

**ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ
ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)**

1. Общие сведения

1.	Кафедра	Математики, физики и информационных технологий
2.	Направление подготовки	01.03.02 Прикладная математика и информатика
	Направленность (профиль)	Системное программирование и компьютерные технологии
3.	Дисциплина (модуль)	Б1.В.01.01 Операционные системы
4.	Форма обучения	очная
5.	Год набора	2023

2. Перечень компетенций

– ПК-3 Способен к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения

3. Критерии и показатели оценивания компетенций на различных этапах их формирования

Этап формирования компетенции (разделы, темы дисциплины)	Формируемая компетенция	Критерии и показатели оценивания компетенций:			Формы контроля сформированности компетенций
		Знать:	Уметь:	Владеть:	
Введение в операционные системы	ПК-3	<ul style="list-style-type: none"> – основные функции и назначение ОС; . – способы построения ОС; – актуальные виды операционных систем; 	<ul style="list-style-type: none"> – использовать функционал операционной системы;в. – анализировать возможности операционных систем; – выбирать оптимальную ОС для решения задач; – использовать стандартные приложения ОС; – использовать средства ОС для работы в сети 	<ul style="list-style-type: none"> – информацией о внутреннем устройстве операционных систем и их функциональных возможностях;. – навыками выбора операционной системы для решения задач; – навыками работы с программным обеспечением операционных систем 	<ul style="list-style-type: none"> Решение тестов Подготовка презентаций Подготовка докладов Работа на практических/лабораторных занятиях Контрольные вопросы Собеседование <p>(по выбору преподавателя)</p>
Работа в среде UNIX-подобной операционной системы	ПК-3	<ul style="list-style-type: none"> – стандартное программное обеспечение ОС. 			
Настройка и обслуживание операционной системы	ПК-3				

Шкала оценивания в рамках балльно-рейтинговой системы:

«неудовлетворительно» – 60 баллов и менее; «удовлетворительно» – 61-80 баллов; «хорошо» – 81-90 баллов; «отлично» – 91-100 баллов

4. Критерии и шкалы оценивания

*В приведенных ниже таблицах указан первичный балл (ПБ). Алгоритм вычисления итогового балла за работу (ИБР) приведен в конце данного раздела.

4.1. Активность на теоретических занятиях

Характеристики ответа студента	Максимальное количество баллов*
Студент принимает активное участие в беседе на лекции	1
Студент не принимает активное участие в беседе на лекции или отсутствует	0

4.2. Работа на практических/лабораторных занятиях

Характеристики ответа студента	Максимальное количество баллов*
Лабораторная работа выполнена не менее чем на 91%	0,9 — 1
Лабораторная работа выполнена не менее чем на 81%	0,81 — 0,9
Лабораторная работа выполнена не менее чем на 61%	0,61 — 0,80
Лабораторная работа выполнена менее чем на 60%	0

4.3. Подготовка доклада

Характеристики ответа студента	Максимальное количество баллов*
<ul style="list-style-type: none">студент глубоко и всесторонне усвоил проблему;уверенно, логично, последовательно и грамотно его излагает;опираясь на знания основной и дополнительной литературы, тесно привязывает усвоенные научные положения с практической деятельностью;умело обосновывает и аргументирует выдвигаемые им идеи;делает выводы и обобщения;свободно владеет понятиями.	0,91 — 1
<ul style="list-style-type: none">студент твердо усвоил тему, грамотно и по существу излагает ее, опираясь на знания основной литературы;не допускает существенных неточностей;увязывает усвоенные знания с практической деятельностью;аргументирует научные положения;делает выводы и обобщения;владеет системой основных понятий.	0,81 — 0,90
<ul style="list-style-type: none">тема раскрыта недостаточно четко и полно, то есть студент усвоил проблему, по существу излагает ее, опираясь на знания только основной литературы;допускает несущественные ошибки и неточности;испытывает затруднения в практическом применении знаний;слабо аргументирует научные положения;затрудняется в формулировании выводов и обобщений;частично владеет системой понятий.	0,61 — 0,80
<ul style="list-style-type: none">студент не усвоил значительной части проблемы;допускает существенные ошибки и неточности при рассмотрении ее;испытывает трудности в практическом применении знаний;не может аргументировать научные положения;не формулирует выводов и обобщений;не владеет понятийным аппаратом.	0

