УТВЕРЖДАЮ

Зав. кафедрой СП ЮУрГУ

\_\_\_\_\_\_\_\_\_Л.Б. Соколинский

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_2022 г.

Фонд оценочных средств ООП «Инженерия информационных и интеллектуальных систем» по направлению 09.03.04 – Программная инженерия

Дисциплина «Администрирование и развертывание программных компонент систем искусственного интеллекта в ОС Linux»

| **№ КМ** | **Вид КМ** | **Наименование КМ** | **Оценочные средства** |
| --- | --- | --- | --- |
|  | Текущий контроль | Проверка практического задания 1 | Задание:  1. Вывести строку на экран. Перенаправить вывод в файл  2. Создать файл. Установить права доступа. Проверить установленые права  3. Написать сценарий, изменяющий права доступа для заданного файла. Имя файла и права сценарий должен получить как параметр  4. В определенном каталоге у Вас имеется некоторое количество подкаталогов с содержимым, а также, возможно, файлы. Необходимо написать сценарий, который:  Заархивирует все подкаталоги. Исходные подкаталоги удалить, должны остаться только архивы.  Удалит все файлы из каталога(включая только что созданные архивы), кроме 5 самых новых  5. Написать сценарий для нахождения корней квадратного уравнения. Если решений нет, вывести "Нет решений". У пользователя сценарий запрашивает коэффициенты a,b,c. Сценарий должен работать с вещественными числами  Для прохождения собеседования по практической работе обучающийся предоставляет отчет. В процессе собеседования обучающийся должен свободно ориентироваться в отчете, сопоставлять теорию и полученные практические результаты, отвечать на дополнительные вопросы.  Вопросы для подготовки:  1.Командная оболочка BASH  2. Переменные окружения  3. Основные команды оболочки  4. Права доступа к файлам  5. Основные утилиты Linux |
|  | Текущий контроль | Проверка практического задания 2 | Задание:  Написать сценарий, обеспечивающий мониторинг доступности хоста. Сценарий должен записать в файл время проверки, информацию о доступности, задержку.  Например:  Thu Nov 15 10:49:39 +05 2018 up 0.00081s  или  Thu Nov 15 10:49:39 +05 2018 down  Проверьте работоспособность сценария, добавив его в crontab, так, чтобы сценарий запускался каждые 5 минут, но только в рабочее время с понедельника по пятницу.  Для прохождения собеседования по практической работе обучающийся предоставляет отчет. В процессе собеседования обучающийся должен свободно ориентироваться в отчете, сопоставлять теорию и полученные практические результаты, отвечать на дополнительные вопросы.  Вопросы для подготовки:  1. Работа с текстовыми файлами  2. Перенаправление ввода-вывода  3. Работа с Cron  4. Сетевые утилиты Linux  5. Утилиты для работы со строками |
|  | Текущий контроль | Проверка практического задания 3 | Задание:  RAID:  Размонтируйте разделы, находящиеся на дисках vdb,vdc,vdd,vde:  df -h посмотреть смонтированные разделы  umount /dev/xxx - размонтировать  либо выполните перезагрузку  1. Создаем RAID1:  Установите mdadm:  yum install mdadm  Удалите таблицы разделов на четырех дисках:  dd if=/dev/zero of=/dev/vdx bs=512 count=1  Создайте RAID:  mdadm --create --verbose /dev/md0 --level=1 --raid-devices=2 /dev/vdb /dev/vdc  Установите еще одно подключение к виртуальной машине и наберите watch cat /proc/mdstat для отслеживания состояния RAID в реальном времени. Расположите окна Putty так, чтобы можно было наблюдать их одновременно.  Отформатируйте и смонтируйте /dev/md0.  Диски имеют размер 200 MiB. Какой размер у RAID?  