Worksheet 2.2: OS Structures, Free-Response Questions

Q1. Give three distinct advantages of writing an operating system in a high-level language compared to writing it in assembly. Give a clear description of each advantage, not just a single word or phrase.

1. **Portability**: Writing an OS in a high-level language allows the code to be easily compiled and run on different hardware platforms with minimal modifications. This is because high-level languages abstract away the underlying hardware specifics, making the OS more adaptable to various architectures.

2. **Ease of Development and Maintenance**: High-level languages provide more readable and understandable code, which simplifies the development process. They offer better support for complex data structures and algorithms, reducing the chance of errors. Additionally, debugging and maintaining high-level code is generally easier due to the availability of advanced debugging tools and the higher level of abstraction.

3. **Enhanced Productivity**: High-level languages typically include libraries and frameworks that provide pre-built functions and utilities, speeding up the development process. Developers can focus on implementing OS features rather than dealing with low-level hardware details, thus increasing overall productivity and reducing development time.

Q2. Compare the **monolithic** and the **loadable-module** OS structures by giving a <u>**clear**</u> advantage of each structure relative to the other. Explain each advantage within the line limit.

Advantage of monolithic relative to loadable modules **(Limit: 3 lines)**

Monolithic OS structures generally offer better performance because all the services run in the same address space, reducing the overhead associated with context switching and inter-module communication. This unified structure allows for faster execution of system calls and more efficient memory management, leading to overall higher system performance.

Advantage of loadable modules relative to monolithic **(Limit: 3 lines)**

Loadable modules provide greater flexibility and extensibility, allowing the OS to dynamically load and unload services as needed without rebooting. This makes it easier to update and expand the OS with new features or drivers.