

Министерство науки и высшего образования Российской Федерации
Нижекамский химико-технологический институт (филиал)
федерального государственного бюджетного образовательного учреждения
высшего образования
«Казанский национальный исследовательский технологический университет»
(НХТИ ФГБОУ ВО «КНИТУ»)

УТВЕРЖДАЮ

Заместитель директора по УР

Н.И. Никифорова

«12» _____ апреля 2021 г.



ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
по дисциплине (модулю)

Б1.В.05 Системное программное обеспечение
(код и наименование дисциплины (модуля))

09.03.01 Информатика и вычислительная техника
(код и наименование направления подготовки/специальности)

Автоматизированные системы обработки информации и управления
(наименование профиля/специализации)

бакалавр
квалификация

очная, очно-заочная, заочная
форма обучения

Составитель ФОС:

доцент

(должность)



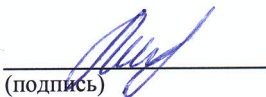
(подпись)

Л.Р. Вотякова

(Ф.И.О)

ФОС рассмотрен и одобрен на заседании кафедры ИСТ,
протокол от 15.03.2021 г. № 7

Зав. кафедрой



(подпись)

О.В. Матухина

(Ф.И.О.)

Эксперт:

Руководитель ООП

Ф.И.О., должность, организация, подпись

Л.А. Амаева



Перечень компетенций и индикаторов достижения компетенций с указанием этапов формирования в процессе освоения дисциплины

ПК-1 Способен разрабатывать требования и проектировать программное обеспечение

1. ПК-1.1: Знает методологии разработки программного обеспечения, назначение и возможности средств проектирования программного обеспечения
2. ПК-1.2: Умеет разрабатывать требования и проектировать программное обеспечение
3. ПК-1.3: Владеет навыками разработки требований к программным продуктам, использования методов и средств проектирования программного обеспечения

ПК-5 Выполнение работ и управление работами по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы

ПК-5.1 Знает стандартные средства интеграции разнородных решений в составе единой системы и методы объективного анализа различных вариантов; технологии построения прикладных и информационных процессов; современные подходы к улучшению информационных систем

ПК-5.2 Умеет осуществлять ведение базы данных и поддержку информационного обеспечения решения прикладных задач

ПК-5.3 Владеет навыками выбора класса ИС для автоматизации предприятия в соответствии с требованиями к ИС; способами автоматизации для конкретного предприятия

Индикаторы достижения компетенции	Этапы формирования в процессе освоения дисциплины				Наименован ие оценочного средства
	Лекции	Практические занятия	Лаборато рные занятия	Курсовой проект (работа)	
ПК-1.1	Разделы дисципли ны 1-7.	Не предусмотрены учебным планом	Разделы дисциплин ы 1-7.	Не предусмотрен ы учебным планом	Расчетно- графические работы, тест
ПК-1.2	Разделы дисципли ны 1-7.	Не предусмотрены учебным планом	Разделы дисциплин ы 1-7.	Не предусмотрен ы учебным планом	Расчетно- графические работы, тест
ПК-1.3	Разделы дисципли ны 1-7.	Не предусмотрены учебным планом	Разделы дисциплин ы 1-7.	Не предусмотрен ы учебным планом	Расчетно- графические работы, тест
ПК-5.1	Разделы дисципли ны 1-7.	Не предусмотрены учебным планом	Разделы дисциплин ы 1-7.	Не предусмотрен ы учебным планом	Расчетно- графические работы, тест
ПК-5.2	Разделы дисципли ны 1-7.	Не предусмотрены учебным планом	Разделы дисциплин ы 1-7.	Не предусмотрен ы учебным планом	Расчетно- графические работы, тест
ПК-5.3	Разделы дисципли ны 1-7.	Не предусмотрены учебным планом	Разделы дисциплин ы 1-7.	Не предусмотрен ы учебным планом	Расчетно- графические работы, тест

Перечень оценочных средств по дисциплине (модулю)

Очная, очно-заочная форма

Оценочные средства	Кол-во	Min, баллов (базовый уровень)	Max, баллов (повышенный уровень)
Расчетно-графические работы	1	15	40
Лабораторные работы	3	45	60

Заочная форма

<i>Оценочные средства</i>	<i>Кол-во</i>	<i>Min, баллов (базовый уровень)</i>	<i>Max, баллов (повышенный уровень)</i>
<i>Контрольная работа</i>	<i>1</i>	<i>36</i>	<i>60</i>
<i>Зачет с оценкой</i>	<i>1</i>	<i>24</i>	<i>40</i>
<i>Итого:</i>		<i>60</i>	<i>100</i>

Министерство образования и науки Российской Федерации
Нижекамский химико-технологический институт (филиал) федерального
государственного бюджетного образовательного учреждения высшего
образования
«Казанский национальный исследовательский технологический университет»

Направление подготовки/специальность: 09.03.01 Информатика и
вычислительная техника

Комплект заданий для выполнения расчетно-графической работы

РГР №1.

Вариант 1

Модель преобразования виртуального адреса в физический адрес

1. Исходные данные:

- организация виртуальной памяти – страничная,
- разрядность виртуального адреса – 32 бита,
- размер физической страницы – 2 Кбайт,
- максимальное число работающих процессов не более восьми,
- количество физических страниц в таблице страниц процесса не более четырех,
- объем оперативной памяти – 32 физических страницы,
- заполнение таблицы страниц – с использованием датчика случайных чисел,
- виртуальный адрес вводится с клавиатуры.

2. Результаты работы модели, отображаемые на дисплее должны включать:

- виртуальный адрес,
- номер процесса,
- содержимое таблицы страниц данного процесса,
- физический адрес.

Вариант 2

Модель преобразования виртуального адреса в физический адрес

1. Исходные данные:

- организация виртуальной памяти – двухуровневая страничная,
- разрядность виртуального адреса – 28 бит,
- размер физической страницы – 4 Кбайт,
- количество физических страниц в таблице страниц второго уровня – 256,
- объем оперативной памяти – 64 физических страницы,
- заполнение таблицы страниц – с использованием датчика случайных чисел,
- виртуальный адрес вводится с клавиатуры.

2. Результаты работы модели должны включать:

- виртуальный адрес,
- физический адрес,
- содержимое таблиц страниц первого и второго уровней.

Вариант 3

Модель преобразования виртуального адреса в физический адрес.

1. Исходные данные:

- организация виртуальной памяти – страничная с TLB (буфером быстрой переадресации),
- емкость TLB – 16 записей
- разрядность виртуального адреса – 32,
- размер физической страницы – 4 Кбайт,
- объем оперативной памяти – 256 физических страниц,
- количество физических страниц в таблице страниц процесса не более 32,
- заполнение таблицы страниц и TLB – датчиком случайных чисел,
- виртуальный адрес вводится с клавиатуры.

2. Результаты работы модели должны включать:

- виртуальный адрес,
- физический адрес,
- содержимое таблицы страниц и TLB.

