачёте.
Задания на зачёте.
Вопросы на экзамене.
Задания на экзамене.

6.2. Типовые контрольные задания и иные материалы для текущей аттестации

Темы для домашних, аудиторных и зачетных заданий:
1. Системное программирование для процессоров архитектур IA-32, x64.
2. Системное программирование в Windows.
3. Системное программирование в Linux.
4. Руткиты.

Примеры домашних, аудиторных и зачетных заданий:
1. Написать обработчик прерывания.
2. Реализовать загрузочный код MBR.
3. Обработать исключения защищённого режима.



4. Обработать в загрузчике виртуальную память.
5. Реализовать элементы функциональности операционной системы (управление виртуальной памятью, обработка прерываний, переключение процессов и др.).
6. Реализовать программу под Windows с использованием WinAPI.
7. Реализовать функции разбора PE-файлов.
8. Реализовать загрузку PE-файлов.
9. Реализовать экспорт из исполняемого PE-файла.
10. Реализовать механизм плагинов с помощью динамических библиотек.
11. Реализовать двусвязный список в драйвере под Windows NT.
12. Написать драйвер под Windows NT, обрабатывающий запросы на чтение и запись к своему устройству.
13. Реализовать в драйвере под Windows NT работу с аппаратными структурами процессора.
14. Написать драйверы под Windows NT, перехватывающие системные вызовы.
15. Написать драйвер под Windows NT, выводющий список процессов.
16. Написать драйвер под Windows NT, скрывающий процесс.
17. Написать драйверы под Windows NT, использующий нотификаторы ядра.
18. Реализовать в драйвере под Windows NT перехват с помощью сплайсинга.
19. Реализовать кейлоггер режима ядра под Windows NT.
20. Реализовать в драйвере под Windows NT скрытие портов.
21. Реализовать в драйвере под Windows NT перехват и модификацию трафика.
22. Реализовать в драйвере под Windows NT повышение привилегий процессов.
23. Реализовать в драйвере под Windows NT синхронизацию с помощью примитивов синхронизации.
24. Реализовать утилиты под Linux с использованием системных вызовов.
25. Модифицировать системные вызовы в ядре Linux.
26. Реализовать модули ядра под Linux.
27. Реализовать перехват и модификацию сетевого трафика в модуле ядра под Linux.
28. Реализовать руткит.

6.3. Типовые контрольные вопросы и задания для промежуточной аттестации

Темы заданий на зачёте

1. Системное программирование для процессоров архитектур IA-32, x64.
2. Системное программирование в Windows.
3. Системное программирование в Linux.

Примеры заданий на зачёте

1. Написать обработчик прерывания.
2. Реализовать загрузочный код MBR.
3. Обработать исключения защищённого режима.
4. Обработать в загрузчике виртуальную память.
5. Реализовать элементы функциональности операционной системы (управление виртуальной памятью, обработка прерываний, переключение процессов и др.).
6. Реализовать программу под Windows с использованием WinAPI.
7. Реализовать функции разбора PE-файлов.
8. Реализовать загрузку PE-файлов.
9. Реализовать экспорт из исполняемого PE-файла.
10. Реализовать механизм плагинов с помощью динамических библиотек.
11. Реализовать двусвязный список в драйвере под Windows NT.
12. Написать драйвер под Windows NT, обрабатывающий запросы на чтение и запись к своему устройству.
13. Реализовать в драйвере под Windows NT работу с аппаратными структурами процессора.
14. Написать драйверы под Windows NT, перехватывающие системные вызовы.
15. Написать драйвер под Windows NT, выводющий список процессов.
16. Написать драйвер под Windows NT, скрывающий процесс.
17. Написать драйвер под Windows NT, использующий нотификаторы ядра.
18. Реализовать в драйвере под Windows NT перехват с помощью сплайсинга.
19. Реализовать кейлоггер режима ядра под Windows NT.
20. Реализовать в драйвере под Windows NT скрытие портов.
21. Реализовать в драйвере под Windows NT перехват и модификацию трафика.
22. Реализовать в драйвере под Windows NT повышение привилегий процессов.
23. Реализовать в драйвере под Windows NT синхронизацию с помощью примитивов синхронизации.
24. Реализовать утилиты под Linux с использованием системных вызовов.
25. Реализовать модули ядра под Linux.



