

МЕТОДИЧЕСКИЕ УКАЗАНИЯ И ЗАДАНИЯ НА КУРСОВУЮ РАБОТУ ПО ДИСЦИПЛИНЕ «ОПЕРАЦИОННЫЕ СИСТЕМЫ»

ИССЛЕДОВАНИЕ МЕХАНИЗМОВ МЕЖПРОЦЕССНОГО ВЗАИМОДЕЙСТВИЯ

ЦЕЛИ И ЗАДАЧИ ВЫПОЛНЕНИЯ КУРСОВОЙ РАБОТЫ

Целью выполнения данной курсовой работы является закрепление и углубление теоретических знаний в области современных операционных систем, приобретение практических навыков разработки клиент-серверных приложений, использующих системные механизмы межпроцессного взаимодействия.

Задачи курсовой работы:

1. Изучить методы разработки клиент-серверных приложений.
2. Изучить механизмы обмена данными в Windows NT и GNU/Linux и особенности их применения на практике.
3. Изучить средства синхронизации процессов в Windows NT и GNU/Linux и особенности их применения на практике.
4. Изучить функции WinAPI и возможности Linux для получения системной информации, информации о процессах и потоках, состоянии памяти.
5. Разработать серверные приложения и приложение-клиент, обменивающиеся данными через заданный интерфейс.
6. Оформить результаты выполнения курсовой работы в виде пояснительной записки в соответствии с предъявляемыми требованиями.

ЗАДАНИЕ НА КУРСОВУЮ РАБОТУ

Разработать многозадачное приложение, выполняющее получение, сбор и вывод системной информации в соответствии с вариантом задания (таблица 1). Номер варианта задания на курсовую работу совпадает с номером, под которым записана фамилия студента в журнале группы.

Таблица 1. Индивидуальные варианты заданий

Вариант	Сервер 1	Сервер 2
	получает/выполняет и возвращает в качестве результата	
1, 19	<ul style="list-style-type: none">• имя компьютера и имя пользователя• перемещает окно серверного процесса согласно переданным координатам и возвращает результат выполнения (успех/ошибка)	<ul style="list-style-type: none">• приоритет серверного процесса• разрешен ли сбор данных SQM в рамках программы улучшения качества программного обеспечения

2, 20	<ul style="list-style-type: none"> • размер файла подкачки в байтах • количество свободных байтов файла подкачки 	<ul style="list-style-type: none"> • время работы серверного процесса • ширину и высоту экрана
3, 21	<ul style="list-style-type: none"> • код последней ошибки • текущее положение курсора 	<ul style="list-style-type: none"> • процент используемой физической памяти • процент используемой виртуальной памяти
4, 22	<ul style="list-style-type: none"> • количество мониторов • ширину и высоту основного монитора 	<ul style="list-style-type: none"> • полное имя модуля текущего процесса • объем свободной физической памяти в единицах, переданных клиентом (в байтах, мегабайтах, гигабайтах по выбору пользователя)
5, 23	<ul style="list-style-type: none"> • приоритет серверного процесса • идентификатор и дескриптор серверного процесса 	<ul style="list-style-type: none"> • код текущей раскладки клавиатуры • версию операционной системы
6, 24	<ul style="list-style-type: none"> • количество клавиш мыши • наличие колеса прокрутки 	<ul style="list-style-type: none"> • идентификатор серверного процесса • время работы серверного процесса в пользовательском режиме
7, 25	<ul style="list-style-type: none"> • название используемого видеоадаптера • размер клиентской области 	<ul style="list-style-type: none"> • количество свободных байтов файла подкачки • объем используемой физической памяти в единицах, переданных клиентом (в байтах, мегабайтах, гигабайтах по выбору пользователя)
8, 26	<ul style="list-style-type: none"> • архитектуру процессора • количество логических процессоров 	<ul style="list-style-type: none"> • количество процессов в системе • количество модулей серверного процесса
9, 27	<ul style="list-style-type: none"> • ширину и высоту рамки окна • изменяет название окна сервера на переданное из клиентского приложения и возвращает результат изменения (успех/ошибка) 	<ul style="list-style-type: none"> • размер файла подкачки в байтах • количество свободных байтов файла подкачки
10, 28	<ul style="list-style-type: none"> • ширину и высоту курсора • количество клавиш мыши 	<ul style="list-style-type: none"> • количество всех потоков в системе • список потоков сервера и их приоритеты
11, 29	<ul style="list-style-type: none"> • ширину и высоту основного монитора • цвет пикселя на сервере для переданных от клиента координат 	<ul style="list-style-type: none"> • идентификатор серверного процесса • дескриптор серверного процесса
12, 30	<ul style="list-style-type: none"> • название используемого видеоадаптера • скрывает окно серверного процесса на переданное время (от 1000 до 10000 мс) и возвращает результат выполнения (успех/ошибка) 	<ul style="list-style-type: none"> • процент используемой физической памяти; • процент используемой виртуальной памяти
13, 31	<ul style="list-style-type: none"> • координаты окна серверного процесса • разрешение основного монитора 	<ul style="list-style-type: none"> • количество модулей серверного процесса • количество потоков серверного процесса
14, 32	<ul style="list-style-type: none"> • цвет шрифта • изменяет цвет шрифта на сервере на переданный из клиентского приложения и возвращает результат (успех/ошибка) 	<ul style="list-style-type: none"> • объем доступной физической памяти; • процент свободной физической памяти