Вариант 4

Модель преобразования виртуального адреса в физический адрес.

1. Исходные данные:

- организация виртуальной памяти – сегментная,
- число сегментов процесса – четыре,
- разрядность виртуального адреса – 32,
- объем оперативной памяти – 1 Гбайт,
- заполнение таблицы сегментов с клавиатуры,
- виртуальный адрес вводится с клавиатуры.

2. Результаты работы модели должны включать:

- виртуальный адрес,
- физический адрес,
- содержимое таблицы сегментов.

Вариант 5

Модель преобразования виртуального адреса в физический адрес.

1. Исходные данные:

- организация виртуальной памяти – сегментно-страничная, принятая в процессоре Pentium,
- разрядность виртуального адреса – 32,
- количество сегментов не более 16,
- размер физической страницы – 4 Кбайт,
- объем оперативной памяти – 256 физических страниц,
- количество физических страниц в таблице страниц процесса не более 32,
- виртуальный адрес вводится с клавиатуры.

2. Результаты работы модели должны включать:

- виртуальный адрес,
- физический адрес,
- содержимое таблицы сегментов и таблицы страниц.

Вариант 6

Модель распределения памяти фиксированными разделами

1. Исходные данные:

- объем оперативной памяти – 256 Мбайт,

- количество разделов 10,
- размер разделов выбирается исполнителем,
- очередь задач – общая,
- размер задачи – случайный – от 30 до 100 Мбайт,
- количество задач в очереди до 20.

2. Результаты работы модели должны включать:
состояние памяти после поступления очередной задачи

Вариант 7

Модель распределения памяти динамическими разделами

1. Исходные данные:

- объем оперативной памяти – 512 Мбайт,
- количество разделов до 15,
- очередь задач – общая,
- размер задачи – случайный – от 30 до 100 Мбайт,
- количество задач в очереди до 20.

2. Результаты работы модели должны включать:
состояние памяти после поступления очередной задачи

Вариант 8

Модель распределения памяти перемещаемыми разделами

1. Исходные данные:

- объем оперативной памяти – 256 Мбайт,
- количество разделов до 10,
- очередь задач – общая,
- размер задачи – случайный – от 30 до 100 Мбайт,
- количество задач в очереди -20.

2. Результаты работы модели должны включать:
состояние памяти после поступления очередной задачи

Вариант 9

Модель алгоритма замены страниц

1. Исходные данные:

- объем области замещения оперативной памяти (резидентное множество) – 5 страниц,
- количество различных страниц - 16,
- последовательность обращения к страницам - задана,
- алгоритм замены – дольше всех неиспользовавшаяся страница (LRU).

2. Результаты работы модели должны включать:

- состояние памяти после поступления очередной страницы,
- число страничных прерываний.

Вариант 10

Модель алгоритма замены страниц

1. Исходные данные:

- объем области замещения оперативной памяти (резидентное множество) – 4 страницы,
- количество различных страниц - 16,
- последовательность обращения к страницам - задана,
- алгоритм замены – “первым вошел – первым вышел” (FIFO).

2. Результаты работы модели должны включать:

- состояние памяти после поступления очередной страницы,
- число страничных прерываний.

Вариант 11

Модель алгоритма замены страниц.

1. Исходные данные:

- объем области замещения оперативной памяти (резидентное множество) – 4 страницы,
- количество различных страниц - 16,
- последовательность обращения к страницам - задана,
- алгоритм замены – “часовой”.

2. Результаты работы модели должны включать:

- состояние памяти после поступления очередной страницы,
- число страничных прерываний.

решаемой задачи в структурной организации операционной системы.

Вариант 12

Модель алгоритма замены страниц

1. Исходные данные:

- объем области замещения оперативной памяти (резидентное множество) – 4 страницы,
- количество различных страниц - 16,
- последовательность обращения к страницам - задана,
- алгоритм замены – “вторая попытка”.

2. Результаты работы модели должны включать:

- состояние памяти после поступления очередной страницы,
- число страничных прерываний.

Вариант 13

Модель алгоритма замены страниц

1. Исходные данные:

- объем области замещения оперативной памяти (резидентное множество) – 3 страницы,
- количество различных страниц - 16,
- последовательность обращения к страницам - задана,
- алгоритм замены – “не использовавшаяся в последнее время” (NRU).

2. Результаты работы модели должны включать:

- состояние памяти после поступления очередной страницы,
- число страничных прерываний.

Вариант 14

Модель алгоритма планирования потоков, основанного на квантовании

1. Исходные данные:

- две фиксированные очереди потоков разного приоритета с заданным временем выполнения,
- фиксированная величина кванта процессорного времени,
- количество процессоров - 1,
- циклическое выделение квантов потокам с учетом приоритета.

2. Результаты работы модели должны включать:

- среднее время выполнения потоков каждого приоритета.

Вариант 15

Модель алгоритма планирования потоков, основанного на квантовании

1. Исходные данные:

- фиксированная единая очередь потоков с заданным временем выполнения,
- фиксированная величина кванта процессорного времени,
- количество процессоров - 1,
- кратчайшая задача - первая.

2. Результаты работы модели должны включать:

- среднее время выполнения коротких и длинных потоков.

Вариант 16

Модель алгоритма планирования потоков, основанного на квантовании

1. Исходные данные:

- фиксированная единая очередь потоков с заданным временем выполнения,
- фиксированная величина кванта процессорного времени,
- количество процессоров - 2,
- циклическое выделение квантов потокам (круговое планирование).

2. Результаты работы модели должны включать:

- среднее время выполнения потоков.

Вариант 17

Модель алгоритма планирования потоков, основанного на квантовании.

1. Исходные данные:

- две фиксированных очереди потоков с заданным временем выполнения и разного приоритета,
- фиксированная величина кванта процессорного времени,
- количество процессоров - 2,
- первыми обрабатываются задачи высшего приоритета.

2. Результаты работы модели должны включать:

- среднее время выполнения потоков разного приоритета.

Вариант 18

Модель синхронизации потоков

1. Исходные данные:

- две программы, работающие с файлом в режиме разделения,
- первая программа записывает в файл произвольную информацию (по выбору исполнителя) в определенном количестве записей, после чего разрешает доступ к файлу второй программе;

- вторая программа удаляет записи, после чего разрешает доступ к файлу первой программе.

2. Результаты работы модели должны включать:

- печать содержимого файла обеими программами с временными отметками.

Вариант 19

Модель графа ресурсов и процессов

1. Исходные данные:

- в системе имеется N типов единичных ресурсов ($N \leq 10$),
- количество процессов, претендующих на ресурсы, - M ($M \leq 10$),

○ исходное состояние характеризуется некоторым распределением и запросами на ресурсы.

2. Результаты работы модели должны включать:

- граф текущего состояния ресурсов и процессов,
- выделение цикла в графе, если он есть.