Примеры вопросов на зачёте:

1. Системная архитектуры процессоров IA-32, x64.
2. Программные интерфейсы Windows NT.
3. Интерфейсы ядра Windows NT.
4. Драйверы режима ядра для Windows NT.
5. Формат исполняемых файлов PE.
6. Программные интерфейсы Linux.
7. Интерфейсы ядра Linux.
8. Формат исполняемых файлов ELF.

Темы заданий на экзамене:

1. Системное программирование для процессоров архитектур IA-32, x64.
2. Системное программирование в Windows.
3. Системное программирование в Linux.
4. Руткиты.

Примеры заданий на экзамене:

1. Написать обработчик прерывания.
2. Реализовать загрузочный код MBR.
3. Обработать исключения защищённого режима.
4. Обработать в загрузчике виртуальную память.
5. Реализовать элементы функциональности операционной системы (управление виртуальной памятью, обработка прерываний, переключение процессов и др.).
6. Реализовать программу под Windows с использованием WinAPI.
7. Реализовать функции разбора PE-файлов.
8. Реализовать загрузку PE-файлов.
9. Реализовать экспорт из исполняемого PE-файла.
10. Реализовать механизм плагинов с помощью динамических библиотек.
11. Реализовать двусвязный список в драйвере под Windows NT.
12. Написать драйвер под Windows NT, обрабатывающий запросы на чтение и запись к своему устройству.
13. Реализовать в драйвере под Windows NT работу с аппаратными структурами процессора.
14. Написать драйверы под Windows NT, перехватывающие системные вызовы.
15. Написать драйвер под Windows NT, выводящий список процессов.
16. Написать драйвер под Windows NT, скрывающий процесс.
17. Написать драйвер под Windows NT, использующий нотификаторы ядра.
18. Реализовать в драйвере под Windows NT перехват с помощью сплайсинга.
19. Реализовать кейлоггер режима ядра под Windows NT.
20. Реализовать в драйвере под Windows NT скрытие портов.
21. Реализовать в драйвере под Windows NT перехват и модификацию трафика.
22. Реализовать в драйвере под Windows NT повышение привилегий процессов.
23. Реализовать в драйвере под Windows NT синхронизацию с помощью примитивов синхронизации.
24. Реализовать утилиты под Linux с использованием системных вызовов.
25. Модифицировать системные вызовы в ядре Linux.
26. Реализовать модули ядра под Linux.
27. Реализовать перехват и модификацию сетевого трафика в модуле ядра под Linux.
28. Реализовать руткит.

Примеры вопросов на экзамене:

1. Системная архитектуры процессоров IA-32, x64.
2. Программные интерфейсы Windows NT.
3. Интерфейсы ядра Windows NT.
4. Драйверы режима ядра для Windows NT.
5. Формат исполняемых файлов PE.
6. Руткиты режима ядра в Windows NT.
7. Программные интерфейсы Linux.
8. Интерфейсы ядра Linux.
9. Формат исполняемых файлов ELF.
10. Руткиты режима ядра в Linux.

6.4. Критерии оценивания



За своевременное и самостоятельное выполнение учебных работ в течение семестра студент получает рейтинговые баллы. Сумма за выполнение основных заданий в полном объёме - 100. Сверх этой суммы могут начисляться баллы за выполнение дополнительных заданий.

Пропуск по не уважительной причине одной пары влечет вычет 1 балла из итоговой суммы за семестр.

При нехватке баллов преподавателем может быть предоставлено дополнительное задание или возможность доделать задание, в котором была оценена не вся функциональность.