Создайте файл:  dd if=/dev/urandom of=/test1/testfile bs=1048576 count=100  Пометьте один диск как сбойный:  mdadm /dev/md0 --fail /dev/vdc (вывод /proc/mdstat должен измениться)  Замените "сбойный" диск:  mdadm /dev/md0 --remove /dev/vdc  mdadm /dev/md0 --add /dev/vdd (наблюдайте процесс восстановления RAID через /proc/mdstat)  Размонтируйте и удалите RAID:  umount ...  mdadm -S /dev/md0  2. По аналогии создать RAID5  Работа с LVM  1. Удалите все RAID из предыдущей работы  2. Установите утилиты LVM  3. Инициализируйте /dev/vdb как Physical Volume:  fdisk /dev/vdb - удалить все разделы  dd if=/dev/zero of=/dev/vdb bs=512 count=100  pvcreate /dev/vdb  4. Создайте Volume Group:  vgcreate Ocelot /dev/vdb  Проверить:  vgdisplay -v  5. Создайте Logical Volume:  lvcreate -L 190M -n Toy Ocelot  vgdisplay -v  6. Отформатируйте и смонтируйте, создайте файл:  mkfs.ext3 /dev/Ocelot/Toy  mount /dev/Ocelot/Toy /test1  dd if=/dev/urandom of=/test1/testfile bs=1048576 count=150  7. Добавьте еще несколько Physical Volume, расширьте логический раздел  Вопросы для подготовки:  1. RAID массивы. Базовые уровни RAID  2. Утилиты для настройки RAID  3. RAID и файловые системы, мониторинг состояния RAID  4. Понятие LVM  5. Утилиты для настройки LVM |
|  | Текущий контроль | Проверка практического задания 4 | Задание:  Создайте двух пользователей, например user1 и user2. Пользователи должны принадлежать группе "work".  [linux]# groupadd work  [linux]# useradd user123 -g work  [linux]# passwd user123  Создайте следующие каталоги:  ReadOnly - создайте в этом каталоге несколько файлов и защитите каталог и вложенные файлы с помощью атрибута "immutable". (Пример: chattr +i file.txt)  ReadWork - все пользователи из группы "work" должны иметь право на чтение. Права настроить через расширенные ACL.  User1 - user1 имеет право на чтение и модификацию, user2 - только чтение. Для вновь создаваемых объектов права наследуются. Права настроить через расширенные ACL.  Настроить квоты раздела таким образом, чтобы для user1 было доступно 10 Мб, для user2 - 15 Мб с возможностью временно превысить квоту на 2 Мб.  [linux]# yum install quota  [linux]# umount /test1 --отмонтируем  [linux]# mount /dev/vdb1 /test1 -o usrquota --монтируем с квотами  [linux]# quotacheck /test1  [linux]# quotaon /test1  [linux]# edquota user123 --учимся работать с редактором vi (квоты измеряются в KiB)  [linux]# quotasync /test1 --на всякий случай  создать какие-нибудь файлы на разделе от имени этого пользователя  [linux]# quota user123 --посмотреть состояние квоты  попробовать превысить квоту от имени пользователя  Проверить работу всех произведенных настроек.  Полезные команды:  chattr, lsattr, setfacl, getfacl, umask, quota, quotacheck, quotaon, edquota.  Для прохождения собеседования по практической работе обучающийся предоставляет отчет. В процессе собеседования обучающийся должен свободно ориентироваться в отчете, сопоставлять теорию и полученные практические результаты, отвечать на дополнительные вопросы.  Вопросы для подготовки:  1. Дисковая квота  2. Утилиты для настройки дисковой квоты  3. ACL файловых систем  4. Расширенные ACL  5. Утилиты для работы с ACL  6. Дополнительные атрибуты файлов. |
|  | Текущий контроль | Проверка практического задания 5 | Задание:  Написать приложение, позволяющее работать с анкетами в режиме совместного доступа (содержимое анкеты придумать самостоятельно, например фамилия, год рождения, стаж работы и т.д.). Анкеты должны храниться в бинарном файле. Необходимо обеспечить следующий функционал:  Просмотр анкеты. Пользователю необходимо выбрать номер анкеты. После выбора содержимое анкеты выводится на экран для просмотра. После команды пользователя просмотр заканчивается. Необходимо обеспечить защиту от редактирования анкеты во время просмотра.  