4.4. Подготовка презентаций

Структура презентации	Максимальное количество баллов*
Содержание:	
<ul style="list-style-type: none">Сформулирована цель работы	0,1
<ul style="list-style-type: none">Понятны задачи и ход работы	0,1
<ul style="list-style-type: none">Информация изложена полно и четко	0,1
<ul style="list-style-type: none">Иллюстрации усиливают эффект восприятия текстовой части информации	0,1
<ul style="list-style-type: none">Сделаны выводы	0,1

Оформление презентации	
<ul style="list-style-type: none"> • Единый стиль оформления 	0,1
<ul style="list-style-type: none"> • Текст легко читается, фон сочетается с текстом и графикой 	0,1
<ul style="list-style-type: none"> • Все параметры шрифта хорошо подобраны, размер шрифта оптимальный и одинаковый на всех слайдах 	0,1
<ul style="list-style-type: none"> • Ключевые слова в тексте выделены 	0,1
Эффект презентации	
<ul style="list-style-type: none"> • Общее впечатление от просмотра презентации 	0,1
Всего	1

4.5. Контрольные вопросы

Характеристики ответа студента	Максимальное количество баллов*
Отвечено без замечаний не менее чем на 91% вопросов	0,9 — 1
Отвечено без замечаний не менее чем на 81% вопросов	0,81 — 0,9
Отвечено без замечаний не менее чем на 61% вопросов	0,61 — 0,80
Отвечено без замечаний менее чем на 60% вопросов	0

4.6. Решение тестовых заданий

Характеристики ответа студента	Максимальное количество баллов*
Тест решен правильно не менее чем на 91%	0,9 — 1
Тест решен правильно не менее чем на 81%	0,81 — 0,9
Тест решен правильно не менее чем на 61%	0,61 — 0,80
Тест решен правильно менее чем на 60%	0

4.7. Собеседование

Характеристики ответа студента	Максимальное количество баллов*
Отвечено без замечаний не менее чем на 91% вопросов	0,9 — 1
Отвечено без замечаний не менее чем на 81% вопросов	0,81 — 0,9
Отвечено без замечаний не менее чем на 61% вопросов	0,61 — 0,80
Отвечено без замечаний менее чем на 60% вопросов	0

4.8. Курсовая работа

Оценка курсовой работы включает в себя: содержание курсовой работы; оформление курсовой работы; процедуру защиты.

Характеристики ответа студента	Максимальное количество баллов*	Оценка
Компетенции сформированы в полном объеме: - работа полностью соответствует установленным требованиям, выполнена и представлена в надлежащие сроки и оформлена в соответствии с действующими нормативами; - работа основана на исследовании значительного массива источников и научной литературы; - в ходе защиты работы студент демонстрирует глубокое знание предмета исследования, понимание его места в системе наук, общую эрудицию, сформированные навыки публичной речи и ведения научной дискуссии.	0,9 — 1	отлично
Компетенции в основном сформированы: - работа соответствует установленным требованиям, выполнена и представлена в надлежащие сроки и оформлена в соответствии с действующими нормативами; - содержание работы отличается актуальностью; - работа основана на исследовании большого числа источников и научной литературы; - выводы, сформулированные в работе, соответствуют современному уровню научного знания;	0,81 — 0,9	хорошо

- в ходе защиты курсовой студент демонстрирует знание предмета исследования, общую эрудицию, общие навыки публичной речи.		
Компетенции сформированы частично: - работа, в основном, соответствует установленным требованиям, выполнена и представлена на кафедру в надлежащие сроки и оформлена в соответствии с действующими нормативам; - работа основана на недостаточном для исследования данной темы объеме источников и научной литературы; - выводы, сформулированные в работе, носят вторичный характер; - в ходе защиты студент демонстрирует минимальные навыки владения методами публичного выступления и научной дискуссии.	0,61 — 0,80	удовлетворительно
Компетенции не сформированы: - работа не соответствует установленным требованиям, выполнена и представлена с нарушением действующих нормативов времени и оформления текста; - содержание работы имеет явные признаки компиляции, изложение материала имеет описательный (реферативный) характер; - объем исследованных источников и научной литературы незначительный; - в ходе защиты студент демонстрирует отсутствие навыков публичной речи и научной дискуссии.	0	Не удовлетворительно

Краткое описание системы оценивания: при оценивании работ используются следующие понятия:

- *первичный балл (ПБ) - выставляется преподавателем в соответствии с приведенными выше таблицами и может принимать значения от 0 до 1;*
- *максимальный бал за работу по технологической карте (МБТК) берется из технологической карты;*
- *итоговый балл за работу (ИБР) — вычисляется по формуле*
$$ИБР = МБТК * ПБ$$