Основные баллы выставляются за выполнение объемных зачётных заданий, которые выполняются дома и сдаются в течение семестра. Сумма в 100 баллов делится между зачётными заданиями (не обязательно равномерно). При выдаче зачётного задания определено сколько баллов выставляется за реализацию определённой функциональности. Для зачётных заданий может быть определена функциональность повышенной сложности, за выполнение которой выставляются дополнительные баллы. Также дополнительные баллы могут быть выставлены по усмотрению преподавателя за особо примечательную реализацию.

При сдаче зачётного задания производится опрос по техническим деталям реализации и по теории, используемой при выполнении заданий. Неудовлетворительный ответ будет означать несамостоятельность выполнения задания, что влечёт выставление 0 баллов за соответствующую функциональность. Если для одного задания это повторяется более 2 раз, то за всё задание выставляется 0 баллов без возможности повторной сдачи.

Выполнение заданий предполагает некоторую программную реализацию, к которой будут предъявляться обычные требования по качеству кода. Код должен быть удобочитаемым, хорошо структурированным, написанным в едином стиле. Иначе возможна сценка до 5 баллов. За программные ошибки, приводящие к работоспособности кода не для всех возможных случаев, возможна сценка до 10 баллов (в зависимости от критичности ошибки). За программные ошибки, приводящие к аварийному некорректному завершению программы, возможна сценка до 10 баллов (в зависимости от критичности ошибки). По пройденному материалу выдаются небольшие задания для выполнения дома и/или во время семинарских занятий. За эти задания выставляются небольшие дополнительные баллы. Сдавать их можно либо в день выдачи либо на следующем занятии. Домашние и аудиторские задания – это небольшие задания, за которые обычно выставляется 1-2 балла. Они оцениваются атомарно: либо задание выполнено (выставляется указанное при выдаче задания количество баллов), либо не выполнено (0 баллов).

Итоговая оценка за дисциплину выставляется по результатам выполнения заданий текущего контроля. При необходимости во время зачёта и экзамена может быть предоставлена возможность получить дополнительные баллы (не более 20), выполнив дополнительные задания и ответив (в устной форме) на вопросы.

Дополнительные задания, выдаваемые на зачёте и экзамене, являются относительно объёмными, за них выставляется до 10-15 баллов. Поэтому к ним применимы описанные выше критерии оценивания зачётных заданий с соответствующей корректировкой баллов: сценка за некорректную работу до 5 баллов, за аварийное завершение – до 5 баллов.

Ответ на зачёте и экзамене оценивается по трём параметрам.

1. Построение ответа (структура ответа, грамотность речи, последовательность и т.д.).

Студент самостоятельно правильно выстраивает структуру ответа, изложение последовательное, речь грамотная без оговорок. 2 балла

Изложение студента не последовательное и обрывочное, взаимосвязи частей ответа не всегда прослеживаются. Раскрытие сути ответа невозможно без уточняющих вопросов. 1 балл.

Студент испытывает существенные трудности при самостоятельном построении ответа, способен только давать краткие ответы на конкретные вопросы. 0 баллов.

2. Фактическая полнота ответа.

Студент правильно ответил на теоретический вопрос билета. Показал отличные знания в рамках усвоенного учебного материала. Ответил на все дополнительные вопросы. 4 балла.

Студент ответил на теоретический вопрос билета с небольшими неточностями. Показал хорошие знания в рамках усвоенного учебного материала. Ответил на большинство дополнительных вопросов. 3 балла.

Студент ответил на теоретический вопрос билета с существенными неточностями. Показал удовлетворительные знания в рамках усвоенного учебного материала. При ответах на дополнительные вопросы было допущено много неточностей. 2 балла.

При ответе на теоретический вопрос билета студент продемонстрировал недостаточный уровень знаний. При ответах на дополнительные вопросы было допущено множество неправильных ответов. 1 балл.

Студент не ответил на вопрос. 0 баллов.

3. Собственный анализ излагаемого материала его оценка в контексте взаимодействия с другими областями, умение применять на практике.

Студент ясно осознаёт место излагаемого материала в общей структуре профессионального знания, знает стандартные примеры использования и предлагает свои, даёт собственные компетентные оценки. 4 балла.