Редактирование анкеты. Пользователю необходимо выбрать номер анкеты. После выбора содержимое анкеты выводится на экран и предлагается ввести новые значения полей. После завершения ввода всех данных процесс редактирования завершается. Необходимо обеспечить эксклюзивный доступ к анкете во время ее редактирования.  Добавление новой анкеты.  Приложение должно обеспечивать корректную работу с анкетами при запуске нескольких экземпляров приложения, работающих с одним файлом.  Для прохождения собеседования по практической работе обучающийся предоставляет разработанный код приложения. Представленный код должен выполнять поставленную задачу, его оформление должно соответствовать общепринятым соглашениям. В процессе собеседования обучающийся должен свободно ориентироваться в коде программы, сопоставлять теорию и полученные практические результаты, отвечать на дополнительные вопросы.  Вопросы для подготовки:  1. Работа с файлами в операционной среде  2. Функции для работы с файлами  3. Совместный доступ к файлам  4. Частичная блокировка файла  5. Функции для частичной блокировки файла |
|  | Текущий контроль | Проверка практического задания 6 | Задание:  Имеется файл со сложной структурой. В файле хранятся несколько массивов неизвестной длины. Массивы состоят из ненулевых элементов. Каждый массив заканчивается нулем. В начале файла имеется служебный массив, показывающий смещение начала каждого массива с данными относительно начала файла  В примере первый массив начинается с 18 байта от начала файла, второй - с 42-го.  Необходимо:  1. Создать тестовый файл с описанной выше структурой.  2. Написать программу, которая выводит на экран количество массивов с данными, а затем выбранный пользователем массив. Использовать отображение файла на память.  Для прохождения собеседования по практической работе обучающийся предоставляет разработанный код приложения. Представленный код должен выполнять поставленную задачу, его оформление должно соответствовать общепринятым соглашениям. В процессе собеседования обучающийся должен свободно ориентироваться в коде программы, сопоставлять теорию и полученные практические результаты, отвечать на дополнительные вопросы.  Вопросы для подготовки:  1. Способы доступа к файлам  2. Отображаемые на память файлы  3. Функции для отображения файлов на память  4. Преимущества отображения файлов на память  5. Работа с указателями |
|  | Текущий контроль | Проверка практического задания 7 | Задание:  Написать приложение для вычисления суммы вида    Вычисление должно происходить в два потока.   Значение N должно выбираться пользователем. Функцию f(x) придумать и реализовать самостоятельно.  Для прохождения собеседования по практической работе обучающийся предоставляет разработанный код приложения. Представленный код должен выполнять поставленную задачу, его оформление должно соответствовать общепринятым соглашениям. В процессе собеседования обучающийся должен свободно ориентироваться в коде программы, сопоставлять теорию и полученные практические результаты, отвечать на дополнительные вопросы.  Вопросы для подготовки:  1. Процессы и потоки  2. Функции для создания и завершения потоков  3. Принципы создания многопоточного приложения  4. Базовые функции для работы с потоками  5. Системные объекты для синхронизации потоков |
