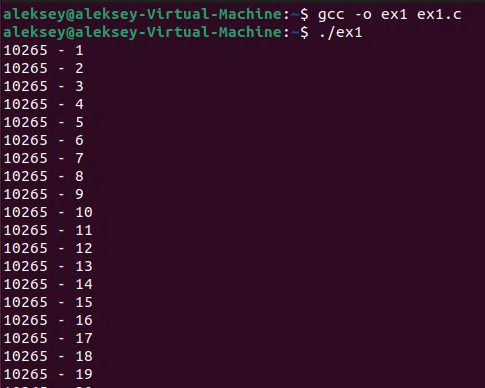
**Задание 01. Команды (утилиты) Linux**

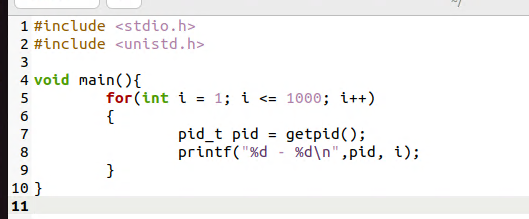
|  |  |
| --- | --- |
| Команда Linux | Краткое описание команды |
| **echo** | вывод данных на консоль |
| **ls** | Содержимое текущего директория |
| pwd | Путь к текущему директорию |
| cd | Переход в другую директорию |
| mkdir | Создания нового директория |
| rmdir | Удаления директория |
| touch | Настройка файлов |
| cp | Копирование файлов и/или директорий |
| mv | Эта утилита позволяет переименовать директории и отдельные объекты, а также перемещать их. |
| rm | Удаление файлов и директориев |
| su | Позволяет зайти в систему под другим пользователем, не завершая текущий процесс |
| whoami | Выводит имя текущего пользователя |
| man | Доступ к справочным материалам |
| whereis | используется для определения местоположения некоторых специальных файлов команды, таких как двоичный файл, исходный файл и файлы страниц руководства. |
| whatis | Показывает одностраничное описание справочной страницы |
| apropos | Поиск в именах справочных страниц и краткое содержание |
| cat | Выводит текст/файлы в консоль |
| less | Просмотр содержимого файлов текущего директория |
| head | Выводит n-ое количество строк, начиная отсчёт с первой строки. По умолчанию первых 10 |
| tail | Выводит строки с конца |
| lpstat | Вывод информации про принтеры |
| lpr | Печать файлов |
| lpq | Просмотр очереди печати |
| lprm | Печать документа |
| chgrp | Изменяет группы, но только те, членами которых они являются |
| chown | Изменяет владельца/группу для указанных файлов |
| chmod | Изменяет права доступа к файлам и каталогам |
| zip | Архивирование |
| gzip | Архивирование/сжатие файов |
| gunzip | Распаковка файлов gzip == gzip -d |
| bzip2 | Алгоритм сжатия файлов, используется в связке с tar-ом |
| bunzip2 | Команда **bunzip2** является ничем иным, как символьной ссылкой на утилиту bzip2 с параметром -d |
| tar | Встроенный архиватор |
| locate | Поиск файлов, расположенных на текущем пк |
| grep | С её помощью можно искать не только строки в файлах, но и фильтровать вывод команд, и много чего ещё. |
| find | Поиск файлов по разным параметрам (название, дата создания…) |
| history | История использованных команд |
| alias | Переменная (ярлык), содержащая исполняемую команду |
| unalias | Удаляет псевдонимы из системы |
| ps | Просмотр списка процессов |
| top | В отличии от ps, выводит информацию про систему и процессы, динамически обновляясь |
| lsof | Информация про файлы, открытых процессом |
| free | Команда **free** предоставляет информацию об использованной и неиспользованной памяти, а так же о разделе подкачки (swap). |
| df | показывает список всех файловых систем по именам устройств, сообщает их размер, занятое и свободное пространство и точки монтирования. |
| du | По умолчанию показывает размер файлового пространства, занимаемого каждым файлом и каталогом в текущем каталоге. Чтобы указать другой путь для работы, необходимо поместить его первым параметром. |
| yum | [открытый](https://ru.wikipedia.org/wiki/%D0%9E%D1%82%D0%BA%D1%80%D1%8B%D1%82%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5) консольный [менеджер пакетов](https://ru.wikipedia.org/wiki/%D0%9F%D0%B0%D0%BA%D0%B5%D1%82%D0%BD%D1%8B%D0%B9_%D0%BC%D0%B5%D0%BD%D0%B5%D0%B4%D0%B6%D0%B5%D1%80) для [дистрибутивов Linux](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D1%81%D1%82%D1%80%D0%B8%D0%B1%D1%83%D1%82%D0%B8%D0%B2_Linux), основанных на пакетах формата [RPM](https://ru.wikipedia.org/wiki/RPM). YUM позволяет системным администраторам и пользователям настроить автоматизированные обновление [ПО](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5) и разрешение зависимостей. |
| ifconfig | Используется для назначения сетевого адреса, изменение настроек параметров сетевого адаптера и IP протокола. |
| ping | Этот очень простой инструмент можно применять для тестирования неполадок сети и их исправления. |
| traceroute | Команда traceroute linux использует UDP пакеты. Она отправляет пакет с TTL=1 и смотрит адрес ответившего узла, дальше TTL=2, TTL=3 и так пока не достигнет цели. Каждый раз отправляется по три пакета и для каждого из них измеряется время прохождения. Пакет отправляется на случайный порт, который, скорее всего, не занят. Когда утилита traceroute получает сообщение от целевого узла о том, что порт недоступен трассировка считается завершенной. |
| host | Настройки dns-сервера |
| iwconfig | Команда **iwconfig** используется для настройки беспроводных сетевых интерфейсов |
| dhclient | Настройка dhcp |
| ifup | Команда **ifup** используется для активации (up) сетевого интерфейса |
| ifdown | ifdown – для его деактивации (down) |
| route | Когда компьютеру нужно отправить пакет в сеть он смотрит таблицу маршрутизации, в ней указанны ip адреса пунктов назначения и адреса интерфейсов и роутеров в домашней сети, которые могут отправить пакет по нужному адресу. Утилита распечатывает таблицу маршрутизации. |
| ssh | это основной протокол для удаленного управления серверами на базе операционной системы **Linux** |
| sftp | **SFTP** (**SSH** **File** **Transfer** **Protocol**) - это безопасный файловый протокол, который используется для доступа, управления и передачи файлов через зашифрованный транспорт **SSH**. |
| scp | Передача файлов на удалённый сервер |
| rsync | это наиболее часто используемая команда для удаленного и локального копирования и синхронизации файлов и каталогов в системах **Linux**/Unix. |
| wget | Команда wget linux, может выполнять загрузку файлов даже в фоновом режиме - без участия пользователя, в отличии от большинства веб браузеров. Кроме скачивания файлов, есть возможность сохранять веб страницы или даже целые веб-сайты, благодаря функции открытия ссылок на страницах. Такую возможность еще называют рекурсивной загрузкой. |
| curl | На самом деле, curl - это больше чем просто утилита командной строки для Linux или Windows. Это набор библиотек, в которых реализуются базовые возможности работы с URL страницами и передачи файлов. Библиотека поддерживает работу с протоколами: FTP, FTPS, HTTP, HTTPS, TFTP, SCP, SFTP, Telnet, DICT, LDAP, а также POP3, IMAP и SMTP. Она отлично подходит для имитации действий пользователя на страницах и других операций с URL адресами. |