Вариант 20

Модель обнаружения блокировок при наличии нескольких ресурсов каждого типа

1. Исходные данные:

- в системе имеется M типов разделяемых ресурсов ($M \leq 10$),
- количество процессов, претендующих на ресурсы, N ($N \leq 10$),
- исходное состояние характеризуется некоторым распределением и запросами на ресурсы.

2. Результаты работы модели должны включать:

- матрицу текущего распределения ресурсов,
- матрицу текущих запросов процессов на ресурсы,
- решение для текущего состояния (есть тупик или нет, запускать новый процесс или нет).

Вариант 21

Модель стека

1. Исходные данные:

- стек списковой структуры,
- перечень операций со стеком: создание, включение элемента, выборка элемента, извлечение данных, уничтожение.

2. Результаты работы модели должны включать:

- меню с перечнем всех операций над стеком,
- печать содержимого стека.

решаемой задачи в структурной организации операционной системы.

Вариант 22

Модель стека

1. Исходные данные:

- стек векторной структуры,
- перечень операций со стеком: создание, включение элемента, выборка элемента, извлечение данных, уничтожение.

2. Результаты работы модели должны включать:

- меню с перечнем всех операций над стеком,
- печать содержимого стека.

Вариант 23

Модель очереди

1. Исходные данные:

- очередь векторной структуры,
- перечень операций с очередью: создание и освобождение, включение в очередь нового элемента, выборка элемента из очереди,
- дисциплина очереди – FIFO (добавление в конец очереди, выборка из головы очереди).

2. Результаты работы модели должны включать:

- меню с перечнем всех операций над очередью,
- печать содержимого очереди.

Вариант 24

Модель очереди

1. Исходные данные:

- очередь списковой структуры,
- перечень операций с очередью: создание и освобождение, включение в очередь нового элемента, выборка элемента из очереди,
- дисциплина очереди – FIFO (добавление в конец очереди, выборка из головы очереди).

2. Результаты работы модели должны включать:

- меню с перечнем всех операций над очередью,
- печать содержимого очереди

Вариант 25

Модель очереди

1. Исходные данные:

- очередь векторной структуры с динамической памятью,
- перечень операций с очередью: создание и освобождение, включение в очередь нового элемента, выборка элемента из очереди,
- дисциплина очереди – FIFO (добавление в конец очереди, выборка из головы очереди).

2. Результаты работы модели должны включать:

- меню с перечнем всех операций над очередью,
- печать содержимого очереди.

Критерии оценки

Оценочные средства	Min, баллов (базовый уровень)	Max, баллов (повышенный уровень)
Расчетно-графическая работа	15	40

Министерство образования и науки Российской Федерации
Нижекамский химико-технологический институт (филиал) федерального
государственного бюджетного образовательного учреждения высшего
образования
«Казанский национальный исследовательский технологический университет»

Направление подготовки/специальность: 09.03.01 Информатика и вычислительная техника

Комплект заданий для выполнения лабораторных работ

Лабораторная работа №1

Задание 1. Работа с Диспетчером задач Windows 7.

1. Запустите ранее установленную ОС Windows 7.
2. Запуск диспетчера задач можно осуществить двумя способами:
 - 1) Нажатием сочетания клавиш Ctrl+Alt+Del. При использовании данной команды не стоит пренебрегать последовательностью клавиш. Появится меню, в котором курсором следует выбрать пункт «Диспетчер задач».
 - 2) Переведите курсор на область с показаниями системной даты и времени и нажмите правый клик, будет выведено меню, в котором следует выбрать «Диспетчер задач».
3. В диспетчере задач есть 6 вкладок:
 - 1) Приложения
 - 2) Процессы
 - 3) Службы
 - 4) Быстродействие
 - 5) Сеть
 - 6) Пользователи
- Вкладка «**Приложения**» отображает список запущенных задач (программ) выполняющиеся в настоящий момент не в фоновом режиме, а также отображает их состояние. Также в данном окне можно снять задачу переключиться между задачами и запустить новую задачу при помощи соответствующих кнопок.
- Вкладка «**Процессы**» отображает список запущенных процессов, имя пользователя запустившего процесс, загрузку центрального процессора в процентном соотношении, а также объем памяти используемого для выполнения процесса. Также присутствует возможность отображать процессы всех пользователей, либо принудительного завершения процесса. Процесс — выполнение пассивных инструкций компьютерной программы на процессоре ЭВМ.
- Вкладка «**Службы**» показывает, какие службы запущены на компьютере. Службы — приложения, автоматически запускаемые системой при запуске ОС Windows и выполняющиеся вне зависимости от статуса пользователя.
- Вкладка «**Быстродействие**» отображает в графическом режиме загрузку процессора, а также хронологию использования физической памяти компьютера. Очень эффективным инструментом наблюдения является «Монитор ресурсов». С его

помощью можно наглядно наблюдать за каждой из сторон «жизни» компьютера. Подробное изучение инструмента произвести самостоятельно, интуитивно.

- Вкладка «Сеть» отображает подключенные сетевые адаптеры, а также сетевую активность.
 - Вкладка «Пользователи» отображает список подключенных пользователей.
5. После изучения диспетчера задач:
- Потренируйтесь в завершении и повторном запуске процессов. Разберитесь мониторинг загрузки и использование памяти.
6. Попробуйте запустить новые процессы при помощи диспетчера, для этого можно использовать команды: cmd, msconfig.

Задание 2. Работа с процессами через командную строку Windows.

1. Для запуска командной строки в режиме Windows следует нажать:
Пуск \ Все программы \ Стандартные \ Командная строка
2. Перейдите в папку C:\Windows и просмотрите его содержимое.
3. Поработайте над выполнением основных команд работы с процессами: запуская, отслеживая и завершая процессы. **Основные команды:**
 - a. Schtasks - выводит выполнение команд по расписанию
 - b. Start - запускает определенную программу или команду в отдельном окне.
 - c. Taskkill - завершает процесс
 - d. Tasklist - выводит информацию о работающих процессах
4. Запустим программу «Блокнот»:
C:\Windows > start notepad.exe
Отследим выполнение процесса:
C:\Windows > tasklist
Затем завершите выполнение процесса: C:\Windows > taskkill /IM notepad.exe
5. Самостоятельно, найдите команду запуска программы WordPad. Необходимый файл запуска найдите в папке Windows.
6. Выполнение задания включить в отчет по выполнению лабораторной работы.

Задание 3. Самостоятельное задание.

1. Отследите выполнение процесса explorer.exe при помощи диспетчера задач и командной строки.
2. Продемонстрируйте преподавателю завершение и повторный запуск процесса explorer.exe из:
 - Диспетчера задач;
 - Командной строки.
3. Выполнение задания включить в отчет по выполнению лабораторной работы.

Задание 4. Изучение команд для работы с файлами

Команда Сору

- Ознакомьтесь со справкой по команде **Сору**.
- Скопируйте все файлы с определенным расширением, расположенные вместе, путь к которому задайте самостоятельно, в точку назначения, заданную путем d:\Temp\.

