

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Косенок Сергей Михайлович
Должность: ректор
Дата подписания: 06.06.2024 06:44:13
Уникальный программный ключ: «Разработка программного обеспечения в ОС Linux»
e3a68f3eaa1e62674b54f4998099d3d6bfdcf836

Оценочные материалы для промежуточной аттестации по дисциплине

«Разработка программного обеспечения в ОС Linux»

Код, направление подготовки	01.03.02 Прикладная математика и информатика
Направленность (профиль)	Прикладная математика и информатика
Форма обучения	Очная
Кафедра-разработчик	Прикладной математики
Выпускающая кафедра	Прикладной математики

Типовые задания на контрольную работу

Вариант 1

- 1) Напишите примеры использования команд `cp` и `mv`
- 2) Напишите простейшую программу на языке программирования C, состоящую из двух исходных файлов. Оснастите один из файлов заголовочным файлом `.h`. Для которого из двух исходных файлов и в каком случае необходим заголовочный файл?

Вариант 2

- 1) Напишите примеры использования команд `mkdir` и `rm`
- 2) Напишите `Makefile`, позволяющий собрать программу, написанную на языке C и состоящую из двух исходных файлов. Предусмотрите в `Makefile` константы для задания компилятора и флагов компиляции

Вариант 3

- 1) Напишите консольные команды, с помощью которых можно клонировать проект из репозитория `git` в локальную папку. Напишите команды для выбора нужной ревизии, просмотра различий между ревизиями и возвращения на главную ветвь.
- 2) Напишите `Makefile`, позволяющий собрать программу, заархивировать проект и очистить проект от промежуточных (объектных) файлов. Предусмотрите отдельные цели (`target`) для сборки, архивации и очистки.

Вариант 4

- 1) Напишите консольные команды, с помощью которых можно клонировать проект из репозитория `SVN` в локальную папку. Напишите команды для выбора нужной ревизии, просмотра различий между ревизиями и возвращения на главную ветвь.
- 2) Напишите `Makefile`, позволяющий собрать программу, заархивировать проект и очистить проект от промежуточных (объектных) файлов. Предусмотрите отдельные цели (`target`) для сборки, архивации и очистки.

Вариант 5

- 1) Напишите примеры использования команд `cd`, `ls`, `touch`.
- 2) Опишите основные принципы использования отладчика `gdb`. Как запустить программу в отладчике? Как просмотреть маршрут, который привёл к ошибке?

Типовые вопросы к зачёту

1. Основы работы в Linux. Структура файловой системы. Права доступа. Работа в консоли.
2. Коллекция компиляторов GCC. Поддерживаемые языки. Основы использования на примерах программ на C и C++.
3. Сборка проектов, состоящих из нескольких файлов с исходным кодом. Сборка статических библиотек пользователя. Сборка динамических библиотек пользователя. Подключение пользовательских библиотек.
4. Автосборка проектов. Утилита make. Синтаксис make-файлов. Опции и параметры make. Комментарии в make-файле. Макросы в make-файле.
5. Make. Множественные задания. Встроенные правила. Суффиксы и шаблоны правил. Управление библиотеками с помощью make. Make-файлы и подкаталоги.
6. Системы контроля версий. Система Mercurial. Система Subversion. Система git.
7. Средства отладки программ. Отладка с помощью gdb. Запуск gdb. Выполнение программы. Трассировка стека. Просмотр переменных. Вывод листинга программы.
8. Gdb. Установка точек останова. Вставка исправлений с помощью отладчика. Дополнительные сведения о gdb. Дополнительные средства отладки. Lint. Средства, отслеживающие вызовы функций.
9. Профилирование с помощью prof/gprof. Проверки соблюдения условий.
10. Устранение ошибок использования памяти. ElectricFence. Valgrind.
11. Введение в систему X. X-сервер. X-клиент. X-протокол. Xlib. Комплекты инструментов. Оконные менеджеры. Введение в GTK+. Система типов Glib. Система объектов GTK+.
12. События, сигналы и обратные вызовы. Виджеты упаковочных контейнеров. Виджеты GTK+.
13. Виджеты GNOME. Меню GNOME. Диалоговые окна.
14. Qt. Сигналы и слоты.
15. Виджеты Qt. Диалоговые окна. Применение qmake для упрощения написания make-файлов.