15, 33	<ul style="list-style-type: none"> • текущее положение курсора; • дескриптор раскладки клавиатуры по умолчанию 	<ul style="list-style-type: none"> • идентификатор серверного процесса • перечень собственных потоков с указанием их приоритета
16, 34	<ul style="list-style-type: none"> • продолжительность текущего сеанса работы • текущий часовой пояс 	<ul style="list-style-type: none"> • процент свободной физической памяти; • время работы серверного процесса в пользовательском режиме
17, 35	<ul style="list-style-type: none"> • размер окна серверного процесса • количество мониторов 	<ul style="list-style-type: none"> • количество потоков серверного процесса; • количество процессов на сервере
18, 36	<ul style="list-style-type: none"> • количество жестких дисков и тип файловой системы каждого • количество логических процессоров 	<ul style="list-style-type: none"> • объем физической памяти в единицах, переданных клиентом (в байтах, мегабайтах, гигабайтах по выбору пользователя) • объем свободной физической памяти в единицах, переданных клиентом (в байтах, мегабайтах, гигабайтах по выбору пользователя)

Общие указания к выполнению работы

Многозадачное приложение реализуется на основе технологии «клиент-сервер» средствами интегрированной среды разработки Microsoft Visual Studio/Visual C++ или редактора Visual Studio Code, по согласованию с преподавателем возможно использование других компиляторов и языков программирования. Процессы 1 и 2 выступают как серверы приложений, третий процесс является их клиентом. Обмен данными между клиентами и серверами осуществляется при помощи сокетов.

И клиентские, и серверные процессы оформляются как отдельные приложения, запускаются и завершают работу независимо друг от друга. Серверы должны быть многопоточными, то есть допускать одновременное подключение нескольких клиентов. Серверные процессы по запросу получают системную информацию в соответствии с вариантом задания и передают ее клиенту с указанием текущего времени. На стороне клиента необходимо реализовать интерфейс, позволяющий пользователю подключиться по выбору к первому или второму серверу, а также возможность раздельного получения и отображения данных по каждому из пунктов индивидуального задания.

Необходимо предусмотреть проверку на повторный запуск серверов, т.е. исключить ситуацию, когда запускается более чем один экземпляр каждого сервера. В течение всего времени подключения клиента к серверу информация в клиентской части должна обновляться либо автоматически, либо по команде от пользователя.

Дополнительные задания

1. При обновлении информации на клиенте по таймеру данные передаются только в том случае, если что-либо изменилось с момента предыдущего обновления.