- Скопируйте файл, расположенный в месте, путь к которому задайте самостоятельно, в точку назначения, заданную другим путем. Иницируйте запрос на подтверждение перезаписи конечного файла в случае, если он существует.
- Продублируйте файл с определенным именем, путь к которому задайте самостоятельно, в точку назначения, заданную тем же путем, добавив к началу имени файла строку «сору-».
- Объедините два текстовых (.txt) файла, пути к которым задайте самостоятельно, в один файл с полным именем d:\Temp\Merged.txt.
- Введите фрагмент текста с клавиатуры, используя ее *источник* Cop, в текстовый файл, путь к которому задайте самостоятельно. Признаком конца ввода строки является Enter. Признаком конца ввода текста в файл являются нажатые клавиши Ctrl+Z и Enter.
- Добавьте несколько строк с клавиатуры в конец существующего текстового файла, полученного в предыдущем пункте текущего задания.

Команда **Xcopy**

- Ознакомьтесь со справкой по команде **Xcopy**.
- Скопируйте все файлы и подкаталоги, включая пустые и скрытые, расположенные в месте, путь к которому задайте самостоятельно, в точку назначения на другом локальном диске. При этом иницируйте запрос на подтверждение перезаписи.
- Скопируйте дерево каталогов, включая пустые, расположенные в месте, путь к которому задайте самостоятельно, в точку назначения на другом локальном диске.
- Скопируйте все файлы с атрибутами «архивный» и «только для чтения» с сохранением этого атрибута для файлов-результатов, расположенные в месте, путь к которому задайте самостоятельно, в точку назначения, заданную путем d:\Temp\.
- Скопируйте все файлы и подкаталоги с датой не позднее определенной. Путь к *источнику* и точке *назначения* задайте самостоятельно. Отобразите список файлов в процессе копирования.

Команда **Move**

- Ознакомьтесь со справкой по команде **Move**.
- Скопируйте пять любых файлов с определенным расширением, расположенные в месте источника, путь к которому выберите самостоятельно, в точку назначения, заданную путем d:\Temporary\. При копировании воспользуйтесь любым методом, изученным ранее.
- Используя команду Move
- Воспользовавшись командой единожды, переместите все только что скопированные файлы, заданные путем d:\Temporary\, обратно в место *источника*. При этом иницируйте вывод запроса на подтверждение перезаписи.

Команда **Replace**

- Ознакомьтесь со справкой по команде **Replace**.
- Скопируйте три любых файла, расположенные в месте каталога - *источника*, путь к которому выберите самостоятельно, в каждый из двух каталогов-*назначения*, заданных следующими путями d:\Temp\Begin\ и d:\Temp\End\. При копировании воспользуйтесь

любым методом, изученным ранее.

- Используя команду **Replace**
- Замените первый по порядку файл в каталоге - назначения d:\Temp\End\ файлом, расположенным в каталоге - источнике d:\Temp\Begin\, осуществив подтверждение замены.
- Замените второй по порядку файл с более ранней датой модификации и путем - назначения d:\Temp\End\ файлом, расположенным в каталоге - источнике d:\Temp\Begin\, предварительно каким-либо образом его модифицировав.
- Активируйте атрибут «только для чтения» у третьего по порядку файла в каталогах d:\Temp\Begin\ и d:\Temp\End\. Замените третий по порядку файл в каталоге - назначения d:\Temp\End\ файлом, расположенным в каталоге - источнике d:\Temp\Begin\.

Команда **Rename**

- Ознакомьтесь с справкой по команде **Ren (Rename)**.
- Скопируйте пять любых файлов с определенными разрешениями, расположенные в месте, путь к которому выберите самостоятельно, в точку назначения, заданную путем d:\Temp\. При копировании воспользуйтесь любым методом, изученным ранее.
- Используя команду **Ren (Rename)**
- Измените типы всех скопированных файлов, заданных путем d:\Temp\, на другой, выбранный самостоятельно тип.
- Переименуйте все файлы, заданные путем d:\Temp\, в файлы с именами Renamed1.Ren, Renamed2.Ren, ... , Renamed5.Ren.

Команда **Fc**

- Ознакомьтесь со справкой по команде **Fc**.
- Используя команду **Fc**
- Сравните два текстовых файла, пути к которым задайте самостоятельно. Результат сравнения выведите в файл Result.txt
- Сравните два бинарных файла, пути к которым задайте самостоятельно. Результат сравнения добавьте в файл Result.txt

Команда **Del (Delete) и Erase**

- Ознакомьтесь со справкой по команде **Del (Delete) и Erase**.
- Скопируйте все файлы, расположенные в месте, путь к которому выберите самостоятельно, в точку назначения, заданную путем d:\Temp\. При копировании воспользуйтесь любым методом, изученным ранее.
- Используя команду **Del (Delete) и Erase**
- Удалите выбранный самостоятельно файл, заданный путем d:\Temp\, запросив подтверждение на удаление.
- Удалите все файлы с атрибутом «Системный», расположенные в месте, заданном путем d:\Temp\. Подтверждение на удаление не выводить.
- Удалите все файлы с определенным расширением, расположенные в месте, заданном путем d:\Temp\, запросив подтверждение на удаление.
- Удалите все оставшиеся файлы, включая каталоги, расположенные в месте, заданном путем d:\Temp\. Подтверждение на удаление не выводить.

Задание 5. Изучение команд для работы с дисками

Команда **Dir**

- Ознакомьтесь со справкой по команде **Dir**.
- Выведите постранично содержимое каталога C:\Windows\, включая вложенные подкаталоги и файлы.
- Выведите постранично все каталоги и файлы на локальном диске D: в алфавитном порядке с сортировкой по столбцам и паузой после заполнения каждого экрана.
- Выведите все файлы с расширением .doc на локальном диске D: в алфавитном порядке с сортировкой по колонкам. Вывод осуществите в файл Doc-Files.txt (Приложение 1).
- Выведите все каталоги на локальном диске C: в алфавитном порядке. Результат добавьте в файл DocFiles.txt (Приложение 1).
- Добавьте сведения о владельцах файлов системного каталога C:\Windows\ в файл DocFiles.txt (Приложение 1).

Команда **cd (ChDir)**

- Ознакомьтесь со справкой по команде **Cd (ChDir)**.
- Используя команду **Cd (ChDir)**
- Смените текущий каталог на каталог, полный путь к которому задан следующим образом C:\WINDOWS\Help\Tours\WindowsMediaPlayer\Video\.
- Перейдите из подкаталога ..\Video на уровень выше.
- Смените текущий каталог на каталог, полный путь к которому задан следующим образом C:\WINDOWS\Help\Tours\WindowsMediaPlayer\Audio\.
- Перейдите из подкаталога ... \Audio на два уровня выше.
- Смените текущий локальный диск на диск D:

Команда **Md (MkDir)**

- Ознакомьтесь со справкой по команде **Md (MkDir)**.
- Используя команду **Md (MkDir)**
- Создайте каталог, путь к которому выберите самостоятельно.
- Единожды воспользовавшись командой, создайте каталог, полный путь к которому задан следующим образом d:\Temp\VMGroup\MyPath\.

