

УДК 004.451:004.94

**ПРОГРАММНЫЙ КОМПЛЕКС ДЛЯ ОЦЕНКИ ПРОИЗВОДИТЕЛЬНОСТИ
ОСНОВНЫХ ФУНКЦИЙ ОПЕРАЦИОННЫХ СИСТЕМ****Горбанёва И.В., Мартышкин А.И.***Пензенский государственный технологический университет, Пенза, e-mail: Alexey314@yandex.ru*

В статье описан созданный программный комплекс для оценки производительности функций операционных систем. Целью работы является создание комплекса программ, необходимого для оценки времени выполнения программных функций однопроцессорных и многопроцессорных операционных систем. Комплекс выполняет анализ производительности ряда функций операционных систем: мьютексов, семафоров, блокировок чтения-записи, каналов FIFO и PIPE, сокетов TCP и UDP, переключения контекста, системных вызовов (чтение, запись, открытие, закрытие). В отличие от существующих известных аналогов, данный программный комплекс представлен дружелюбным графическим интерфейсом пользователя, за реализацию которого отвечают библиотеки Qt и Qwt, имеющие богатый набор виджетов – программ, облегчающих доступ к информации, с помощью которых можно создавать сложный графический интерфейс. Используемые библиотеки являются кроссплатформенными, что позволяет сделать более простой процедуру переноса приложений на разные ОС, применяется сглаживание графиков результатов тестирования. В конце работы приводятся результаты тестовых измерений.

Ключевые слова: Linux, измерение производительности, функции операционных систем, библиотеки QT и QWT, язык C++

**PROGRAM COMPLEX FOR EVALUATION OF PERFORMANCE OF MAIN
FUNCTIONS OF OPERATING SYSTEMS****Gorbaneva I.V., Martyshkin A.I.***Penza State Technological University, Penza, e-mail: Alexey314@yandex.ru*

The article describes the created software package for evaluating the performance of operating system functions. The purpose of the work is to create a set of programs necessary for evaluating the execution time of the software functions of single-processor and multiprocessor operating systems. The complex performs performance analysis of a number of functions of operating systems: mutexes, semaphores, read-write locks, FIFO and PIPE channels, TCP and UDP sockets, context switching, system calls (read, write, open, close). Unlike existing known analogs, this software package is presented by a user-friendly graphical user interface, for the implementation of which the Qt and Qwt libraries have a rich set of widgets – programs that facilitate access to information with which you can create a complex graphical interface. The libraries used are cross-platform, which makes it possible to make the procedure of transferring applications to different OSes more simple, the graphs of test results are smoothed. At the end of the article, the authors give the results of the test measurements.

Keywords: Linux, performance measuring, features, operating systems, Qt and Qwt libraries, C++ language

Современные операционные системы (ОС) предоставляют разработчикам программного обеспечения большое количество разнообразных средств создания сложных программных систем, состоящих из нескольких процессов, которые могут выполняться на разных ЭВМ [1, 2]. Была поставлена цель – разработать программный комплекс (ПК), предназначенный для анализа времени выполнения программных функций однопроцессорных и многопроцессорных ОС.

Измерение производительности различных функций ОС необходимо не только разработчикам ОС, для оценки эффективности реализации определенной функции и сравнения производительности реализации в других ОС, но и прикладным программистам, которым измерение производительности функций ОС необходимо для выбора наиболее подходящих средств реализации своих проектов [3, 4].

ПК выполняет тестирование функций ОС, таких как: средства синхронизации процессов: взаимные исключения (мьютексы), семафоры, блокировки чтения-записи; средства межпроцессного обмена: именованные каналы FIFO, неименованные каналы PIPE, сокет TCP и UDP; переключение контекста; системные вызовы (чтение, запись, открытие, закрытие).

Сегодня получил распространение ПК Lmbench, позволяющий проводить тестирование nix-совместимых систем, имея в своём составе широкий спектр тестов для разных подсистем ОС. Данный пакет позволяет получить довольно целостное представление о проверяемой ОС. Однако пакет Lmbench имеет ряд недостатков, которые делают работу с ним не удобной, требующей многократного повторения однотипных действий и выполнения дополнительных вычислений для получения результатов [5, 6].

