

Практическая работа №1

Разработка простого графического интерфейса пользователя для решения различных прикладных задач

Цель работы: получение практических навыков по разработке простого графического интерфейса пользователя для решения различных задач.

Задачи:

1. Изучить теоретический материал
2. Написать программы для решения практических заданий
3. Ответить на контрольные вопросы.
4. Оформить отчет.

Теоретический материал

Интерфейс (от англ. interface — поверхность раздела, перегородка) — совокупность средств, методов и правил взаимодействия (управления, контроля и т.д.) между элементами системы.

Этот термин используется во многих

Интерфейсы являются основой взаимодействия всех современных информационных систем. Если интерфейс какого-либо объекта (персонального компьютера, программы, функции) не изменяется (стабилен, стандартизирован), это даёт возможность модифицировать сам объект, не перестраивая принципы его взаимодействия с другими объектами (например, научившись работать с одной программой под Windows, пользователь с лёгкостью освоит и другие — потому, что они имеют одинаковый интерфейс).

В вычислительной системе взаимодействие может осуществляться на пользовательском, программном и аппаратном уровнях. Соответственно, согласно этому, интерфейсы могут существовать как:

Способ взаимодействия физических устройств («Железный» (аппаратный) интерфейс, чаще всего речь идёт о компьютерных портах)

Способ взаимодействия виртуальных устройств (Программный интерфейс) - функциональность, которую некоторый программный компонент предоставляет другим программным компонентам. Можно различать два вида такой функциональности: та, что используется при создании прикладных программ, и та, что используется при создании системных компонентов. Первая обычно называется интерфейсом программирования приложений (API, англ. application programming interface), вторая может называться интерфейсом программирования компонентов операционной системы или интерфейсом системного программирования (SPI, англ. system programming interface). Кроме того, программные интерфейсы могут быть разноуровневыми, относясь либо к уровню программного кода (API и SPI), либо к уровню кода машинного (ABI, англ. application binary interface, и SBI, англ. system binary interface).

Способ взаимодействия человек-машина (Интерфейс пользователя) - — разновидность интерфейсов, в котором одна сторона представлена человеком (пользователем), другая — машиной/устройством. Представляет собой совокупность средств и методов, при помощи которых пользователь взаимодействует с различными, чаще всего сложными, машинами, устройствами и аппаратурой.

Интерфейс пользователя, он же пользовательский интерфейс (UI — англ. user interface) — разновидность интерфейсов, в котором одна сторона представлена человеком (пользователем), другая — машиной/устройством. Представляет собой совокупность средств и методов, при помощи которых пользователь взаимодействует с различными, чаще всего сложными, машинами, устройствами и аппаратурой.

Под совокупностью средств и методов интерфейса пользователя подразумеваются:

Средства:

- вывода информации из устройства к пользователю — весь доступный диапазон воздействий на организм человека (зрительных, слуховых, тактильных, обонятельных и тд.) — экраны (дисплеи, проекторы) и лампочки, динамики, зуммеры и сирены, вибромоторы и тд. и тп.

- ввода информации/команд пользователем в устройство — множество всевозможных устройств для контроля состояния человека — кнопки, переключатели, потенциометры, датчики положения и движения, сервоприводы, жесты лицом и руками, даже съём мозговой активности пользователя.

По наличию тех или иных средств ввода, интерфейсы разделяются на типы — жестовый, голосовой, брэйи, и тд., возможны смешанные варианты. Средства эти должны быть необходимыми и достаточными, быть удобными и практичными, расположенными и скомпонованными разумно и понятно, соответствовать физиологии человека, не должны приводить к негативным последствиям для организма пользователя (всё это входит в понятие эргономики).