Команда **Rd (Rmdir)**

- Ознакомьтесь со справкой по команде **Rd (Rmdir)**.
- Используя команду **Rd (Rmdir)**
- Удалите подкаталог третьего уровня MyPath, созданный в предыдущем задании №1к.
- Скопируйте несколько файлов, расположенных в месте, путь к которому выберите самостоятельно, в точку назначения, заданную путем d:\Temp\VMGroup\.. При копировании воспользуйтесь любым методом, изученным ранее. Единожды воспользовавшись командой, без запроса подтверждения удалите дерево каталогов d:\Temp\VMGroup\, включая подкаталог второго уровня VMGroup с содержащимися внутри файлами.

Команда **Format**

- Ознакомьтесь со справкой по команде **Format**.
- Для форматирования используйте USB - диск.
- Используя команду Format
- Отформатировать диск с использованием размера кластера по умолчанию.
- Повторно быстро отформатируйте ранее отформатированный гибкий диск, находящийся в накопителе.

Команда **Label**

- Ознакомьтесь со справкой по команде **Label**.
- Отобразите метку тома, по умолчанию присвоенную диску, находящемуся в накопителе.
- Присвойте новую метку тома диску.
- Отобразите новую метку тома диска.

Лабораторная работа №2

Задание 1. Изучение команды SET

1. Отобразите переменные среды двумя способами: из командной оболочки и окна свойств системы (Пуск | Панель управления | Система).
2. Задайте переменную среды, содержащую определенный путь к месту назначения, выбранный самостоятельно.
3. Проверьте наличие в системе переменной среды, заданной в предыдущем пункте задания.
4. Выведите значение выражения, определенного в соответствии с вариантом задания (табл. 8), в качестве переменной среды Result.
5. Задайте переменную среды с различными вариантами динамически формируемых значений (табл. 4). Варианты динамических значений выберете самостоятельно.

При выполнении задания используйте следующие инструкции:

- по каждому из пунктов задания в окне командной оболочки наберите соответствующую команду с необходимыми ключами,
- нажмите **Enter** для ввода,
- изучите полученный результат и сделайте вывод о проделанной работе,
- запишите полученную информацию в отчет, заполнив табл. 5.

Таблица 5. Результаты выполнения команды Set

№ п.п.	Команда с ключами	Результат и вывод по способу применения команды
1		
2		
3		
4		
5		

Задание 2. Изучение команд REM и ECHO

Создайте пакетный файл, воспользовавшись любым текстовым редактором. Имя пакетного файла выберете самостоятельно.

1. Введите в созданный пакетный файл текст, приведенного выше примера.

2. Сохраните текст пакетного файла.

При выполнении задания используйте следующие инструкции:

- воспользовавшись командой Start и указав путь к пакетному файлу,
- запустите его на выполнение, нажав Enter для ввода,
- изучите пример и полученный с его помощью результат, обратив внимание
- на то, что команда Echo с точкой (.) в конце выводит на экран пустую строку, а символ «коммерческое И» (@) перед командой Echo отключает режим отображения команд.
- сделайте соответствующий вывод и запишите его в отчет.

Задание 3. Изучение утилиты FOR

1. Скопируйте файлы каталога, путь к которому задайте самостоятельно, в точку назначения, заданную путем d:\Temp\. При копировании воспользуйтесь любым методом, изученным ранее.
2. К каждому из файлов, местоположение которых определено путем d:\Temp\, добавьте символ «!» в начале имени, воспользовавшись командой циклической обработки данных.
3. Подсчитать количество каталогов на локальном диске, воспользовавшись командой циклической обработки данных, в процессе выполнения вывода результат в переменную среды, выбранную самостоятельно. Проверьте полученный результат в файловом диспетчере Total Commander (Файл | Подсчитать занимаемое место), предварительно выделив содержимое локального диска.
4. Модифицируйте пакетный файл, полученный в предыдущем задании, воспользовавшись командой циклической обработки данных таким образом, чтобы в процессе его выполнения отображалось определенное количество раз выражение «***** the For command *****».

При выполнении задания используйте следующие инструкции: по каждому из пунктов задания в окне командной оболочки наберите соответствующую команду с необходимыми ключами, нажмите Enter для ввода, изучите полученный результат и сделайте вывод о проделанной работе, запишите полученную информацию в отчет, заполнив табл. 6.

Таблица 6. Результаты выполнения команды For

№ п.п.	Команда с ключами	Результат и вывод по способу применения команды
1		
2		
3		
4		
5		

Задание 4. Изучение команды IF

Модифицируйте пакетный файл, полученный в предыдущем задании таким образом, чтобы выполнялись следующие условия:

1. Если не существует каталог d:\Temp\MyFont\, создайте его любым способом, изученным ранее. В противном случае выведите сообщение «Folder exists» (Каталог существует).
2. Если в каталоге d:\Temp\MyFont\ не существует файлов-шрифтов, скопируйте любые три

одним из методов, изученных ранее, из системного каталога c:\Windows\Fonts\ . В противном случае выведите сообщение «Fonts exist» (Шрифты присутствуют).

3. Если в каталоге d:\Temp\MyFont\ существует файлы, удалите каталог вместе с его содержимым, изученным ранее способом и выведите сообщение «Folder deleted». В противном случае выведите сообщение «Folder is empty. Deleting is senseless» (Каталог пуст. Удаление бессмысленно).

При выполнении задания используйте следующие инструкции:

- по каждому из пунктов задания в командном файле наберите соответствующий код из команд с необходимыми ключами,
- сохраните модифицированный пакетный файл, воспользовавшись командой Start и указав путь к пакетному файлу,
- запустите его на выполнение, нажав Enter для ввода,
- изучите полученный результат и сделайте вывод о проделанной работе,
- запишите полученную информацию в отчет.

Задание 5. Изучение команды GOTO

Модифицируйте существующий пакетный файл, введя в него следующий текст:

```
Pause
Echo.
Format
If not Errorlevel 1 Goto End
Echo.
Echo *** Error of formatting ***
Rem *** Ошибка форматирования ***
Echo *** The end of batch program ***
Rem *** Конец пакетной программы ***
Echo.
Pause
```

Сохраните текст пакетного файла.

При выполнении задания используйте следующие инструкции: воспользовавшись командой **Start** и указав путь к пакетному файлу,

- запустите его на выполнение, нажав **Enter** для ввода,
- изучите пример и полученный с его помощью результат,
- сделайте вывод о проделанной работе и запишите его в отчет.