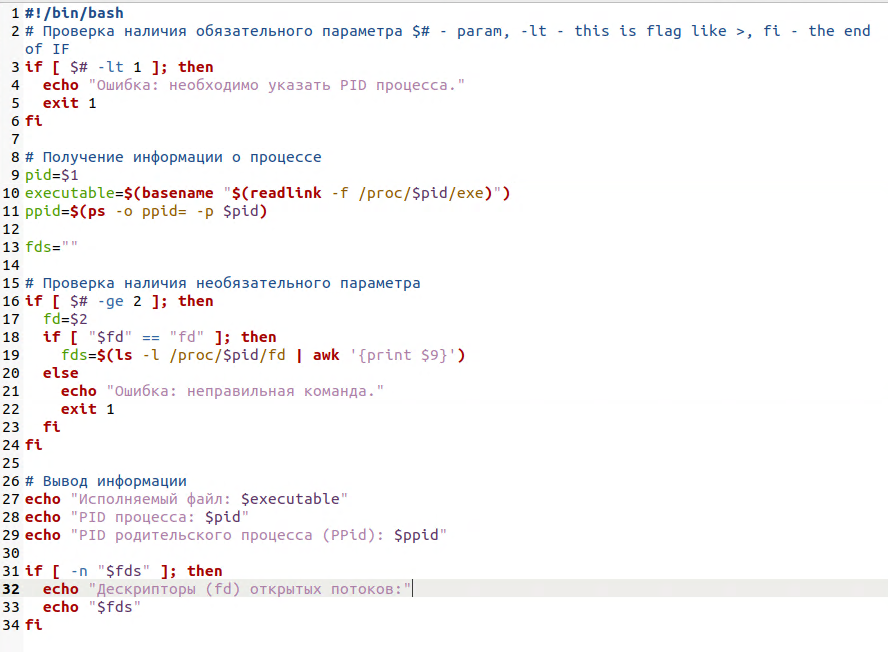
**Задание 02.Переменные окружения среды Linux**