Итак, разработанный ПК для тестирования функций операционных систем делает работу тестирования более удобной, более продуктивной и доступной, чем ПК LMBench [7].

На рис. 1 представлена последовательность действий, которые производит разработанный ПК.

Работа с ПК выполняется на ЭВМ под управлением ОС семейства Linux. Сегодня наиболее распространенными ОС являются – Ubuntu, Fedora, SUSE и Mandriva. Для обеспечения работоспособности программы требуется nix-совместимая ОС с установленными библиотеками Qt и Qwt.

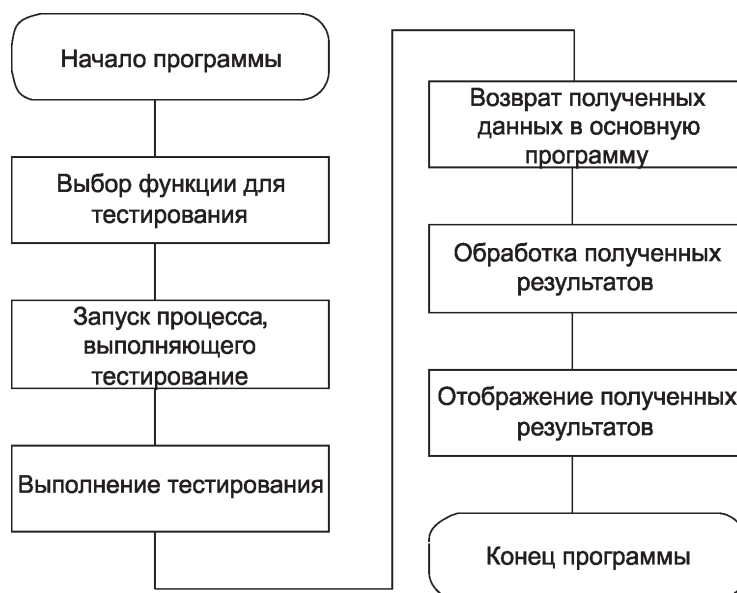


Рис. 1. Последовательность действий, выполняемых представляемым ПК



Рис. 2. Структура данных разработанного ПК

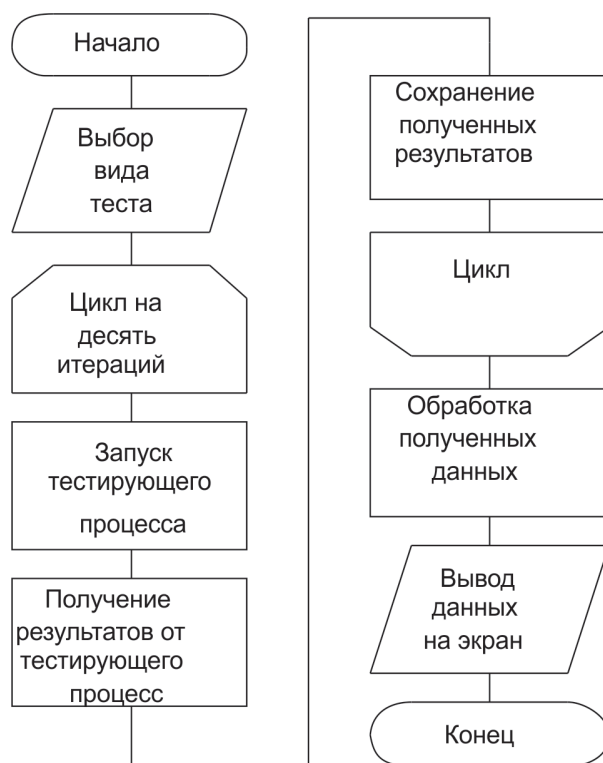


Рис. 3. Алгоритм работы основной программы

Структура данных разработанного ПК представлена на рис. 2. Алгоритм работы основной программы представлен на рис. 3.

Как основа пользовательского интерфейса принята структура с многооконным интерфейсом. Для различных задач создаются отдельные окна, которые обеспечивают не перегруженный интерфейс, необходимый для их решения. В описываемой ПК применена процедура фильтрации (сглаживания) графиков – технология, которая используется для устранения эффекта «зубчатости», который возникает на краях одновременно выводимого на экран множества плоских или объёмных изображений. Процедуру сглаживания производят по нескольким наборам расположенных подряд данных, причем их число обычно выбирают экспериментальным подбором [7, 8]. Почти все графики предлагаются в двух видах – необработанном и фильтрованном (сглаженном).