Методы:

- набор правил, заложенных разработчиком устройства, согласно которым совокупность действий пользователя должна привести к необходимой реакции устройства и выполнения требуемой задачи — т. н. логический интерфейс

- Интерфейс пользователя компьютерного приложения включает:

- средства отображения информации, отображаемую информацию, форматы и коды;

- командные режимы, язык «пользователь — интерфейс»;

- устройства и технологии ввода данных;

- диалоги, взаимодействие и транзакции между пользователем и компьютером, обратную связь с пользователем;

- поддержку принятия решений в конкретной предметной области;

- порядок использования программы и документацию на неё.

Пользовательский интерфейс часто понимают только как внешний вид программы. Однако, на деле пользователь воспринимает через него всю программу в целом, а значит, такое понимание является слишком узким.

В действительности ПИ объединяет в себе все элементы и компоненты программы, которые способны оказывать влияние на взаимодействие пользователя с программным обеспечением (ПО), это не только экран, который видит пользователь.

Текстовый пользовательский интерфейс, ТПИ (англ. Text user interface, TUI; также Character User Interface, CUI) — разновидность интерфейса пользователя, использующая при вводе-выводе и представлении информации исключительно набор буквенно-цифровых символов и символов псевдографики. Характеризуется малой требовательностью к ресурсам аппаратуры ввода-вывода (в частности, памяти) и высокой скоростью отображения информации, поэтому широко использовался на начальном этапе развития вычислительной техники. Также, его разновидность - интерфейс командной строки - имеет отдельные преимущества в юзабилити перед графическим интерфейсом. Поэтому программы, основанные на ТПИ, имеют некоторое распространение до настоящего времени, особенно в специфических сферах и на маломощном оборудовании.

ТПИ, использующий в качестве элементов интерфейса только и исключительно вводимые с клавиатуры текстовые строки (команды), называется интерфейсом командной строки (т. н. консольные программы — программы, использующие интерфейс командной строки, где информация выводится на консоль).

В простейшем случае ТПИ использует интерфейс командной строки, однако многие программы с помощью интерактивных элементов создают более дружелюбный интерфейс, приближающийся по удобству к графическому.

Интерфейс командной строки (англ. Command line interface, CLI) — разновидность текстового интерфейса (CUI) между человеком и компьютером, в котором инструкции компьютеру даются в основном путём ввода с клавиатуры текстовых строк (команд), в UNIX-системах возможно применение мыши. Также известен под названием консоль.

Интерфейс командной строки противопоставляется системам управления программой на основе меню, а также различным реализациям графического интерфейса.

Формат вывода информации в интерфейсе командной строки не регламентируется; обычно это также простой текстовый вывод, но может быть и графическим, звуковым и т. д.

CLI использует меньше ресурсов компьютера по сравнению с GUI.

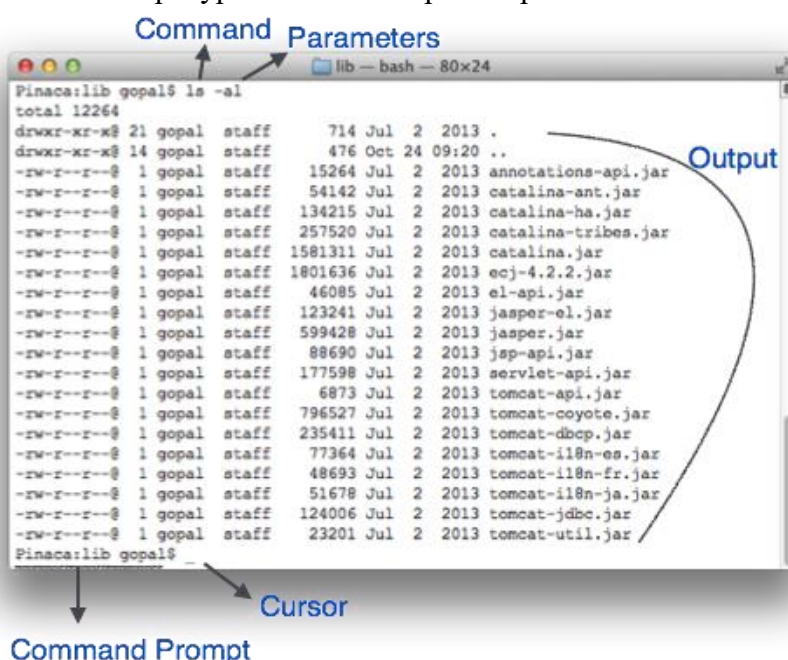


Рисунок 1 - Элементы CLI

Текстовый интерфейс командной строки может иметь следующие элементы (рис.1):