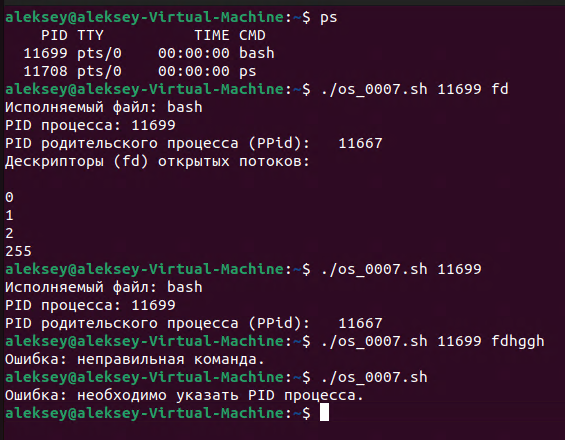
|  |  |
| --- | --- |
| Переменная окружения | Краткое описание |
| $HOME | исходный каталог |
| $PATH | Пути, занесённые в переменные среды |
| $PS1 | Строка приглашения |
| $PS2 | The value of this parameter is expanded as with PS1 and used as  the secondary prompt string. The default is ``> ''. |
| $IFS | Разделитель |

**Задание 04.Разработка С-приложения и bash-скрипт**

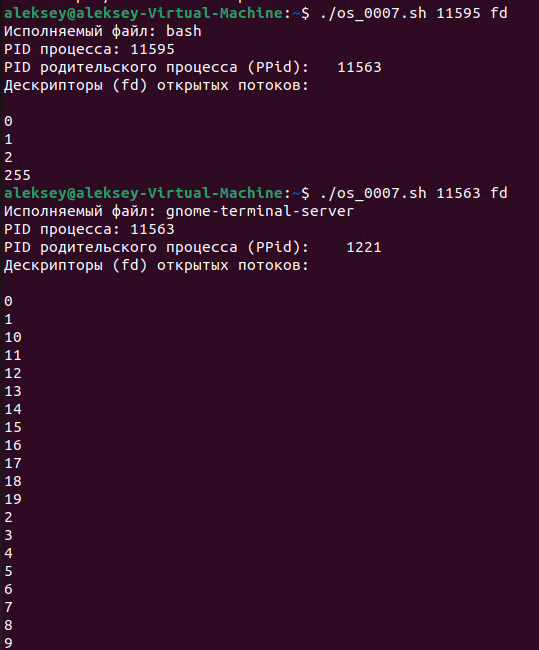


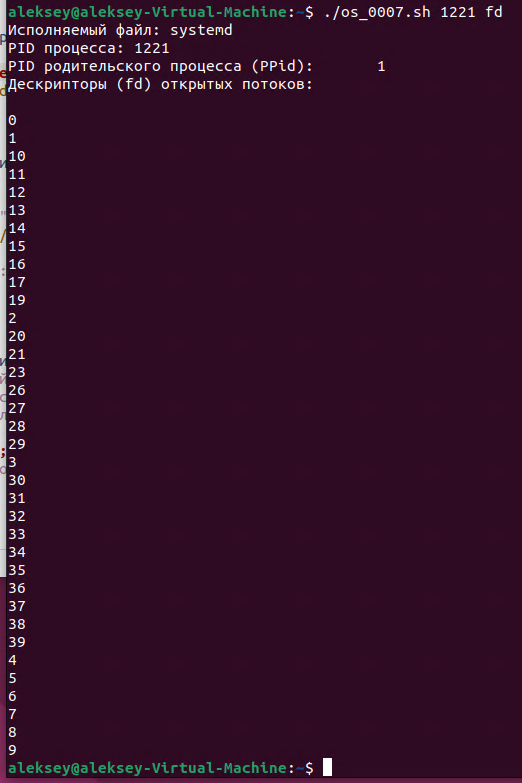






**Задание 05.Построение цепочки родительских процессов**





**Задание 06.ответьте на следующие вопросы**

1. Что такое фреймворк OS?