Например, студент за выполненную им лабораторную работу получает первичный балл равный 0.81. В технологической карте указано, что за полностью правильно выполненную лабораторную работу студент может получить 3 итоговых балла, т. е.

$$ИБР = 3 * 0,81 = 2,43$$

5. Типовые контрольные задания и методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

5.1. Типовое тестовое задание

1. Совокупность программных средств, осуществляющая управление ресурсами компьютера, запуск прикладных программ и их взаимодействие с внешними устройствами и другими программами
 - a). драйвер
 - b). системная операция
 - c). программа
 - d). операционная система
 - e). программное обеспечение
2. Утилита для сброса пароля в операционной системе Windows
 - a) chntpw
 - b) resetpwd
 - c) passwordreset
 - d) changepwd
 - e) pwdrestore
3. Команда смены текущей директории
 - a) ps
 - b) ls
 - c) cd
 - d) rmdir
 - e) rm

4. Опция команды ls для отображения всего содержимого (кроме ссылки на текущий каталог . и ссылки на родительский каталог ..)
 - a) -h
 - b) -f
 - c) -A
 - d) -g
 - e) -r
5. Крайне упрощённое и минимальное ядро, выполняет лишь одну задачу
 - a) Наноядро
 - b) Микроядро
 - c) Экзоядро
 - d) Монолитное ядро
 - e) Модульное ядро
6. Ядро операционной системы предоставляет только элементарные функции управления процессами и минимальный набор абстракций для работы с оборудованием
 - a) Наноядро
 - b) Микроядро
 - c) Экзоядро
 - d) Монолитное ядро
 - e) Модульное ядро
7. Ядро операционной системы предоставляет лишь набор сервисов для взаимодействия между приложениями, а также необходимый минимум функций, связанных с защитой
 - a) Наноядро
 - b) Микроядро
 - c) Экзоядро
 - d) Монолитное ядро
 - e) Модульное ядро
8. Ядро, в котором все части ядра работают в одном адресном пространстве
 - a) Наноядро
 - b) Микроядро
 - c) Экзоядро
 - d) Монолитное ядро
 - e) Модульное ядро
9. Более современное ядро операционной системы, в котором используется модульная структура не требующая перекомпиляции
 - a) Наноядро
 - b) Микроядро
 - c) Экзоядро
 - d) Монолитное ядро
 - e) Модульное ядро

Ключ: 1-d; 2-a; 3-c; 4-c; 5-a; 6-b; 7-c; 8-d; 9-e; 10-e

5.2. Типовые темы презентаций

Все темы докладов полностью совпадают с темами докладов/рефератов. Каждый студент может предложить свою тему презентации, выходящую за рамки предложенных тем.

5.4. Примерные темы докладов

1. Особенности операционной системы FreeBSD
2. Особенности операционной системы Icar OS
3. Особенности операционной системы ReactOS
4. Особенности операционной системы Syllable Desktop
5. Особенности операционной системы AIX
6. Особенности операционной системы BolgenOS
7. Особенности операционной системы BeOS
8. Особенности операционной системы NexentaOS
9. Особенности операционной системы Puppy Linux
10. Особенности операционной системы AROS
11. Особенности операционной системы Dreamlinux
12. Особенности операционной системы Parted Magic
13. Особенности операционной системы AtheOS
14. Особенности операционной системы Minix

15. Особенности операционной системы Linux Ubuntu
16. Особенности операционной системы Doudoulinux

5.5. Типовое кейс-задание лабораторной/практической работы

Задание 1. Запуск операционной системы Kolibri без установки на компьютер:

1. Запустите операционную систему согласно инструкции (скачать).
2. Вычислите в среде операционной системы Kolibri следующий пример $255+256*3$ и сделайте снимок экрана с получившимся результатом.
3. Определите назначение следующих программ: KFM, Mine, Animage, Fasm, TiniPad и заполните приведенную ниже таблицу в текстовом процессоре (например, OpenOffice.org Writer, LibreOffice.org Writer, AbiWord, Microsoft Office Word, LyX).

Название программы	Тип	Краткое описание

4. Подготовьте во встроенном в ОС Kolibri табличном редакторе таблицу "ПОКУПКИ" (пример). Сделайте снимок экрана.
5. Нарисуйте с помощью встроенного в ОС Kolibri графического редактора простенький пейзаж. Сделайте снимок экрана.