Рабочая программа дисциплины "Системное программирование" по направлению подготовки (специальности)
10.05.01 "Компьютерная безопасность" направленности (профилю) специализация N 6
"Информационно-аналитическая и техническая экспертиза компьютерных систем" ФГБОУ ВО «ЧелГУ»

стр. 10

Студент осознаёт взаимосвязи и знает стандартные примеры использования. Допускает неточности при самостоятельном анализе. 3 балла.

Студент в общих чертах осознаёт взаимосвязи и знает стандартные примеры использования. При проведении самостоятельного анализа нуждается в уточняющих вопросах, при этом допускает существенные неточности. 2 балла.

Студент знает основные стандартные примеры использования. Не осознаёт взаимосвязей с другими областями. 1 балл.

Студент не осознаёт взаимосвязей и практическое приложение излагаемого материала. 0 баллов.

Перевод рейтинговых баллов в оценки за зачет:

61 и более баллов – зачтено;

60 и менее баллов – не зачтено.

Перевод рейтинговых баллов в оценки за экзамен:

91 и более баллов – отлично;

76 - 90 баллов – хорошо;

61 - 75 баллов – удовлетворительно;

60 и менее баллов – неудовлетворительно.

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

7.1. Рекомендуемая литература

7.1.1. Основная литература

	Авторы, составители	Заглавие	Издательство, год	Ресурс
Л1.1	Зубков С. В.	Assembler. Для DOS, Windows и Unix (https://e.lanbook.com/books/element.php?pl1_cid=25&pl1_id=1243)	Москва : ДМК Пресс, 2008	ЭБС
Л1.2	Кузнецов А.С., Якимов И.А., Пересунько П.В.	Системное программирование: учебное пособие (https://znanium.com/catalog/document?id=342172)	Красноярск : Сибирский федеральный университет, 2018	ЭБС

7.1.2. Дополнительная литература

	Авторы, составители	Заглавие	Издательство, год	Ресурс
Л2.1	Рудаков П. И., Финогенов К. Г.	Язык ассемблера: уроки программирования: практическое пособие (https://biblioclub.ru/index.php?page=book&id=89393)	Москва : Диалог- МИФИ, 2001	ЭБС
Л2.2	Коньков К. А.	Устройство и функционирование ОС Windows: практикум к курсу «Операционные системы»: учебное пособие (https://biblioclub.ru/index.php?page=book&id=233308)	Москва : Интернет- Университет Информационны х Технологий (ИНТУИТ) Бином. Лаборатория знаний, 2008	ЭБС
Л2.3	Котельников Е.	Введение во внутреннее устройство Windows: курс лекций (https://biblioclub.ru/index.php?page=book&id=429084)	Москва : Национальный Открытый Университет «ИНТУИТ», 2016	ЭБС
Л2.4	Пильщиков В. Н.	Программирование на языке ассемблера IBM PC: учебное пособие (https://biblioclub.ru/index.php?page=book&id=447687)	Москва : Диалог- МИФИ, 2014	ЭБС



