Operating Systems Assignment

Windows and Linux

Zain Mirza

2021

Windows

**Explain the purpose of the Win32 API subsystem and identify three functional areas provided through the windows application programming interface. Support your answer by including some suitable native function call names for each of the three functional areas.**

*The Windows Application Programming Interface (API) is an interface that is used to develop 32-bit applications. The windows API allows developed applications to work on all Windows versions. The main advantage here is that applications can continue to work for generations whilst also making use of the features that are unique to the in-use Windows version.*

*Three examples of functional areas of the Windows API include:*

* *Graphics and Multimedia – Integrate formatted text, graphics, audio, and video.*
* *Networking – The ability to communicate between applications on different computers on a network*
* *Systems Service – Gives access to computer resources and the operating system which includes memory, file system, devices, processes, and threads*

**Analyse the source code of the Windows Procedure (WndProc) function in the Appendix. Explain the general functionality of the program. Identify which windows messages are decoded and how they are made use of. How is it possible to distinguish between menu item selections? Refer to the source code in your answers.**

## *The wndProc is an application-defined function that processes messages sent to a window. This function is called when an application registers its window class. The program is creating the properties of a basic window including the window height and width. “WM\_CREATE” is used to set a new message that the application displays when opened. This is then followed by a basic if statement that will allow the message on the app to change depending on what the user clicks on.*

**Text

Description automatically generated**

*This is the opening part of the program; it declares some variables needed for a later assignment. Each of these are given a unique ID which the program stores as one of the variables in the background.*

Text

Description automatically generated with medium confidence

*This part of the code is creating the general properties of the Windows. It is using resources from the Windows library such as “HDC” and “PAINTSTRUCT.” This sets how the application will function look at a default state.*

*“WM\_CREATE” is used to create a default state of the application when it opens.*

A picture containing text

Description automatically generatedGraphical user interface, text, application

Description automatically generated

*“WM\_COMMAND” is used to prepare the application to expect some user input. When a user opens the menu and selects one of the options, it generates an ID. The program then looks to see which variable matches the ID. For example, at the start of the program, ID 40003 matches with “IDM\_LINUX,” which means the program will print the if statement that corresponds to the matched variable.*

Chart

Description automatically generated with medium confidenceGraphical user interface

Description automatically generated with low confidence**Discuss how such an event-driven approach is beneficial for interactive graphical user interface applications. Comment on the use of system resources and latencies in your discussion.**

*“WM\_PAINT” is used to request the operating system to paint a specified portion of the application window.*

*“WM\_DESTROY” is a message that is used when the user closes the application. This prints the closing message and ends the application.*

* *Flexibility – Windows has a library of functions that work on applications. This means that these functions can be used across multiple versions of Windows.*
* *Functionality – Graphical user interfaces can offer lots of options for a user by adding specific functions to user interface objects such as buttons*
* *Simplicity – From the perspective of the user, they can only see objects such as buttons, hyperlinks, and tabs. There is a lot of background processing the operating system is doing which the user cannot normally see.*

Linux

**The Linux operating system consist of the Linux kernel and a layer of GNU standard libraries. Describe the purpose of the kernel and its general architecture. Explain why the kernel runs in the protected mode (kernel mode) and the standard libraries in user mode.**

*The kernel in the Linux is* *the main component in a Linux operating system. The role of the kernel is to control all major functions of a systems hardware. The kernel takes control of a system in four main areas:*

* *Memory Management – Keep track of how much memory is being used from the system*
* *Process Management – Determine what processes need to use the CPU, how long they need to use it, and how important one process is.*
* *Device Drivers – Interpret any incoming hardware and processes*
* *System calls and security – Receive requests for service from processes.*

*Having direct access to the kernel when booting up a Linux system will give one direct access to the four core areas of a system. Any mishandling of these areas by a user could compromise the entire system. Therefore, the Linux kernel is protected and only runs on kernel mode. Standard libraries on the other hand runs on user mode so that a Linux user can make use of numerous functions stored in a Linux library that can be used in all versions of Linux.*

**The Linux command line interface allows to combine commands using pipes to form more complex command pipelines. Briefly explain the concept of pipes and discuss the benefits of such functionality.**

*Pipes in Linux is used to generate multiple commands together. When doing this, the output of one command acts as input. The main benefit of using pipes is that one can go through multiple commands and procedures in one line as opposed to typing line by line thus increasing efficiency.*

**Search for information in a server log file using Linux command line tools. Using a Linux text editor, create the log.txt file with the content shown below.**

1. List all lines with error messages in the log.txt file.

Text

Description automatically generated

*The “cat” command is used to view the context of a file without editing it. The “grep” command can be used to search for a specific term inside the file. A pipe symbol is used to separate the two commands.*

1. Print out the number of warnings in the log.txt file.

A picture containing text

Description automatically generated

*The “wc” command is a word counting command for Linux. -l is added so that only the word warning that starts on a new line is counted*

1. List all warnings regarding the user andreas in the log.txt file.

*Another benefit of the pipe symbol is that multiple “grep” commands can be used on one line.*

A screenshot of a computer

Description automatically generated with medium confidence