
МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Государственное образовательное учреждение высшего профессионального образования
«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И
РАДИОЭЛЕКТРОНИКИ» (ТУСУР)

УТВЕРЖДАЮ

Заведующий кафедрой ЭМИС

_____ И. Г. Боровской

«___» _____ 2012 г.

Е.А. ПАНАСЕНКО

СИСТЕМНОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

*Методические указания по выполнению лабораторных работ
для студентов 230100 «Информатика и вычислительная техника»*

2012

Панасенко Е.А. Системное программное обеспечение: методическое пособие – Томск: Изд-во ТУСУР, 2012. – 11 с.

Методическое пособие для студентов ВУЗов посвящено изучению системного программного обеспечения. Основной акцент делается на изучение организации управления процессами на примере ОС Linux, которая является многопроцессной Unix-о подобной операционной системы (ОС). Данная ОС представляет собой систему с открытым кодом, что обеспечивает возможность использования ее в образовательных учреждениях.

СОДЕРЖАНИЕ ЛАБОРАТОРНЫХ ЗАНЯТИЙ
по дисциплине «Системное программное обеспечение»
для студентов 230100 «Информатика и вычислительная техника»

Краткое содержание тем и результатов их освоения.....	4
Лабораторная работа №1	5
Лабораторная работа №2	6
Лабораторная работа №3	9
Лабораторная работа №4	10
Лабораторная работа №5	10

Краткое содержание тем и результатов их освоения

Тема лабораторных занятий	Деятельность студента. Решая задачи, студент:
1. Установка Cygwin, его настройка	<ul style="list-style-type: none"> • <i>изучает</i> особенности установки Cygwin; • <i>получает навыки</i> установки и настройки Cygwin;
2. Среда программирования. Командные интерпретаторы. Переменные окружения.	<ul style="list-style-type: none"> • <i>изучает</i> понятия “среда программирования” и “переменные окружения”; • <i>получает навыки</i> работы с командными интерпретаторами;
3. Системные команды. Команды Linux.	<ul style="list-style-type: none"> • <i>знакомится</i> с основными системными командами; • <i>применяет на практике</i> команды Linux;
4. Разработка программ в среде Linux с помощью инструментария GCC.	<ul style="list-style-type: none"> • <i>получает навыки</i> программирования в среде Linux; • <i>создает</i> самостоятельно программу, определяющую, например, наибольший общий делитель;
5. Генерация документации по разрабатываемому программному обеспечению с использованием пакетов Graphviz и DoxyGen.	<ul style="list-style-type: none"> • <i>знакомится</i> с пакетами Graphviz и DoxyGen; • <i>получает навыки</i> генерации документации для программного обеспечения;

ХОД ПРАКТИЧЕСКИХ РАБОТ

1. Ознакомиться со справочными интернет-сведениями (СРС)
2. Ознакомиться с указанной темой в основной и дополнительной литературе.

Синицин С. В., Налютин Н. Ю., Батаев А.В. Операционные системы. – М.: Академия, 2010.

Дополнительная литература

Гусятников В. Н., Безруков А. И. Стандартизация и разработка программных систем. – Издательство: Финансы и статистика, 2010

Таненбаум Э. С. Современные операционные системы, Питер, 2011

3. Ознакомиться с принципом выполнения лабораторных работ.
4. Составить и предоставить преподавателю отчет о работе, если он входит в форму отчетности по данному разделу знаний.

Введение

В настоящем пособии описаны лабораторные работы по дисциплине «Системное программное обеспечение». Целью проведения лабораторных работ является закрепление основных теоретических положений, изложенных в лекциях на примере широко используемых в различных областях ОС UNIX (Linux). В процессе выполнения лабораторных работ студенты должны получить практические навыки работы в ОС Linux.

Лабораторная работа №1

Установка Cygwin, его настройка (2 часа)

Цель работы: научиться устанавливать и настраивать Cygwin.

Cygwin — UNIX-подобная среда и интерфейс командной строки для Microsoft Windows. Cygwin обеспечивает тесную интеграцию Windows приложений, данных и ресурсов с приложениями, данными и ресурсами UNIX-подобной среды. Из среды Cygwin можно запускать Windows приложения, также можно использовать инструменты Cygwin из Windows.

Cygwin состоит из двух частей: динамически подключаемая библиотека (DLL) `cygwin1.dll`, которая обеспечивает совместимость API и реализует значительную часть стандарта POSIX и огромная коллекция приложений, которые обеспечивают привычную среду UNIX.

Изначально Cygwin был разработан компанией Cygnus Solutions, которую позднее приобрела Red Hat. Это свободное ПО, опубликованное под GNU General Public License версии 2. В настоящее время Cygwin разрабатывается сотрудниками Red Hat, NetApp и множеством добровольцев.