Процедура сглаживания действует на стиль отображения графика, который может быть построен прямолинейно от сообщения к сообщению, отчего он будет иметь угловатые очертания. Но при выборе сглаживания график будет иметь более плавные очертания. Уровень и алгоритм сглаживания для всех графиков одинаковый.

На рис. 4 и 5 представлены примеры тестов по измерению производительности переключения контекста и семафора.

Разработан программный комплекс для тестирования функций ОС: мьютексов, семафоров, переключения контекста системных вызовов (чтение, запись, открытие, закрытие) и других. Программа после выбора конкретного теста позволяет в автоматическом режиме собирать всю статистику, обрабатывать её и показывать конечные результаты в удобной и понятной для пользователя форме.

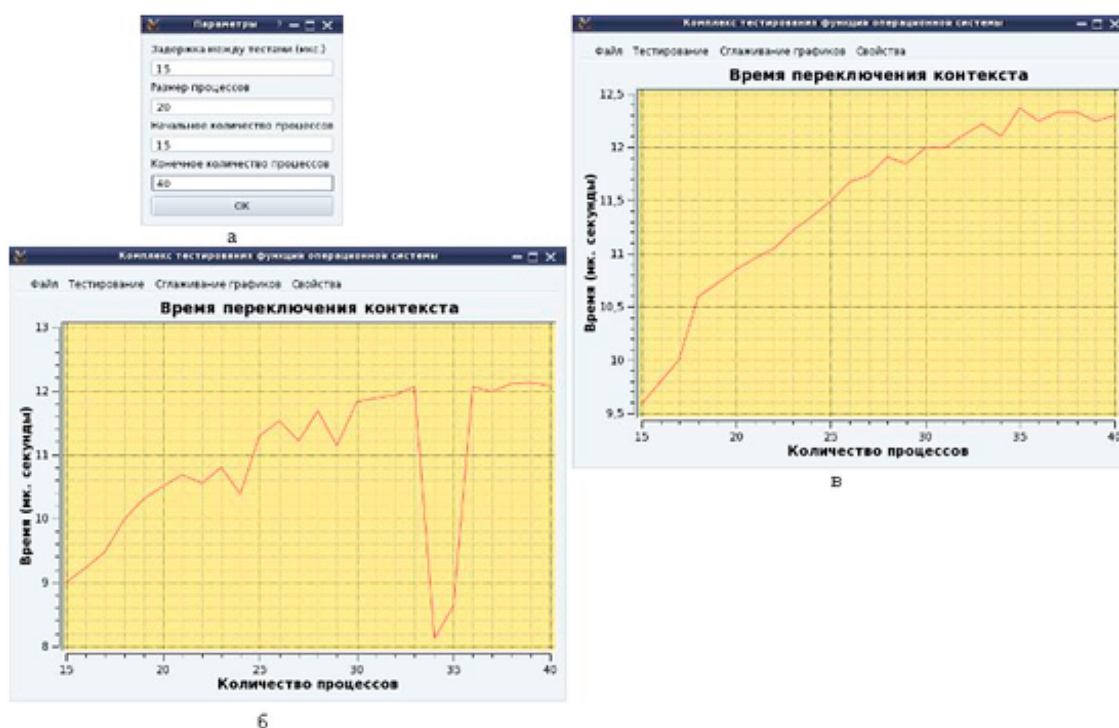


Рис. 4. Тест времени переключения контекста:
задание параметров (а), без сглаживания (б), с наиболее вероятным сглаживанием (в)

