

ПРИМЕНЕНИЕ FREE PASCAL ДЛЯ ПОДДЕРЖКИ КУРСА СИСТЕМНОГО ПРОГРАММИРОВАНИЯ В UNIX

А.П. Полищук^а, С.А. Семериков^б

г. Кривой Рог, Криворожский государственный педагогический университет

^а apol@cabletv.dp.ua

^б cc@kpi.dp.ua

Одной из составных частей системного программирования является совокупность программных средств интерфейса программиста с операционной системой. В этом классическом определении не учитывается то, что системные программы (утилиты) обязательно должны взаимодействовать и с пользователем, поэтому предполагается наличие минимального пользовательского интерфейса (текстовый режим, командная строка).

Интерфейс системных вызовов ОС UNIX отличается стабильностью (более 35 лет разработки), компактностью (менее 1000 базовых системных вызовов) и универсальностью (один системный вызов применим к различным ситуациям). В начале нынешнего десятилетия интерфейс системных вызовов ОС UNIX стал основой международного стандарта переносимого открытого программного обеспечения (POSIX).

Таким образом, освоение системного программирования в ОС UNIX дает возможность в дальнейшем создавать мобильное программное обеспечение под любую современную операционную систему, удовлетворяющую требования стандарта POSIX (Windows NT/2000/2003/XP, UNIX-подобные системы Linux, FreeBSD, Solaris, Mac OS X).

Традиционно в качестве языка системного программирования используется язык Си. Это, в частности, обусловлено тем, что становление языка Си и ОС UNIX проходило в одно и то же время в одном и том же авторском коллективе. Долгое время язык Си выгодно отличался от конкурирующих процедурных языков гибкостью, простотой, переносимостью и компактностью записи. В то же время на рынке образовательных услуг чаще использовался язык Паскаль, отличавшийся высокой степенью формализации и структурирования программ. Как язык для начального обучения программированию, он четко реализовал в своем синтаксисе основные алгоритмические конструкции, содержал в качестве ключевых слова английского языка, имел самоочевидные имена процедур и функций.

В процессе своего развития язык Си стал более четко структурированным, менее терпимым к свободной работе с указателями, более полно соответствующим парадигмам структурного программирования. В свою очередь, Паскаль приобрел ограниченную адресную арифметику, способность гибко управлять циклами, модульную структуру и другие позаимствованные у языка Си средства. Такое взаимное обогащение языков-конкурентов привело к тому, что постепенно оба они стали применяться и в системном, и

в прикладном программировании.

Наивысшая степень гибкости языка Паскаль, эквивалентная Си, но с сохранением традиционного Паскаль-синтаксиса, реализована в кроссплатформенном свободно распространяемом компиляторе Free Pascal.

Наличие общих программных библиотек для языков Си и Паскаль, реализованная в семействе компиляторов GCC и Free Pascal, позволяет решать любые задачи одинаковыми или хотя бы однотипными средствами, оставляя различие лишь в синтаксисе.

Это позволило нам поставить перед собой следующую цель: создать справочник системного программиста в UNIX на языке Паскаль.

Для реализации этой цели мы решили следующие задачи:

1) построили интерфейсную библиотеку к ядру операционной системы UNIX на языке Паскаль, расширяющую функциональность стандартных модулей linux и sysutils;

2) создали аналоги функций библиотеки stdio языка Си для доступа к файлам и процессам;

3) составили систематическое руководство программиста для модулей linux, sysutils и stdio;

4) проиллюстрировали руководство примерами системных утилит.

В результате было создано руководство системного программиста в UNIX в среде Free Pascal [1], которое включает такие разделы:

1. Основные понятия и терминология
2. Файл
3. Работа с файлами
4. Каталоги, файловые системы и специальные файлы
5. Процесс
6. Сигналы и их обработка
7. Межпроцессное взаимодействие при помощи программных каналов
8. Дополнительные методы межпроцессного взаимодействия
9. Терминал
10. Сокеты
11. Стандартная библиотека ввода/вывода
12. Разные дополнительные системные вызовы и библиотечные процедуры.

Особенностью выбранных средств реализации поставленных задач является их открытость, низкая стоимость и переносимость. Это позволяет разработанному программному обеспечению функционировать под управлением многих операционных систем.

Литература:

1. Полищук А.П., Семериков С.А. Системное программирование в UNIX средствами Free Pascal. – Кривой Рог, 2005. – 418 с. – www.sources.ru/pascal/docs/sysprfpc.htm