Подобные функциональные возможности предлагала также и Microsoft в своём пакете Services for UNIX, включающем в себя подсистему Interix, и *Subsystem for UNIX-based Applications* (в более новых версиях).

Cygwin представляет собой инструмент для портирования ПО UNIX в Windows и представляет собой библиотеку, которая реализует интерфейс прикладного программирования POSIX на основе системных вызовов Win32. Кроме того, Cygwin включает в себя инструменты разработки GNU для выполнения основных задач программирования, а также и некоторые прикладные программы, эквивалентные базовым программам UNIX. В 2001 году в Cygwin был включён пакет X Window System.

Кроме того, Cygwin содержит библиотеку MinGW, позволяющую работать с библиотекой Microsoft MSVCRT (Windows API); библиотека MinGW менее требовательна к объёму оперативной и дисковой памяти, распространяется под более свободной лицензией и может работать с любым программным обеспечением, но

функциональные возможности спецификации POSIX реализованы в ней не так полно, как в Cygwin.

Red Hat реализует библиотеку Cygwin под лицензией GNU GPL, таким образом, её можно использовать только в программах, распространяемых под GPL. (Для распространения программ, использующих библиотеку Cygwin, под лицензией, отличной от GPL, необходимо приобретение лицензии у RedHat).

На странице Cygwin Mailing Lists можно подписаться на рассылку, посвящённую Cygwin.

Работа с кириллицей. В современных версиях Cygwin работа с русскими символами не требует дополнительных настроек. Символы кириллицы работают в нейтральной локали LANG=C.UTF-8, в том числе имена файлов, имена пользователей и данные реестра. Можно установить русскоязычную локаль LANG=ru_RU.UTF-8, при этом включатся русскоязычные сообщения (у некоторых программ), даты по русски (LC_TIME) и другие категории локали.

В старых версиях, для того, чтобы видеть и вводить русские символы при работе в cygwin, необходимо установить переменную среды окружения LANG=ru_RU.CP1251 и создать файл ~/.inputrc или /etc/inputrc, который использует библиотека GNU readline следующего содержания:

```
set input-meta on
set meta-flag on
set convert-meta off
set output-meta on
```

Кроме этого желательно определить (например в файле ~/.bashrc) алиас

```
alias ls='ls --show-control-chars'.
```

Задание. Запустить и настроить Cygwin.

Лабораторная работа №2

Среда программирования. Командные интерпретаторы. Переменные окружения (2 часа)

Цель работы: ознакомиться с командными интерпретаторами Linux. Научиться работать с переменными окружения.

Переменная окружения в Windows используются для хранения текстовых строк пользователя и информации о настройках операционных систем.

В современных ОС компании Microsoft на платформе NT для получения информации о системе используются следующие переменные (**полужирным выделен текст, относящийся только к Windows Vista/7**):

Название	Тип	Описание
ALLUSERSPROFILE	локальная	Возвращает размещение профиля «All Users». В Windows 7 возвращает путь к папке C:\ProgramData\.
APPDATA	системная	Возвращает используемое по умолчанию размещение данных приложений. В Windows XP это C:\Documents and Settings\%UserName%\Application Data. В Windows 7 - C:\Users\%UserName%\AppData\Roaming.
CD	локальная	Указывает путь к текущему каталогу. Идентична команде CD без аргументов.
CLIENTNAME	локальная динамическая	Переменная определена только в среде удаленного подключения к рабочему

		столу. Хранит имя компьютера (%COMPUTERNAME%) клиента удаленного доступа.
CMDCMDLINE	локальная	Точная команда, использованная для запуска текущего cmd.exe.
CMDEXTVERSION	системная	Версия текущего Command Processor Extensions.
COMMONPROGRAMFILES	системная	Расположение каталога "Common Files" (обычно %ProgramFiles%\Common Files).
COMMONPROGRAMFILES(x86)	системная	Расположение каталога "Common Files" в Program Files (x86) для 64-разрядной ОС (обычно %ProgramFiles(x86)%\Common Files).
COMPUTERNAME	системная	Имя компьютера
COMSPEC	системная	Путь к исполняемому файлу командного процессора (shell).
DATE	системная	Возвращает текущую дату. Использует тот же формат, что и команда date /t. Создается командой Cmd.exe.
ERRORLEVEL	системная	Возвращает код ошибки последней использовавшейся команды. Значение, не равное нулю, обычно указывает на наличие ошибки.
HOMEDRIVE	системная	Возвращает имя диска локальной рабочей станции, связанного с основным каталогом пользователя. Задаётся на основании расположения основного каталога. Основной каталог пользователя указывается в оснастке «Локальные пользователи и группы».
HOMEPATH	системная	Возвращает полный путь к основному каталогу пользователя. Задаётся на основании расположения основного каталога. Основной каталог пользователя указывается в оснастке «Локальные пользователи и группы».
HOMESHARE	системная	Возвращает сетевой путь к общему основному каталогу пользователя. Задаётся на основании расположения основного каталога. Основной каталог пользователя указывается в оснастке «Локальные пользователи и группы».
LOCALAPPDATA	системная	Возвращает используемое по умолчанию локальное размещение данных приложений (C:\Users\%UserName%\AppData\Local).
LOGONSERVER	локальная	Имя контроллера домена, использовавшегося для авторизации