|  | Текущий контроль | Проверка практического задания 8 | Задание:  Написать два приложения для симплексной передачи текстовых сообщений через именованный канал.  Первое приложение:  В цикле считывает с клавиатуры текстовые строки и отправляет в именованный канал.  Второе приложение:  Считывает строки из именованного канала и выводит на экран  Полезные функции:  mkfifo  unlink  open  close  read  write  gets  Для прохождения собеседования по практической работе обучающийся предоставляет разработанный код приложения. Представленный код должен выполнять поставленную задачу, его оформление должно соответствовать общепринятым соглашениям. В процессе собеседования обучающийся должен свободно ориентироваться в коде программы, сопоставлять теорию и полученные практические результаты, отвечать на дополнительные вопросы.  Вопросы для подготовки:  1. Совместная работа процессов  2. Функции для управление процессами  3. Именованные каналы  4. Функции для работы с именованными каналами  5. Утилиты оболочки для работы с именованными каналами |
|  | Текущий контроль | Тест | Примеры вопросов для теста:  Способ управления памятью, при котором каждый процесс имеет "личное" адресное пространство, которое практически не ограничено в размерах, называется:  a. Свопинг  b. Виртуальная память  c. Распределение памяти динамическими разделами  Linux использует:  a. Автономное существование файловых систем  b. Монтирование файловых систем  c. Максимум одну файловую систему  Семафоры служат для:  a. Синхронизации потоков  b. Диспетчеризации потоков  c. Передачи произвольных данных между процессами  Вопросы для подготовки к тесту:  1. Понятие операционной системы.  2. Краткая история эволюции вычислительных систем  3. Архитектурные особенности операционных систем  4. Мультипрограммирование.  5. Процессы и потоки  6. Синхронизация потоков  7. Управление памятью  8. Файловые системы |
|  | Текущий контроль | Тест | Примеры вопросов для теста:  Команда  mdadm --create /dev/md0 --level=1 --raid-devices=2 /dev/hde1 /dev/hdf2  Выберите один ответ:  a. Создаст два RAID массива первого уровня  b. Создаст RAID массив, где блоки четности равномерно распределяются по всем дискам  c. Создаст RAID массив, у которого 50% дискового пространства будет использовано для обеспечения отказоустойчивости  Для инициализации диска или раздела для использования с LVM необходимо использовать команду:  Выберите один ответ:  a. vgcreate  b. lvcreate  c. pvcreate  В Linux параметр vm.swappiness управляет:  Выберите один ответ:  a. Размером SWAP-памяти  b. "Агрессивностью" использования SWAP  c. Возможностью выделять памяти больше, чем реально есть в системе  Вопросы для подготовки к тесту:  1. Работа с дисками. Таблица разделов. Утилиты  2. RAID массивы. Уровни RAID. Утилиты для работы с RAID.  3. LVM. Утилиты для работы с LVM. Снимки  4. Особенности кеширования в Linux. Параметры для управления кешированием  5. Виртуальная память в Linux. Параметры для управления виртуальной памятью.  6. ACL. Дополнительные атрибуты файлов |
|  | Промежуточный контроль | Тест | Вопросы для подготовки к экзамену по дисциплине:  1. Понятие операционной системы.  2. Краткая история эволюции вычислительных систем  3. Архитектурные особенности операционных систем  4. Мультипрограммирование.  5. Процессы и потоки  6. Синхронизация потоков  7. Управление памятью  8. Файловые системы  9. Работа с файлами  10. Полная и частичная блокировка файлов  11. Отображение файлов на память  12. Работа с потоками  13. Синхронизация потоков  14. Работа с дисками. Таблица разделов. Утилиты  15. RAID массивы. Уровни RAID. Утилиты для работы с RAID.  16. LVM. Утилиты для работы с LVM. Снимки  17. Особенности кеширования в Linux. Параметры для управления кешированием  18. Виртуальная память в Linux. Параметры для управления виртуальной памятью.  19. ACL. Дополнительные атрибуты файлов |