Задание 6. Изучение принципов работы команды CALL

1. Создайте новый (дочерний) пакетный файл, воспользовавшись любым текстовым редактором. Имя пакетного файла выберите самостоятельно.
2. Введите в дочерний пакетный файл процедуру форматирования гибкого диска, учитывающую переход в начало процедуры в случае ошибки, из приведенного выше примера.
3. Модифицируйте родительский пакетный файл, удалив из него лишние команды и добавив ссылку на дочерний пакетный файл для его вызова.
4. Сохраните тексты обоих пакетных файлов.

При выполнении пунктов 1-4 задания используйте следующие инструкции:

- воспользовавшись командой Start и указав путь к родительскому файлу, запустите его на выполнение, нажав Enter для ввода,

- изучите полученный результат и сделайте вывод о проделанной работе,
- запишите полученную информацию в отчет.
- 5. Вспомните команду форматирования **Format** и ее параметры.
- 6. Модифицируйте родительский и дочерний файлы таким образом, чтобы осуществилась передача из родительского файла двух значений параметров (%переменная) команды Format (табл. 9), находящейся внутри дочернего файла. Обратите внимание на то, что в таблице 9 передаваемые параметры команды Format имеют числовое (%0-%9), а не символьное представление.
- 7. Сохраните тексты обоих пакетных файлов.

При выполнении пунктов 5-7 задания используйте следующие инструкции:

- воспользовавшись командой Start и указав путь к родительскому файлу с параметрами для команды Format, запустите его на выполнение, нажав Enter для ввода,
- изучите полученный результат и сделайте вывод о проделанной работе, перенесите тексты модифицированных пакетных файлов, а также значения используемых пакетных параметров в отчет.

Варианты для заданий 1, 6.

Таблица 9. Варианты для заданий 1 и 6.

Вар. №	Задание 1.				Задание 6.	
	Переменная среды				Параметры команды Format	
	a	b	c	Result	%1	%2
1	24	11	35	$(a + b - c) * 10$	/v:System	/a:512
2	AA	01	C1	$a * 5 - b / 5 + c$	/a:512	/q
3	12	33	10	$a * 4 / c - b * 2$	/v:IICT	/a:1024
4	25	A3	B4	$a - b * 3 - c / 2$	/a:1024	/q
5	49	02	65	$a - b * b + c / 5$	/v:VMgroup	/a:2048
6	21	99	12	$(b * (a + c)) / 3$	/a:2048	/q
7	BC	BC	CB	$10 + (a - b) * c$	/v:MyCore	/a:4096
8	01	94	04	$(b - c) / (a * 9)$	/a:4096	/q
9	84	D2	2A	$a / 10 - (b * c)$	/v:Useless	/a:8192
10	10	39	92	$a * a + b - c / 2$	/v:MyDisk	/q
11	D1	CC	1C	$a * b * c - b / c$	/v:Temp	/q
12	FF	00	F1	$(a - c) * b + 25$	/q	/v:Apps
13	CA	DA	FA	$a * b * (FA - c)$	/v:Double	/q
14	45	78	87	$((c - b) * a) / 8$	/q	/v:HomeUse
15	88	88	00	$(a - b) / (1 - c)$	/v:MyDocs	/q
16	75	93	02	$(b + c) * 2 - a$	/q	/v:VMComp
17	A1	CD	0E	$a * a * b * b * c * c$	/v:MyList	/a:512
18	C4	EA	E3	$(a + b + c) * a$	/a:512	/q
19	44	55	33	$c * (b - a) / 10$	/v:Admin	/a:1024
20	B3	E5	DE	$(a * b * c) / 25$	/a:1024	/q
21	BB	ED	AE	$(a - b) * 2 - c$	/v:SysCore	/a:2048
22	55	56	31	$(b - a) * 31 - c$	/a:2048	/q
23	D1	EC	EE	$(c - b) * 5 / a$	/v:Kernel	/a:4096
24	D6	E6	FE	$(a + b) * c - 11$	/a:4096	/q
25	71	65	32	$(a - b) * c / 32$	/v:User	/a:8192
26	84	32	10	$(a * (b + c)) / 5$	/a:8192	/q

Задание 7. Использование команд Setlocal, Endlocal

Модифицируйте существующий пакетный файл, введя в него следующий текст, иллюстрирующий локальное изменение переменных среды:

```
@Echo off
Echo.
Echo *** Local changing the environment variables ***
Rem *** Локальное изменение переменных среды ***
Setlocal
Path=c:\Windows\system32\help;%path%
Call help>c:\help.out

Endlocal
Start notepad c:\help.out
Pause
Сохраните текст пакетного файла.
```

При выполнении задания используйте следующие инструкции:

- о воспользовавшись командой Start и указав путь к пакетному файлу, запустите его на выполнение, нажав Enter для ввода,
- о изучите пример и полученный с его помощью результат,
- о сделайте вывод о проделанной работе и запишите его в отчет.

Лабораторная работа №3

Задание:

1. Для выполнения данной работы будем использовать ранее установленный Linux Ubuntu. Запускаем Linux. После прохождения идентификации включаем терминал (рис. 34).

Applications > Accessories > Terminal

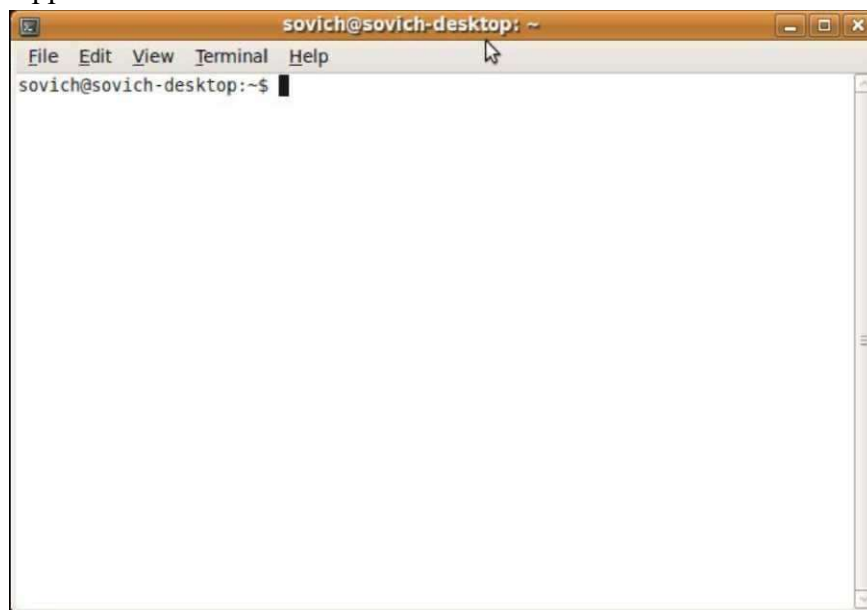


Рис. 34. Терминал Linux Ubuntu

2. Для работы в терминале Ubuntu требуются права пользователя root, но, к сожалению, по умолчанию, он недоступен, поэтому для выполнения некоторых (не всех) команд

надо писать `sudo <команда>`, и подтверждать свои права вводом пароля. И не пугайтесь того, что его не видно в терминале! Наберите точно по памяти, по окончании ввода нажмите Enter.