		текущего пользователя.
NUMBER_OF_PROCESSORS	системная	Количество процессоров в системе.
OS	системная	Название операционной системы. Windows XP и Windows 2000 отображаются как Windows_NT.
PATH	системная	Указывает путь поиска исполняемых файлов.
PATHEXT	системная	Возвращает список расширений файлов, которые рассматриваются операционной системой как исполняемые.
PROCESSOR_ARCHITECTURE	системная	Архитектура процессора. Возможные варианты: x86, IA64, AMD64.
PROCESSOR_IDENTIFIER	системная	Описание процессора (в Интернете часто встречается ошибочное написание PROCESSOR_IDENTFIER - пропущена буква).
PROCESSOR_LEVEL	системная	Номер модели процессора.
PROCESSOR_REVISION	системная	Ревизия процессора.
PROGRAMDATA	системная	Возвращает путь к каталогу C:\ProgramData\ (аналогично ALLUSERSPROFILE).
PROGRAMFILES	системная	Путь к каталогу Program Files.
PROGRAMFILES(x86)	системная	Путь к каталогу Program Files (x86) в 64-разрядных системах для приложений архитектуры x86.
PROMPT	локальная	Возвращает параметры командной строки для текущего интерпретатора. Создается командой Cmd.exe.
PUBLIC	системная	В Windows 7 возвращает путь к каталогу C:\Users\Public.
RANDOM	системная	Случайное десятичное число от 0 до 32767. Генерируется Cmd.exe.
SESSIONNAME	локальная	Хранит имя активного пользовательского сеанса. При локальном входе имеет значение "Console", при удаленном доступе имеет вид RDP-Тср#<номер сеанса>
SYSTEMDRIVE	системная	Диск, на котором расположен корневой каталог Windows.
SYSTEMROOT	системная	Путь к корневому каталогу Windows.
TEMP or TMP	системная и локальная	Возвращает временные каталоги, по умолчанию используемые приложениями, которые доступны пользователям, выполнившим вход в систему. Некоторые приложения требуют переменную TEMP, другие — переменную TMP. Потенциально TEMP и TMP могут указывать на разные каталоги, но обычно - совпадают.

TIME	системная	Возвращает текущее время. Использует тот же формат, что и команда time /t. Создается командой Cmd.exe. Дополнительные сведения о команде time см. в разделе Time.
USERDOMAIN	локальная	Имя домена, которому принадлежит текущий пользователь.
USERNAME	локальная	Имя текущего пользователя.
USERPROFILE	локальная	Путь к профилю текущего пользователя.
WINDIR	системная	Каталог, в котором установлена Windows.

Работа с переменными. Чтобы получить доступ к значению переменной, необходимо поставить знак % перед её названием и после него, например:

```
echo %PROCESSOR_ARCHITECTURE%
```

Чтобы установить значение переменной:

```
SET TEMP=C:\TEMP
```

Кроме того, с переменными среды можно работать, нажав кнопку «Переменные среды» на вкладке «Дополнительно» в диалоговом окне «Свойства системы» (контекстное меню значка «Мой компьютер» → пункт «Свойства»).

Также необходимо помнить, что, в отличие от UNIX-систем, в DOS и в стандартной командной строке Windows переменные среды не наследуются запускаемыми процессами, а используются совместно. Это позволяет, например, возвращать через них результаты работы программ, но, с другой стороны, создаёт возможность порчи или переполнения. Особенно актуально переполнение для переменной PATH, поскольку она часто дополняется при работе различных скриптов. При возникновении такого переполнения достаточно просто перезапустить интерпретатор командной строки. Интерпретаторы, портированные из UNIX-систем (например, Z-shell), такой проблеме не подвержены.