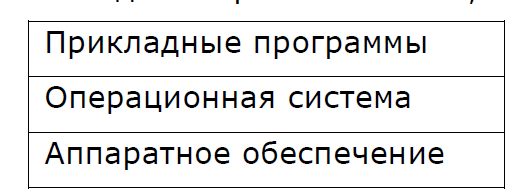
Фреймворк - это набор функций и порядок(правила) их использования (Смелов).

**Операционная система** — это комплект программ, которые служат интерфейсом между

модулями вычислительных систем и прикладными программными приложениями, а также

управляют компьютерным оборудованием и процессами вычислений, эффективным

распределением вычислительных мощностей среди процессов вычислений.



1. Что такое POSIX?

**POSIX** (англ. Portable Operating System Interface — переносимый интерфейс

операционных систем) — набор стандартов, описывающих интерфейсы между

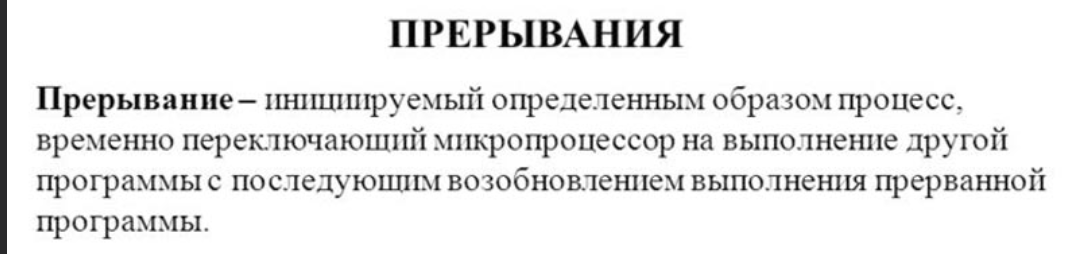
операционной системой и прикладной программой (системный API), библиотеку языка C

и набор приложений и их интерфейсов. Стандарт создан для обеспечения совместимости

различных UNIX-подобных операционных систем и переносимости прикладных программ

на уровне исходного кода, но может быть использован и для не-Unix систем.

1. Что такое аппаратное прерывание?



Аппаратное прерывание вызвано каким-либо аппаратным устройством, таким как запрос на запуск ввода-вывода, аппаратный сбой или что-то подобное. Аппаратные прерывания были введены как способ избежать траты драгоценного времени процессора на циклы опроса в ожидании внешних событий.

Например, когда завершена операция ввода-вывода, такая как считывание некоторых данных в компьютер с ленточного накопителя.

* Прерывание, генерируемое мышью при нажатии кнопки
* Прерывание, генерируемое сетевой картой при приеме данных
* Прерывание, генерируемое дисководом при завершении операции чтения или записи

1. Что такое программное прерывание?

Программные прерывания:

- Программные прерывания генерируются самой программой или операционной системой.

- Они используются для передачи управления от одной части программы к другой или для запроса обслуживания операционной системы.

- Обработка программных прерываний осуществляется программным уровнем, обычно через специальные прерывания операционной системы или среды выполнения.

- Программные прерывания могут быть инициированы программистом для обработки определенных условий или событий, и они обычно имеют более высокий приоритет, чем аппаратные прерывания.

В общем, аппаратные прерывания связаны с внешними устройствами и сигнализируют о событиях, требующих вмешательства центрального процессора. С другой стороны, программные прерывания используются для передачи управления программно и обеспечения синхронизации и взаимодействия различных частей программы или операционной системы.

1. Что такое системный вызов?

Обращение прикладной программы к ядру ОС для выполнения каких-либо операций

1. Что такое процесс OS?

**Процесс** - это экземпляр программы, которая выполняется на компьютере. Он представляет собой набор инструкций, данных и состояния, необходимых для выполнения определенной задачи. Каждый процесс имеет свой уникальный идентификатор (PID) и может включать в себя один или несколько потоков выполнения.

1. Что такое контекст процесса OS?

Контекст процесса в операционной системе (OS) - это состояние процесса, которое включает информацию о его текущем выполнении, такую как значения регистров процессора, указатели на стек и данные, открытые файлы и другие ресурсы, а также другие атрибуты, необходимые для возобновления выполнения процесса.