5.6. Вопросы к зачету/экзамену

1. Эволюционная история развития операционных систем.
2. Классификация операционных систем.
3. Структура операционной системы.
4. Ядро операционной системы. Типы архитектур ядра операционной системы.
5. Режим работы ядра: защищенный и пользовательский.
6. Понятие процесса. Процессы с поддержкой многопоточности.
7. Сигналы как простейшие средства коммуникации.
8. Типы памяти. Управление виртуальной памятью.
9. Управление внешней памятью. Файлы и файловые системы.
10. Распределенные файловые системы.
11. Драйверы устройств. Классы драйверов. Принципы управления устройствами ввода-вывода.
12. Стандартные сервисные программы современных ОС.
13. Сетевая подсистема операционной системы.
14. Командная строка и средства автоматизации выполнения задач.
15. Графический интерфейс операционных систем семейства UNIX.
16. Системы управления пакетами в операционных системах семейства UNIX.

5.7. Типовые темы курсовых работ

1. Приложение для реализации простейшей виртуальной файловой системы. Необходимо разработать консольное приложение для реализации простейшей обычной файловой системы, например FAT, в виде виртуальной файловой системы расположенной в файле.
2. Сервер Интернет-пейджера. Необходимо разработать сервер Интернет-пейджера по типу ICQ или Jabber. Сервер обеспечивает подключение до 100 клиентов. Поддерживается передача, как текстовых сообщений, так и файлов размером до 10 Мб.
3. Клиент Интернет-пейджера. Необходимо разработать клиент Интернет-пейджера по типу ICQ или Jabber. Клиент обеспечивает подключение к заданному серверу (указывается IP-адрес и порт). Поддерживает получение списка подключенных клиентов и позволяет передавать, как текстовые сообщения, так и файлов размером до 10 Мб.
4. Приложение для мониторинга состояния SMART регистров жесткого диска. Разработать консольное приложение для мониторинга состояния SMART регистров жесткого диска. Приложение выводит список всех дисковых устройств в системе и выдает полную информацию об указанном жестком диске как устройстве, его идентификационные данные, модель, состояние SMART регистров и т. д.
5. Сервер и клиент для хранения резервных копий файлов в сети. Необходимо разработать протокол для приема файлов по протоколу TCP от удаленных клиентов, на основе протокола следует разработать консольный сервер и консольный клиент, который будет архивировать принятые файл с помощью библиотеки zlib и раскладывать по каталогам клиентов для хранения. В каталоге клиента следует создавать подкаталоги с названиями дней, когда файлы были приняты. При передаче, следует оставлять исходное имя и расширение принятого файла, однако через точку указывать время приема, далее через точку следует указывать расширение архива (zip, gzip и т.д.).
6. Оконное приложение, которое позволяет построить графическую карту раздела диска, посредством которой можно определить какой файл, где находится (какие кластеры занимает). Необходимо разработать оконное Windows-приложение аналогичное Sysinternals DiskView.