Рабочая программа дисциплины "Системное программирование" по направлению подготовки (специальности) 10.05.01 "Компьютерная безопасность" направленности (профилю) специализация N 6 "Информационно-аналитическая и техническая экспертиза компьютерных систем" ФГБОУ ВО «ЧелГУ»			стр. 11	
	Авторы, составители	Заглавие	Издательство, год	Ресурс
Л2.5	Аблязов Р. З.	Программирование на ассемблере на платформе x86-64 (http://e.lanbook.com/books/element.php?pl1_cid=25&pl1_id=1273)	Москва : ДМК Пресс, 2011	ЭБС
7.1.3. Методические разработки				
	Авторы, составители	Заглавие	Издательство, год	Ресурс
Л3.1	Безруков В. А.	WIN32 API программирование (http://e.lanbook.com/books/element.php?pl1_id=40770)	Санкт- Петербург : НИУ ИТМО, 2009	ЭБС
Л3.2	Костюкова Н. И.	Программирование на языке Си: методические рекомендации и задачи по программированию: методическое пособие (https://biblioclub.ru/index.php?page=book&id=57176)	Новосибирск : Сибирское университетское издательство, 2003	ЭБС
Л3.3	Маркова В. П., Киреев С. Е., Остапкевич М. Б., Перепелкин В. А.	Эффективное программирование современных микропроцессоров: учебное пособие (https://biblioclub.ru/index.php?page=book&id=435972)	Новосибирск : Новосибирский государственный технический университет, 2014	ЭБС
Л3.4	Абрамов Е. С., Сидоров И. Д.	Машинно-ориентированное программирование: учебное пособие (https://biblioclub.ru/index.php?page=book&id=492941)	Таганрог : Южный федеральный университет, 2016	ЭБС
7.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"				
Э1	Лань [Электронный ресурс] : электронно-библиотечная система (ЭБС) / издательство Лань. – URL: http://e.lanbook.com/ .			
Э2	Университетская библиотека онлайн [Электронный ресурс] : электронно-библиотечная система (ЭБС) / ООО Директмедиа Паблишинг. – URL: http://biblioclub.ru/ .			
7.3 Перечень информационных технологий				
7.3.1 Программное обеспечение				
Visual Studio				
VirtualBox				
Adobe Reader				
Notepad++				
7.3.2 Профессиональные базы данных и информационно-справочные системы				
1. Электронный каталог научной библиотеки ЧелГУ [Электронный ресурс] : база данных / Челяб. гос. ун-т. – Челябинск, 1992 – .				
2. MSDN Library [Электронный ресурс]. URL: https://msdn.microsoft.com/ .				

8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Для реализации дисциплины используются учебные аудитории для проведения занятий лекционного типа, занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы.

Учебные аудитории укомплектованы специализированной мебелью и техническими средствами обучения: проектором, экраном, магнитно-маркерной доской, маркером; с возможностью демонстрации электронных презентаций при уровне освещения, достаточном для работы с конспектом, персональными компьютерами.

Для проведения занятий лекционного типа имеется демонстрационное оборудование: проектор, экран.



Лабораторные занятия проходят в учебных лабораториях технических средств защиты информации и "Сетевой полигон" (ауд. 421, 423, учебный корпус №1). Материально-техническое обеспечение приведено в паспортах лабораторий.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с подключением к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.

9. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

Для изучения дисциплины предусмотрены проведение лекционных и лабораторных занятий, а также самостоятельная работа студентов.

Лекционные занятия обеспечивают теоретическое изучение дисциплины. Основными методами обучения являются информационно-объяснительный и проблемный. На лекциях излагается основное содержание тем программы, проводится анализ основных понятий и методов, разбираются примеры программного кода, демонстрируются методы системного программирования, обсуждаются возможные приложения изложенных методов.

Лабораторные занятия служат для закрепления теоретических основ, излагаемых в лекциях, и получения практического опыта системного программирования. Основные методы: репродуктивный и частично-поисковый. Для проведения текущего промежуточного контроля на каждом лабораторном занятии выдаются небольшие практические задания, рассчитанные на выполнение в течение занятия, и домашние задания.

Для самостоятельной работы студентам следует использовать методические материалы, имеющиеся в Научной библиотеке ЧелГУ, а также выложенные на сайте математического факультета ЧелГУ и на сайте кафедры компьютерной безопасности и прикладной алгебры. Для студентов проводятся индивидуальные консультации, каждому студенту при необходимости могут быть выданы индивидуальные задания для самостоятельной работы, позволяющие углубленно изучить отдельные темы дисциплины.

Стоит особо подчеркнуть, что курс носит практический характер. Основным результатом освоения курса является умение применять изученные методы на практике. Поэтому для успешного освоения курса обязательно полное выполнение всех практических заданий.

В случае применения при обучении дисциплины электронного обучения, дистанционных образовательных технологий общение обучающихся и преподавателя осуществляется в режиме реального времени (онлайн-лекции (вебинары), чаты, видео-конференции и др.) или отложенного времени (система дистанционного обучения Moodle, форумы, электронная почта и др.).