1. Что такое адресное пространство процесса?

Адресное пространство процесса - это набор адресов, которые процесс может использовать для доступа к памяти. Каждый процесс в операционной системе имеет свое собственное адресное пространство, которое обеспечивает изоляцию и защиту процессов друг от друга. Адресное пространство процесса обычно подразделяется на несколько областей с разными назначениями.

1. Перечислите области памяти адресного пространства процесса и поясните их назначение.

- Сегмент кода (Code segment): Эта область содержит исполняемый код программы.

- Сегмент данных (Data segment): Здесь хранятся глобальные и статические переменные программы.

- Сегмент стека (Stack segment): В этой области размещаются локальные переменные и данные, относящиеся к текущим вызовам функций.

- Сегмент кучи (Heap segment): Здесь располагаются динамически выделяемые данные, такие как объекты, массивы и структуры данных.

1. Что такое стандартные потоки процесса?

**Поток (или нить выполнения)** - это последовательность инструкций, которые выполняются в рамках процесса. Потоки являются более легковесными, чем процессы, поскольку они используют общие ресурсы процесса, такие как память и файловые дескрипторы. Внутри одного процесса может быть создано несколько потоков, которые работают параллельно или конкурируют за ресурсы процесса. Каждый поток имеет свой собственный стек выполнения, указатель команд и набор регистров.

Стандартные потоки процесса - это три основных потока ввода-вывода, связанных с каждым процессом: стандартный ввод (stdin), стандартный вывод (stdout) и стандартный вывод ошибок (stderr). Они представляют собой каналы, через которые процесс взаимодействует с окружающей средой. Например, стандартный ввод может использоваться для чтения пользовательского ввода, а стандартный вывод и вывод ошибок - для вывода результатов и сообщений.

1. Перечислите системные вызовы Windows для создания процесса?

Некоторые системные вызовы Windows для создания процесса:

- CreateProcess: Создает новый процесс и его первичный поток выполнения.

- CreateProcessAsUser: Создает новый процесс от имени указанного пользователя.

- CreateProcessWithLogonW: Создает новый процесс с использованием имени пользователя и пароля для аутентификации.

1. Перечислите системные вызовы Linux для создания процесса?

Некоторые системные вызовы Linux для создания процесса:

- fork: Создает новый процесс путем дублирования вызывающего процесса. Новый процесс называется "потомком", а исходный процесс - "родителем".

- exec: Заменяет текущий процесс новым процессом, загружая и выполняя новую программу из исполняемого файла.

- clone: Создает новый поток или процесс с заданными параметрами. Он предоставляет более гибкий контроль над создаваемыми процессами и потоками.

1. С помощью каких утилит можно увидеть перечень процессов в Windows?

- Диспетчер задач (Task Manager): Встроенная утилита, которая позволяет просматривать и управлять запущенными процессами, мониторить системные ресурсы и многое другое.

- PowerShell: Командлеты Get-Process и Tasklist позволяют получить информацию о запущенных процессах.

1. С помощью каких утилит можно увидеть перечень процессов в Linux?

В Linux для просмотра перечня процессов можно использовать следующие утилиты:

- ps: Команда ps отображает текущие процессы в виде списка.

- top: Утилита top позволяет просматривать список процессов в реальном времени и мониторить использование системных ресурсов.

- htop: Альтернативная утилита для просмотра процессов с интерактивным интерфейсом.

1. Перечислите свойства процесса OS.

Свойства процесса в операционной системе могут включать:

- Идентификатор процесса (PID): Уникальный числовой идентификатор, который идентифицирует процесс в системе.

- Состояние процесса: Текущее состояние процесса, например, запущен, приостановлен, ожидает и т. д.

- Приоритет процесса: Значение, определяющее относительную важность процесса и его приоритет выполнения.

- Ресурсы процесса: Информация о выделенных ресурсах, таких как память, файловые дескрипторы, открытые файлы и сетевые соединения.

- Родительский процесс(ppid): Информация о процессе, который создал текущий процесс (родительский процесс).

- Потоки процесса: Список потоков выполнения, работающих в рамках процесса.

- Время выполнения: Информация о времени, затраченном процессом на выполнение задач.

- Используемые системные ресурсы: Информация о системных ресурсах, используемых процессом, таких как процессорное время, память и дисковое пространство.