7. Приложение для просмотра активных процессов в памяти. Необходимо разработать консольное приложение для просмотра активных процессов в памяти. Для каждого процесса следует показывать объем оперативной памяти, число открытых ресурсов, число чтений, число записей, число прочитанных байт, число записанных байт, возможность уничтожить выбранный процесс и все его дочерние процессы.
8. Приложение для получения полной информации о процессоре, модулях памяти и материнской плате. Необходимо разработать оконное Windows-приложение для получения полной информации о процессоре, модулях памяти и материнской плате. За основу следует взять информацию, которую можно получить посредством SiSoftware Sandra.
9. Приложение для создания виртуального CD/DVD диска. Необходимо разработать приложение для создания виртуального CD/DVD диска, с возможностью монтирования файлов ISO в качестве CD/DVD.
10. Приложение для записи CD/DVD диска посредством Image Mastering API. Необходимо разработать приложение для записи CD/DVD диска посредством Image Mastering API.
11. Приложение для мониторинга обращений к заданной директории/файлу файловой системы. Необходимо разработать приложение для мониторинга обращений к заданной директории/файлу. Все действия ОС по доступу к тому или ному файлу/директории необходимо заносить в журнал аудита расположенный в памяти, по окончании мониторинга результаты следует заносить в текстовый файл.
12. Приложение для мониторинга обращений к заданной ветке/значению реестра. Необходимо разработать консольное приложение для мониторинга обращений к заданной ветке/значению реестра. Все действия ОС по доступу к тому или ному значению/ветке реестра необходимо заносить в журнал аудита (текстовый файл).
13. Приложение для гарантированного уничтожения заданного файла/директории. Необходимо разработать консольное приложение для гарантированного уничтожения заданного файла/директории. Удаленную информацию невозможно восстановить посредством специализированного ПО.
14. Приложение для криптографической защиты файлов и каталогов с использованием алгоритма AES. Необходимо разработать консольное приложение, которое позволяет генерировать ключи, шифровать и расшифровывать с помощью алгоритма AES файлы и каталоги файловой системы. Доступ к секретному ключу (в файле контейнере), следует организовать посредством пароля. Криптографические преобразования выполнять посредством криптопровайдера Microsoft Windows CSP.
15. Приложение для криптографической защиты файлов и каталогов с использованием алгоритма AES. Необходимо разработать консольное приложение, которое позволяет генерировать ключи, шифровать и расшифровывать с помощью алгоритма AES файлы и каталоги файловой системы. Доступ к секретному ключу (в файле контейнере), следует организовать посредством пароля. Криптографические преобразования использовать из Intel Performance Primitives Cryptography Package.
16. Приложение для криптографической защиты файлов и каталогов с использованием алгоритма AES. Необходимо разработать консольное приложение, которое позволяет генерировать ключи, шифровать и расшифровывать с помощью алгоритма AES файлы и каталоги файловой системы. Доступ к секретному ключу (в файле контейнере), следует организовать посредством пароля. Криптографические преобразования использовать из библиотеки Crypto++.
17. Приложение для защищенного обмена файлами в сети Ethernet. Необходимо разработать консольное Windows-приложение для защищенного обмена файлами в сети Ethernet между двумя компьютерами. Один иницирует соединение, другой ожидает это соединение. Предполагается, что на стороне клиента и сервера уже имеются секретные ключи для организации защищенного файлового обмена. За основу следует взять socket соединения. Криптографические преобразования следует выполнять посредством криптопровайдера Microsoft Windows CSP.
18. Приложение для просмотра содержимого файлов образов *.iso для CD/DVD. Необходимо разработать консольное или оконное приложение для просмотра содержимого файлов образов дисков *.iso для CD/DVD.
19. Приложение для анализа логов web-сервера с использованием библиотеки Open MPI. Необходимо разработать консольное распределенное приложение с использованием библиотеки Open MPI. Предполагается, что приложение обрабатывает файлы логов, которые находятся на общедоступном сетевом ресурсе посредством различных компьютеров сети. Результаты работы приложения доступны через основное приложение.
20. Приложение, для восстановления файлов после удаления в файловых системах NTFS и FAT. Необходимо разработать оконное приложение, для восстановления файлов после удаления. Поддерживаются следующие файловые системы: NTFS, FAT.
21. Приложение для защищенного обмена файлами в сети Ethernet. Необходимо разработать консольное приложение для защищенного обмена файлами в сети Ethernet между двумя компьютерами. Один иницирует соединение, другой ожидает это соединение. Предполагается, что на стороне клиента и

сервера уже имеются секретные ключи для организации защищенного файлового обмена. За основу следует взять socket соединения. Криптографические преобразования следует выполнять посредством библиотеки OpenSSL.