Большую часть времени обучающиеся самостоятельно работают с учебно-методическими материалами. Студенты имеют возможность консультироваться с преподавателем по всем вопросам, возникающим в ходе самостоятельной работы посредством электронной почты, мессенджеров, социальных сетей и т.п.

Доступ обучающегося к учебным ресурсам в режиме отложенного времени, самостоятельной работы осуществляется через сеть Интернет в удобном для него месте, времени и темпе.

При обучении лиц с ограниченными возможностями здоровья электронное обучение, дистанционные образовательные технологии предусматривают возможность приема-передачи информации в доступных для них формах.

Реализация дисциплины с применением электронного обучения, дистанционных образовательных технологий (далее – ЭО, ДОТ) осуществляется на основании «Положения о реализации основных и дополнительных образовательных программ с применением электронного обучения и дистанционных образовательных технологий в федеральном государственном бюджетном образовательном учреждении высшего образования «Челябинский государственный университет», «Положения о порядке зачета обучающимися по основным профессиональным образовательным программам высшего образования в ФГБОУ ВО «ЧелГУ» результатов освоения в организациях, осуществляющих образовательную деятельность, учебных предметов, курсов, дисциплин (модулей), практик, дополнительных образовательных программ» посредством электронной информационно-образовательной среды ФГБОУ ВО «ЧелГУ». В исключительных случаях (форс-мажор и т.п.) при реализации образовательной деятельности с применением ЭО, ДОТ могут применяться компоненты, не входящие в перечень электронной информационно-образовательной среды.

10. СПЕЦИАЛЬНЫЕ УСЛОВИЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ ОБУЧАЮЩИМИСЯ С ИНВАЛИДНОСТЬЮ И ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

Освоение дисциплины инвалидами и лицами с ограниченными возможностями здоровья осуществляется с использованием специальных технических средств и информационных технологий, предоставляемых Ресурсным учебно-методическим центром по обучению инвалидов и лиц с ограниченными возможностями здоровья ЧелГУ по запросу обучающегося (мобильные специальные технические средства для лиц с нарушениями зрения и с нарушением слуха, ассистивные информационные технологии).



При необходимости для обучающихся с нарушениями зрения на рабочих местах для проведения практических или лабораторных занятий устанавливается специальное программное обеспечение (программа речевой навигации, речевые синтезаторы, экранные лупы).

В учебные аудитории обеспечивается беспрепятственный доступ для обучающихся с инвалидностью и с ограниченными возможностями здоровья. В каждой аудитории, где обучаются инвалиды и лица с ограниченными возможностями здоровья, предусматривается соответствующее количество мест для обучающихся с учетом нарушений их здоровья.

Для освоения дисциплины инвалидам и лицам с ограниченными возможностями здоровья предоставляется доступ к печатным источникам, имеющимся в научной библиотеке ЧелГУ, с помощью специальных технических средств; доступ с помощью специальных технических и программных средств к электронным источникам, представленным в форме электронного документа в фонде научной библиотеки ЧелГУ или электронно-библиотечных системах.

Учебно-методические материалы для обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья предоставляются в формах, адаптированных к ограничениям их здоровья и особенностям восприятия информации.

Для инвалидов и лиц с ограниченными возможностями здоровья освоение дисциплины может быть частично или полностью осуществлено с использованием дистанционных образовательных технологий.

При проведении промежуточной аттестации по дисциплине обучающимся с инвалидностью и с ограниченными возможностями здоровья обеспечивается по их заявлению предоставление в доступной форме в зависимости от их индивидуальных особенностей инструкции о порядке проведения промежуточной аттестации, оценочных средств и возможности ответов на задания (письменно на бумаге, набор ответов на компьютере, письменно шрифтом Брайля, с использованием услуг ассистента, устно).

При проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование предоставленных ЧелГУ или собственных технических средств, необходимых им в связи с их индивидуальными особенностями. При необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на задания, процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