- Командная строка – это текстовое уведомление, которое в основном показывает контекст, в котором работает пользователь. Он генерируется системой программного обеспечения.

- Курсор – это небольшая горизонтальная линия или вертикальная черта высоты линии, представляющая положение символа при наборе текста. Курсор в основном находится в состоянии мигания. Он перемещается, когда пользователь пишет или удаляет что-то.

- Команда – команда является исполняемой инструкцией. Может иметь один или несколько параметров. Выходные данные при выполнении команды отображаются на экране в виде строки. Когда вывод получен, командная строка отображается на следующей строке.

Общие требования к приложениям CLI привел Дэвид Коупленд в книге «Build Awesome Command-Line Application in Ruby»:

- Easy to use — Оно должно быть простым в использовании и иметь четкую цель.

- Helpful — Это означает, что пользователь должен иметь простой доступ к помощи о том, как что делает приложение, как его запускать, и как его настроить.
- Plays well with others — Приложение должно быть способным взаимодействовать с другими приложениями.
- Has sensible defaults but is configurable — Стандартные сценарии использования должны быть доступны.
- Installs painlessly — Легко устанавливается вместе со всеми зависимостями, устанавливает путь к приложению в переменные окружения для более простого запуска.
- Fails gracefully — В случае ошибки в вызове приложения, оно должно сообщить, в чем была ошибка и как её исправить.
- Gets new features and bug fixes easily — Приложение должно быть поддерживаемым.
- Delights users — Вывод приложения должен быть эстетичным.

Практические задания

В соответствии с вариантом написать программу с интерфейсом командной строки с учетом требований к приложениям CLI.

1. Составить программу, которая будет генерировать случайные неповторяющихся целых чисел в диапазоне, который укажет пользователь.
2. Составить программу Калькулятор.
3. Составить программу нахождения точку пересечения отрезка с окружностью. Координаты вершин отрезка, центра окружности и радиус задаются с клавиатуры.
4. Составить программу для работы с матрицей. Дана матрица $A(x,y)$. Заполнить ее целыми случайными числами в диапазоне от 1 до 100 (x, y задаются пользователем)
5. Написать программу для перевода числа из десятичной системы в двоичную.
6. Написать программу, которая для введенных пользователем чисел a, b, c осуществляет перераспределение значений по возрастанию
7. Написать программу для вычисления корней уравнения квадратного трехчлена.
8. Написать программу для решения задачи. Даны различные целые числа (вводятся с клавиатуры). Найти среди них два числа, модуль разности которых имеет: а) наибольшее значение; б) наименьшее значение.
9. Написать программу, которая будет определять количество цифр в заданном числе и их сумму.
10. Написать программу для подсчета суммы чисел, меньших определенного числа, из последовательности из случайных чисел.

Контрольные вопросы

1. Что представляет собой интерфейс?
2. Назовите уровни интерфейсов в компьютерных системах?
3. Что представляет собой интерфейс пользователя?
4. Какие элементы входят в пользовательский интерфейс?
5. Что относится к средствам интерфейса пользователя?
6. Что включают в себя методы пользовательского интерфейса?
7. Как выглядит текстовый пользовательский интерфейс?
8. Какие элементы входят в интерфейс командной строки?
9. Какие общие требования к приложениям CLI привел Дэвид Коупленд
10. Какой интерфейс пользователя наиболее распространен в настоящее время?