2. Обновление информации на клиенте происходит по инициативе сервера как реакция на изменения на сервере.
3. Существует отдельный сервер, который ведет лог операций (отдельный для каждого сервера, с сохранением в файл), в котором фиксируются подключения клиентов, переданные запросы, ошибки и т.д. Взаимодействие между серверами осуществляется с помощью каналов.
4. Запуск клиентов в контейнерах.
5. Наличие версии сервера или клиента под ОС Linux (т.е., если клиент и сервер работают под разными ОС).
6. Демонстрация работы разработанных приложений на двух устройствах (физических или в виртуальных машинах).
7. Реализация графического интерфейса на стороне клиента.

СТРУКТУРА И ПРИМЕРНОЕ СОДЕРЖАНИЕ ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ

Пояснительная записка к курсовой работе должна включать в себя следующие структурные элементы в указанной ниже последовательности:

титульный лист;

задание на курсовую работу;

содержание;

введение, в котором раскрываются актуальность и значение темы, формулируется цель работы;

основная часть:

- постановка задачи;
- краткие теоретические основы разрабатываемой темы (суть, область применения, достоинства и недостатки вычислительной архитектуры «клиент-сервер»; обзор системных средств коммуникации и синхронизации процессов);
- разработка и описание алгоритма работы приложения: словесное описание функционирования или UML-диаграмма последовательности, перечень запросов от клиента к серверу, с указанием входных и выходных данных, описание выбранного алгоритма обновления информации на клиенте, описание алгоритма проверки на повторный запуск сервера;
- описание структуры программы (модулей, основных функций, классов и т.д.);
- результаты применения программы в виде снимков экрана после запуска приложения;

заключение;

список использованных источников;

приложение, содержащее исходные коды разработанных программ с необходимыми комментариями.

Требования к оформлению пояснительной записки

Пояснительная записка к курсовой работе выполняется на стандартных листах бумаги формата А4. Поля: левое поле - 35 мм, правое – 10 мм, верхнее и нижнее – 20 мм. Примерное количество знаков на странице – 2000. Шрифт Times New Roman размером 12, межстрочный интервал 1,5. Страницы текста и рисунки имеют сквозную нумерацию. Первой страницей является титульный лист, на котором номер страницы не проставляется.

ЗАЩИТА КУРСОВОЙ РАБОТЫ

Защита курсовой работы является заключительным этапом курсового проектирования и проводится не позднее середины последней недели перед началом сессии. Выполненная курсовая работа (оформленная пояснительная записка и архив с исходными текстами программ) в электронной форме загружается в личный кабинет студента в ЭУ МТУСИ.

Защита состоит в демонстрации студентом функционирования разработанного многозадачного приложения и ответах на вопросы преподавателя (по алгоритмам и кодам программ, а также на теоретические вопросы в рамках содержания работы). Защита курсовой работы позволяет выявить уровень знаний студента, степень его самостоятельности в выполнении курсовой работы.

В результате защиты курсовая работа, согласно Положению о промежуточной аттестации студентов МТУСИ, оценивается дифференцированной отметкой («неудовлетворительно», «удовлетворительно», «хорошо», «отлично»).

Студенты, не защитившие курсовую работу, до экзамена по дисциплине "Операционные системы" не допускаются.

Критерии оценки курсовой работы

Критерии оценки курсовой работы	Оценка
Функционал приложения реализован полностью и правильно, выполнены 3 дополнительных задания, пояснительная записка оформлена в соответствии с предъявляемыми требованиями, грамотные и точные ответы на поставленные вопросы, свободное владение средствами языка программирования	<i>отлично</i>
Функционал приложения реализован полностью и правильно, выполнено 1 дополнительное задание, допустимы	<i>хорошо</i>

незначительные недочеты, устраненные студентом самостоятельно по указанию преподавателя, допустимы несущественные неточности в ответе на вопросы, правильное применение теоретических положений при решении задач, владение средствами языка программирования	
Функционал приложения реализован полностью, допустимы незначительные недочеты, допустимы неточности в ответе на вопросы, недостаточно правильные формулировки, погрешности в оформлении пояснительной записки	удовлетворительно
Допущены существенные ошибки в ходе работы, незнание теоретического материала, относящегося к выполненной работе, неумение сформулировать правильные ответы на вопросы, много неточностей в оформлении пояснительной записки	<i>неудовлетворительно</i>