Получить значение переменной в программном коде можно с помощью метода ExpandEnvironmentStrings COM-объекта WScript.Shell:

```
WScriptShell = ObjCreate("WScript.Shell")
UserName = WScriptShell.ExpandEnvironmentStrings("%USERNAME%")
```

Следует учитывать, что набор значений переменных среды получается процессом во время запуска, все запускаемые подчиненные процессы получают значения переменных из этого набора, если другой независимый процесс изменяет значение переменной среды или создает новую переменную, эти изменения не будут доступны процессу до его перезапуска.

Задание. Применить приведенные переменные окружения на практике.

Лабораторная работа №3

Системные команды. Команды Linux (8 часов)

Цель работы: ознакомиться с основными командами Linux.

Подробное теоретическое обоснование данной темы представлено в следующих источниках:

1. Колисниченко Д. Самоучитель Linux openSUSE 11. - СПб. : БХВ-Петербург, 2009. – 419 с.
2. Таненбаум Э. С. Современные операционные системы, Питер, 2011

Задание. Выполнить следующие действия:

1. Запустить программу putty, ввести адрес сервера для подключения (Host name) 217.71.139.65, выбрать протокол подключения SSH (порт 22), в категории Window/Appearance изменить (Change) шрифт на Courier Bold 16-pixel, вернуться в категорию Session, ввести имя сохраняемых настроек Saved Session, сохранить настройки (Save), открыть соединение (Open).

2. Ввести имя пользователя user№, пароль user№, где № состоит из двух цифр - порядкового номера группы в микропотоке и номера бригады. Выполните команды help, ls, cd, pwd, mkdir, rmdir. Изучить описания этих команд с помощью инструкций man и info. Изучить описания команды script.

3. Создать текстовый файл с произвольным содержанием при помощи команды tee. Просмотреть содержимое файла при помощи команды cat и исправить в нем ошибки при помощи текстового редактора vi. Изучить справку на команды sr, unlink.

4. Сделать копию файла командой sr, удалить ее командой unlink, запротоколировать эти действия.

5. Создайте структуру каталогов в соответствии с вариантом. Черными линиями представлена вложенность файлов/подкаталогов в каталоги. Синими линиями представлены ссылки. Красными линиями - символические ссылки. Стрелка на красной линии указывает на целевой файл ссылки. Файлы создаются копированием ранее созданного файла командой sr с внесением в копии некоторых изменений. Ссылки создаются командой ln, символические ссылки - ей же, но с ключом -s:

Лабораторная работа №4

Разработка программ в среде Linux с помощью инструментария GCC (8 часов)

Цель работы: получение навыков работы программирования в среде Linux.

Подробное теоретическое обоснование данной темы представлено в следующих источниках:

1. Колисниченко Д. Самоучитель Linux openSUSE 11. - СПб. : БХВ-Петербург, 2009. – 419 с.
2. Таненбаум Э. С. Современные операционные системы, Питер, 2011.

Задание. Установить компилятор GNU C Compiler под UNIX/LINUX. Написать и отладить программу, выводющую на экран строку «Hello UNIX» с использованием стандартной библиотеки ввода-вывода и с использованием функции непосредственной записи в стандартный поток вывода.

Написать и отладить программу, запрашивающую 2 числа и выводющую их наибольший общий делитель. Использовать алгоритм Евклида.

Написать и отладить программу, принимающую из командной строки 2 числа. Если эти числа равны, вывести на экран только одно число, если нет – вывести числа в порядке возрастания.

Написать и отладить программу, определяющую является ли файл исполняемым файлом формата ELF. Программа должна запросить у пользователя имя файла. Если файл не удастся открыть, проанализировать ошибку и выдать соответствующее сообщение.

Лабораторная работа №5

Генерация документации по разрабатываемому программному обеспечению с использованием пакетов Graphviz и DoxyGen (8 часов)

Цель работы: получение навыков разработки документации программного обеспечения с использованием пакетов Graphviz и DoxyGen.

Doxygen - это утилита создания документации для программ, написанных на C++, C, Java, Objective-C, Python, IDL (Corba и Microsoft) и некоторых версиях PHP, C#, и D. Выходная документация представляет собой файлы в html-формате, в формате LaTeX (в частности, для дальнейшего создания pdf - файлов), rtf, справочного руководства в стиле Unix - man, справочного руководства в стиле кроссплатформенной библиотеки Qt, xml и так далее. Даже, если не оформлять специальным образом исходные тексты, Doxygen сгенерирует из них замечательные справочники программного интерфейса приложения. В выходной документ могут быть включены разделы с описанием используемых namespaces, классов, файлов и их указатели на соответствующие разделы. В случае использования пакета визуализации графов - Graphviz, описание классов может сопровождаться диаграммами.

Задание. Получить документацию на основе исходных текстов для разрабатываемого программного обеспечения из предыдущих лабораторных работ.