Паспорт фонда оценочных средств приведен в п. 6.3 РПД.

Разработчик С.А. Скрипов

ФГАОУ ВО «Южно-Уральский государственный университет   
(национальный исследовательский университет)»

Кафедра системного программирования

Дисциплина «Администрирование и развертывание программных компонент систем искусственного интеллекта в ОС Linux»

ИТОГОВЫЙ ТЕСТ

| № | Вопрос | Варианты ответа |
| --- | --- | --- |
|  | Журналирование файловой системы позволяет: | a.Сохранить целостность файловой системы  b.Сохранить данные, находящиеся в кэше при отключении питания  c.Узнать, кто и когда создавал, удалял и редактировал файлы |
|  | Монтирование предполагает: | a.Создание специальных файлов, ассоциированных с устройствами ввода-вывода  b.Объединение нескольких файловых систем в единое дерево  c.Удаление всех файлов и создание новой файловой системы |
|  | Операционная оболочка (в отличие от операционной среды и операционной системы) определяет: | a. Интерфейс прикладного программирования  b. Интерфейс пользователя  c. Особенности управления ресурсами системы |
|  | Виртуальная память позволяет: | a. Обеспечить предсказуемость времени доступа к памяти  b. Обойтись без аппаратной поддержки при управлении памятью  c. Защитить процессы друг от друга |
|  | Операционная система выполняет следующую функцию: | a. Управление процессами  b. Управление памятью  c. Управление внешними устройствами  d. Все перечисленное |
|  | Выберите правильное утверждение: | a. Процесс не может иметь более одного потока  b. В настоящее время процесс и поток - это одно и то же  c. Несколько потоков могут принадлежать одному процессу |
|  | Критическая секция это: | a. Участок исполнямого кода  b. Данные  c. Конструктивная часть процессора |
|  | Следующий фрагмент программы:  int k = 186500;  FILE \*f = fopen("file1.txt", "w+");  fprintf(f, "%d", k);  запишет в файл: | a. 8 байт  b. 4 байта  c. 6 байт  d. 2 байта |
|  | В Linux параметр vm.swappiness управляет: | a. Размером SWAP-памяти  b. "Агрессивностью" использования SWAP  c. Возможностью выделять памяти больше, чем реально есть в системе |
|  | Особенностью планировщика CFQ является: | a. Используется простая очередь FIFO, запросы исполняются без переупорядочивания  b. Каждому процессу присваивается собственная очередь  c. Используются три очереди: одна отсортированная и две неотсортированных |
|  | Основная цель форматирования раздела, это: | a. Исправление ошибок файловой системы  b. Очистка раздела и информации о нем в MBR  c. Создание структуры файловой системы  d. Запись информации о разделе в MBR |
|  | Недостатком RAID0 является: | a. Высокая вероятность потери данных  b. Низкая скорость записи  c. Низкая скорость чтения |
|  | Что из этого является файловой системой? | a. CHS  b. GPT  c. MBR  d. XFS  e. LBA |
|  | Что из этого не является файловой системой? | a. Reiser4  b. Btrfs  c. NTFS  d. EXT2  e. LBA |
|  | На разделе настроены квоты. Пользователь превысил "мягкое" ограничение (soft limit). | a. Пользователь не может читать и изменять файлы, но после окончания периода отсрочки доступ будет разблокирован  b. Пользователь может полноценно пользоваться файловой системой до окончания периода отсрочки  c. Новые файлы, после создания которых произошло превышение квоты, будут автоматически удалены  d. Пользователь может читать содержимое файлов, но запись недоступна |
|  | LVM это: | a. Подсистема, управляющая виртуальной памятью  b. Виртуальный диск, который использует для хранения данных раздел жесткого диска и специально выделенные страницы оперативной памяти  c. Дополнительный слой абстракции, позволяющий собрать несколько разнородных дисков в один, и затем снова разбить его так, как необходимо |
|  | Компьютер использует BIOS, EFI не поддерживается. Какие стандарты таблиц разделов можно использовать? | a. MBR  b. MBR и GPT  c. GPT |
|  | pdflash выполняет следующую задачу: | a. Перемещение данных между swap и физической памятью  b. Поддержка Memory Overcommit  c. Запись "грязных" данных на жесткий диск |
|  | Сколько байт в одном мебибайте? | a. 1000000000  b. 1048576  c. 1000000  d. 1073741824 |
|  | Одним из принципов работы OOMKiller является: | a. Завершить максимальное количество процессов  b. Завершить минимальное количество процессов  c. В первую очередь завершаются процессы, потребляющие мало памяти |