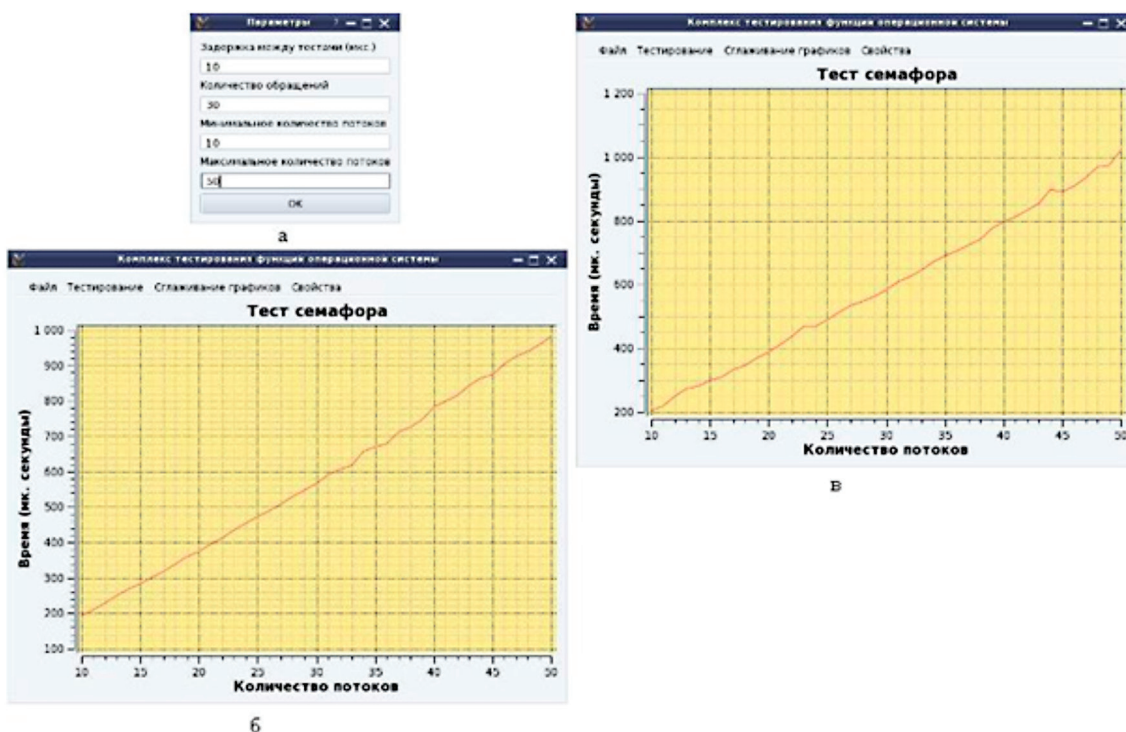


Рис. 5. Тест семафора:
задание параметров (а), без сглаживания (б), с наиболее вероятным сглаживанием (в)

В отличие от существующих программ-аналогов, описанный комплекс представлен дружественным интерфейсом пользователя, за визуализацию которого отвечают подключаемые библиотеки Qt и Qwt. Указанные библиотеки кроссплатформенные, что в свою очередь делает процедуру переноса программы на разные ОС более простой. Применяемый язык программирования C++ упрощает перенос программного комплекса на различные платформы и обеспечивает высокое быстродействие программы, что весьма сильно сказывается на получаемой точности результатов тестирования функций ОС.

Список литературы

1. Таненбаум Э., Бос Х. Современные операционные системы. – СПб.: Питер, 2015. – 1120 с.

2. Мартышкин А.И. Основные функции ядра современных операционных систем реального времени // Синтез науки и общества в решении глобальных проблем современности: сборник статей Международной научно-практической конференции. В 2-х частях / Отв. ред.: А.А. Сукиасян, 2016. – С. 82–84.

3. Стивенс У.Р. UNIX: разработка сетевых приложений. – Питер, 2007. – 1039 с.

4. Карасева Е.А., Мартышкин А.И. Обзор средств управления процессами и ресурсами многопроцессорных операционных систем // Международный студенческий научный вестник. – 2016. – № 3–1. – С. 80–81.

5. Стивенс У.Р. UNIX: взаимодействие процессов. – Питер, 2003. – 576 с.

6. Родригес К.З., Фишер Г., Смолски С. Linux. Азбука ядра. – М.: Кудиц-пресс, 2007. – 584 с.

7. Бикташев Р.А., Мартышкин А.И. Комплекс программ для измерения производительности функций операционных систем // XXI век: итоги прошлого и проблемы настоящего плюс. – 2013. – № 10 (14). – С. 190–197.

8. Валова О.О., Мартышкин А.И. Разработка, исследование и применение моделей вычислительных систем с виртуализацией // Современные информационные технологии. – 2014. – № 20. – С. 50–57.