22. Приложение, для восстановления файлов после удаления в файловых системах Ext2 и Ext4. Необходимо разработать оконное приложение, для восстановления файлов после удаления. Поддерживаются следующие файловые системы: Ext2, Ext4.
23. Приложение, для моделирования менеджера управления памятью со страничной организацией на основе алгоритма WSClock. Необходимо разработать приложение, для моделирования менеджера управления памятью со страничной организацией на основе алгоритма WSClock. Исследовать заданный алгоритм для различного размера страницы, общего размера памяти вычислительной системы, и т.д.
24. Приложение, для моделирования менеджера управления памятью со страничной организацией на основе алгоритма «Рабочий набор». Необходимо разработать приложение, для моделирования менеджера управления памятью со страничной организацией на основе алгоритма «Рабочий набор». Исследовать заданный алгоритм для различного размера страницы, общего размера памяти вычислительной системы, и т.д.
25. Приложение для побайтного сравнения содержимого двух каталогов, с поддержкой операций файлового ввода-вывода с нестабильным устройством хранения данных. Необходимо разработать приложение, с поддержкой операций файлового ввода-вывода с нестабильным устройством хранения данных. Предполагается, что могут происходить сбои во время открытия/закрытия, чтения/записи данных в файл.
26. Драйвер для подсчета нажатий клавиш и сохранения их скан-кодов в отдельный файл. Необходимо разработать WDF-драйвер, который фиксирует скан-коды всех нажатых клавиш в текстовом файле.
27. Приложение для чтения физических адресов оперативной памяти компьютера. Необходимо разработать оконное приложение для чтения физических адресов оперативной памяти компьютера. Приложение позволяет осуществлять: переход по заданному адресу, поиск заданной последовательности. Информация представляется в 16-ричной системе исчисления и в виде ASCII символов.
28. Приложение для чтения физических адресов памяти заданного накопителя компьютера. Необходимо разработать оконное приложение для чтения физических адресов памяти заданного накопителя компьютера. Приложение позволяет осуществлять: переход по заданному адресу, поиск заданной последовательности. Информация представляется в 16-ричной системе исчисления и в виде ASCII символов.
29. Приложение, которое использует старшие адреса памяти (выше 2-го Гбайта) с использованием Address Windowing Extensions API для 32-х разрядных ОС. Необходимо разработать консольное приложение, которое использует старшие адреса памяти (выше 2-го Гбайта) с использованием Address Windowing Extensions API (AWE API) для адресации виртуального 32-битного адресного пространства при наличии 3-х и более Гбайт физической оперативной памяти. Произвести замеры времени копирования различных блоков памяти в старшие адреса и из старших адресов.
30. Приложение для планирования выполнения задач, используя Windows Task Scheduler. Необходимо разработать оконное приложение для управления (планирования) задач используя Windows Task Scheduler. Приложение позволяет: создавать, редактировать и удалять задания для выполнения их ОС.
31. Приложение для анализа сетевого трафика используя Network Monitor API. Необходимо разработать оконное приложение для анализа сетевого трафика используя Network Monitor API (входит в Platform SDK).
32. Приложение для синхронизации заданных каталогов на 2-х компьютерах между собой. Необходимо разработать консольное приложение для синхронизации заданных папок на 2-х компьютерах между собой. Другими словами, на этих компьютерах запущено приложение, которое по команде проверяет содержимое заданной папки и производит синхронизацию с содержимым папки (папки не являются публичными) на другом компьютере (пример такого приложения GoodSync).
33. Приложение для защищенного обмена файлами используя Named Pipe. Необходимо разработать консольное приложение для защищенного обмена файлами используя Named Pipe между двумя компьютерами. Один иницирует соединение, другой ожидает это соединение. Предполагается, что на стороне клиента и сервера уже имеются секретные ключи для организации защищенного файлового обмена. За основу следует взять socket соединения. Криптографические преобразования следует выполнять посредством криптопровайдера Microsoft Windows CSP.
34. Консольное приложение для установки хуков для Диспетчера задач Windows. Windows-приложение которое устанавливает hook, который позволяет изменить заголовок окна Диспетчера задач Windows, например на «Курсовая работа студента 431 группы Николаева Петра Васильевича».

35. Консольное приложение для управления (планирования) задач. Необходимо разработать консольное приложение, которое позволяет: создавать, редактировать и удалять заданиями для выполнения их ОС, например cron-скрипты.
36. Консольное приложение для защищенного обмена файлами в сети. Необходимо разработать консольное приложение для защищенного обмена файлами в сети. За основу следует взять socket соединения. Криптографические преобразования следует выполнять посредством библиотеки OpenSSL. Клиент и сервер обязаны обладать соответствующими сертификатами, с длиной ключа RSA не менее 1024 бит.
37. Консольное приложение для получения информации о каталоге. Необходимо разработать приложение, которое может получить размер всех файлов, число файлов, число каталогов, которые находятся в заданном каталоге. В случае необходимости, может быть произведено рекурсивное сканирование всех каталогов, которые находятся в заданном каталоге. За пример следует взять приложение Sysinternals Disk Usage.
38. Консольное приложение для получения списка загруженных динамических DLL в оперативную память. Необходимо разработать консольное приложение, для отображения списка всех DLL загруженных в память, с указанием имени процесса, который ее загрузил, а также версию загруженной библиотеки. За пример следует взять приложение Sysinternals ListDLL.
39. Консольное приложение для дефрагментации наиболее часто используемых файлов. Необходимо разработать консольное приложение, которое может дефрагментировать (последовательно разместить все кластеры) файл. За пример следует взять приложение Sysinternals Contig.