3. Для получения справки о дополнительных возможностях некоторых программ следует набрать `<команда> --help`

4. Потренируйтесь в выполнении команд:

- `date`
- `oclock`
- `finger`
- `hwclock`
- `uname`
- `history`
- `clear`
- `ls`

Найдите данные команды в таблице 2, опишите их. После выполнения результат внесите в отчет.

5. Создайте нового пользователя, при помощи терминала Ubuntu, и введите его в группу admin. Создайте пароль пользователю. Войдите под ним в систему. Процесс создания и ввода в группу внесите в отчет.

6. Разберите выполнение незадействованных команд таблицы 2. Потренируйтесь в выполнении, определите их назначение и область применения. Результат работы внесите в отчет.

7. Подготовьте отчет о выполнении лабораторной работы и сдайте преподавателю в соответствии с графиком.

Критерии оценки

Оценочные средства	Min, баллов (базовый уровень)	Max, баллов (повышенный уровень)
Лабораторная работа №1	15	20
Лабораторная работа №2	15	20
Лабораторная работа №3	15	20

Министерство образования и науки Российской Федерации
Нижекамский химико-технологический институт (филиал) федерального
государственного бюджетного образовательного учреждения высшего
образования
«Казанский национальный исследовательский технологический университет»

Направление подготовки/специальность: 09.03.01 Информатика и
вычислительная техника

Комплект заданий для выполнения контрольной работы

Вариант 1

Модель преобразования виртуального адреса в физический адрес

1. Исходные данные:

- организация виртуальной памяти – страничная,
- разрядность виртуального адреса – 32 бита,
- размер физической страницы – 2 Кбайт,
- максимальное число работающих процессов не более восьми,
- количество физических страниц в таблице страниц процесса не более четырех,
- объем оперативной памяти – 32 физических страницы,
- заполнение таблицы страниц – с использованием датчика случайных чисел,
- виртуальный адрес вводится с клавиатуры.

2. Результаты работы модели, отображаемые на дисплее должны включать:

- виртуальный адрес,
- номер процесса,
- содержимое таблицы страниц данного процесса,
- физический адрес.

Вариант 2

Модель преобразования виртуального адреса в физический адрес

1. Исходные данные:

- организация виртуальной памяти – двухуровневая страничная,
- разрядность виртуального адреса – 28 бит,
- размер физической страницы – 4 Кбайт,
- количество физических страниц в таблице страниц второго уровня – 256,
- объем оперативной памяти – 64 физических страницы,
- заполнение таблицы страниц – с использованием датчика случайных чисел,
- виртуальный адрес вводится с клавиатуры.

2. Результаты работы модели должны включать:

- виртуальный адрес,
- физический адрес,
- содержимое таблиц страниц первого и второго уровней.

Вариант 3

Модель преобразования виртуального адреса в физический адрес.

1. Исходные данные:

- организация виртуальной памяти – страничная с TLB (буфером быстрой переадресации),
- емкость TLB – 16 записей
- разрядность виртуального адреса – 32,
- размер физической страницы – 4 Кбайт,
- объем оперативной памяти – 256 физических страниц,
- количество физических страниц в таблице страниц процесса не более 32,
- заполнение таблицы страниц и TLB – датчиком случайных чисел,
- виртуальный адрес вводится с клавиатуры.

2. Результаты работы модели должны включать:

- виртуальный адрес,
- физический адрес,
- содержимое таблицы страниц и TLB.

Вариант 4

Модель преобразования виртуального адреса в физический адрес.

1. Исходные данные:

- организация виртуальной памяти – сегментная,
- число сегментов процесса – четыре,
- разрядность виртуального адреса – 32,
- объем оперативной памяти – 1 Гбайт,
- заполнение таблицы сегментов с клавиатуры,
- виртуальный адрес вводится с клавиатуры.

2. Результаты работы модели должны включать:

- виртуальный адрес,
- физический адрес,
- содержимое таблицы сегментов.

Вариант 5

Модель преобразования виртуального адреса в физический адрес.

1. Исходные данные:

- организация виртуальной памяти – сегментно-страничная, принятая в процессоре Pentium,
- разрядность виртуального адреса – 32,
- количество сегментов не более 16,
- размер физической страницы – 4 Кбайт,
- объем оперативной памяти – 256 физических страниц,
- количество физических страниц в таблице страниц процесса не более 32,
- виртуальный адрес вводится с клавиатуры.

2. Результаты работы модели должны включать:

- виртуальный адрес,
- физический адрес,
- содержимое таблицы сегментов и таблицы страниц.

Вариант 6

Модель распределения памяти фиксированными разделами

1. Исходные данные:

- объем оперативной памяти – 256 Мбайт,
- количество разделов 10,

- размер разделов выбирается исполнителем,
- очередь задач – общая,
- размер задачи – случайный – от 30 до 100 Мбайт,
- количество задач в очереди до 20.

2. Результаты работы модели должны включать:
состояние памяти после поступления очередной задачи

Вариант 7

Модель распределения памяти динамическими разделами

1. Исходные данные:

- объем оперативной памяти – 512 Мбайт,
- количество разделов до 15,
- очередь задач – общая,
- размер задачи – случайный – от 30 до 100 Мбайт,
- количество задач в очереди до 20.

2. Результаты работы модели должны включать:
состояние памяти после поступления очередной задачи

Вариант 8

Модель распределения памяти перемещаемыми разделами

1. Исходные данные:

- объем оперативной памяти – 256 Мбайт,
- количество разделов до 10,
- очередь задач – общая,
- размер задачи – случайный – от 30 до 100 Мбайт,
- количество задач в очереди -20.

2. Результаты работы модели должны включать:
состояние памяти после поступления очередной задачи

Вариант 9

Модель алгоритма замены страниц

1. Исходные данные:

- объем области замещения оперативной памяти (резидентное множество) – 5 страниц,
- количество различных страниц - 16,
- последовательность обращения к страницам - задана,
- алгоритм замены – дольше всех неиспользовавшаяся страница (LRU).

2. Результаты работы модели должны включать:

- состояние памяти после поступления очередной страницы,
- число страничных прерываний.

Вариант 10

Модель алгоритма замены страниц

1. Исходные данные:

- объем области замещения оперативной памяти (резидентное множество) – 4 страницы,
- количество различных страниц - 16,
- последовательность обращения к страницам - задана,
- алгоритм замены – “первым вошел – первым вышел” (FIFO).

2. Результаты работы модели должны включать:

- состояние памяти после поступления очередной страницы,
- число страничных прерываний.

Вариант 11

Модель алгоритма замены страниц.

1. Исходные данные:

- объем области замещения оперативной памяти (резидентное множество) – 4 страницы,
- количество различных страниц - 16,
- последовательность обращения к страницам - задана,
- алгоритм замены – “часовой”.

2. Результаты работы модели должны включать:

- состояние памяти после поступления очередной страницы,
- число страничных прерываний.

решаемой задачи в структурной организации операционной системы.

Вариант 12

Модель алгоритма замены страниц

1. Исходные данные:

- объем области замещения оперативной памяти (резидентное множество) – 4 страницы,
- количество различных страниц - 16,
- последовательность обращения к страницам - задана,
- алгоритм замены – “вторая попытка”.

2. Результаты работы модели должны включать:

- состояние памяти после поступления очередной страницы,
- число страничных прерываний.

Вариант 13

Модель алгоритма замены страниц

1. Исходные данные:

- объем области замещения оперативной памяти (резидентное множество) – 3 страницы,
- количество различных страниц - 16,
- последовательность обращения к страницам - задана,
- алгоритм замены – “не использовавшаяся в последнее время” (NRU).

2. Результаты работы модели должны включать:

- состояние памяти после поступления очередной страницы,
- число страничных прерываний.

Вариант 14

Модель алгоритма планирования потоков, основанного на квантовании

1. Исходные данные:

- две фиксированные очереди потоков разного приоритета с заданным временем выполнения,
- фиксированная величина кванта процессорного времени,
- количество процессоров - 1,
- циклическое выделение квантов потокам с учетом приоритета.

2. Результаты работы модели должны включать:

- среднее время выполнения потоков каждого приоритета.

Вариант 15

Модель алгоритма планирования потоков, основанного на квантовании

1. Исходные данные:

- фиксированная единая очередь потоков с заданным временем выполнения,
- фиксированная величина кванта процессорного времени,
- количество процессоров - 1,
- кратчайшая задача - первая.

2. Результаты работы модели должны включать:

- среднее время выполнения коротких и длинных потоков.

Вариант 16

Модель алгоритма планирования потоков, основанного на квантовании

1. Исходные данные:

- фиксированная единая очередь потоков с заданным временем выполнения,
- фиксированная величина кванта процессорного времени,
- количество процессоров - 2,
- циклическое выделение квантов потокам (круговое планирование).

2. Результаты работы модели должны включать:

- среднее время выполнения потоков.

Вариант 17

Модель алгоритма планирования потоков, основанного на квантовании.

1. Исходные данные:

- две фиксированных очереди потоков с заданным временем выполнения и разного приоритета,
- фиксированная величина кванта процессорного времени,
- количество процессоров - 2,
- первыми обрабатываются задачи высшего приоритета.

2. Результаты работы модели должны включать:

- среднее время выполнения потоков разного приоритета.

Вариант 18

Модель синхронизации потоков

1. Исходные данные:

- две программы, работающие с файлом в режиме разделения,
- первая программа записывает в файл произвольную информацию (по выбору исполнителя) в определенном количестве записей, после чего разрешает доступ к файлу второй программе;
- вторая программа удаляет записи, после чего разрешает доступ к файлу первой программе.

2. Результаты работы модели должны включать:

- печать содержимого файла обеими программами с временными отметками.

Вариант 19

Модель графа ресурсов и процессов

1. Исходные данные:

- в системе имеется N типов единичных ресурсов ($N \leq 10$),
- количество процессов, претендующих на ресурсы, - M ($M \leq 10$),
- исходное состояние характеризуется некоторым распределением и запросами на ресурсы.

2. Результаты работы модели должны включать:

- граф текущего состояния ресурсов и процессов,
- выделение цикла в графе, если он есть.

Вариант 20

Модель обнаружения блокировок при наличии нескольких ресурсов каждого типа

1. Исходные данные:

- в системе имеется M типов разделяемых ресурсов ($M \leq 10$),
- количество процессов, претендующих на ресурсы, N ($N \leq 10$),
- исходное состояние характеризуется некоторым распределением и запросами на ресурсы.

2. Результаты работы модели должны включать:

- матрицу текущего распределения ресурсов,
- матрицу текущих запросов процессов на ресурсы,
- решение для текущего состояния (есть тупик или нет, запускать новый процесс или нет).

Вариант 21

Модель стека

1. Исходные данные:

- стек списковой структуры,
- перечень операций со стеком: создание, включение элемента, выборка элемента, извлечение данных, уничтожение.

2. Результаты работы модели должны включать:

- меню с перечнем всех операций над стеком,
- печать содержимого стека.

решаемой задачи в структурной организации операционной системы.

Вариант 22

Модель стека

1. Исходные данные:

- стек векторной структуры,
- перечень операций со стеком: создание, включение элемента, выборка элемента, извлечение данных, уничтожение.

2. Результаты работы модели должны включать:

- меню с перечнем всех операций над стеком,
- печать содержимого стека.

Вариант 23

Модель очереди

1. Исходные данные:

- очередь векторной структуры,
- перечень операций с очередью: создание и освобождение, включение в очередь нового элемента, выборка элемента из очереди,
- дисциплина очереди – FIFO (добавление в конец очереди, выборка из головы очереди).

2. Результаты работы модели должны включать:

- меню с перечнем всех операций над очередью,

- печать содержимого очереди.

Вариант 24

Модель очереди

1. Исходные данные:

- очередь списковой структуры,
- перечень операций с очередью: создание и освобождение, включение в очередь нового элемента, выборка элемента из очереди,
- дисциплина очереди – FIFO (добавление в конец очереди, выборка из головы очереди).

2. Результаты работы модели должны включать:

- меню с перечнем всех операций над очередью,
- печать содержимого очереди

Вариант 25

Модель очереди

1. Исходные данные:

- очередь векторной структуры с динамической памятью,
- перечень операций с очередью: создание и освобождение, включение в очередь нового элемента, выборка элемента из очереди,
- дисциплина очереди – FIFO (добавление в конец очереди, выборка из головы очереди).

2. Результаты работы модели должны включать:

- меню с перечнем всех операций над очередью,
- печать содержимого очереди.

Критерии оценки:

№	Количество баллов	Критерии оценивания
1	60 баллов	работа выполнена полностью; в логических рассуждениях и обосновании решения нет пробелов и ошибок; в решении нет математических ошибок (возможны некоторые неточности, описки, которая не является следствием незнания или непонимания учебного материала), т.е. правильно выполнено 86–100 %работы.
2	55 баллов	работа выполнена полностью, но обоснования шагов решения недостаточны (если умение обосновывать рассуждения не являлось специальным объектом проверки); допущены одна ошибка, или есть два – три недочёта в выкладках, рисунках, чертежах или графиках (если эти виды работ не являлись специальным объектом проверки), т.е. правильно выполнено 74 – 85 %работы.
3	36 баллов	ставится, если: допущено не более двух ошибок или более двух – трех недочетов в выкладках, чертежах или графиках, но обучающийся обладает обязательными умениями по проверяемой теме, т.е. правильно выполнено 60 – 73 %работы.