COM 1122 – introduction to computer systems assignment 2

Name:

Tshegameno Kamonewa Bianca

Student Number:

24035383

Date:

13 April 2025

Question 1

1. **The central processing unit :**

This is the brain of any computer system. The central processing unit is made up of three parts

- The control unit
- The arithmetic logic
- The primary storage unit

The control unit:

The control unit controls the operations of the entire computer system. The control unit gets the instructions from the program stored in primary storage unit interprets these instructions an subsequently directs the other units to execute the instructions. Thus it manages and coordinates the entire computer system.

The Arithmetic logic:

The arithmetic logic unit (ALU) actually executes the instructions and performs all the instructions and performs all the calculations and decisions. The data is held in the primary storage unit and transferred to ALU whenever necessary. Data can be moved from primary storage to arithmetic logic unit a number of times before the entire processing is complete. After the completion, the results are sent to output storage and output storage and output devices.

The primary storage unit:

The is also called the main memory. Before the actual processing starts the data and the instructors fed to the computer through the input units are stored in this primary storage unit. Similarly, the data which is to be output from the computer system is also temporarily stored in the primary memory. It is also the area where the intermediate results of calculations are stored. The main memory has the storage section that holds the computer programs during execution.

Thus the primary storage unit

- Stores data and programs during actual processing
- Stores temporary results of intermediate processing
- Stores results of execution temporarily

The clock speed:

The clock speed of a computer measures how fast its central processing unit can process instructions. It is measured in GHz

A higher clock speed can improve performance but efficiency depend on the workload and how well the CPU and other components work together.

Slower processor **2.0 GHz, dual core**	**Faster processor** **4.0 GHz, quad core**
Handles basic tasks like word processing and web browsing efficiently	Handles resources intensive task like video editing, gaming and software development
May struggle with multitasking	Excels in multitask

2. **RAM (Random Access Memory)** plays a critical role in multitasking because it temporarily stores data that your system and active applications are currently using. It's like a workspace: the more RAM you have, the bigger the desk. You can spread out more papers (applications) and work on them simultaneously without running out of space.

How More RAM Enhances Multitasking

When you run multiple applications, each requires a portion of RAM to store its active data. If you have more RAM:

Your system can allocate enough memory for each application, allowing them to run smoothly.

It reduces the need to constantly move data back and forth between RAM and the hard drive (a slower process), which leads to faster and more efficient multitasking.

What Happens When the System Runs Out of RAM

If your system exhausts its RAM:

It resorts to using the hard drive or SSD for temporary storage, known as "virtual memory" or "paging." This process is significantly slower than RAM.

Applications may freeze, crash, or take longer to respond, affecting the overall performance.

Practical Example

Enough RAM: Imagine you have 16 GB of RAM. You’re running a browser with multiple tabs, a word processor, and video editing software. Each application gets sufficient memory, and you can switch between them seamlessly.

Insufficient RAM: With only 4 GB of RAM, running the same set of applications leads to trouble. Your system may start swapping data to virtual memory, resulting in slower performance. Switching between applications becomes laggy, and your video editor might crash due to insufficient memory.

3. **When choosing between an HDD (Hard Disk Drive) and an SSD (Solid State Drive),** it's essential to understand their differences and how they impact a computer's speed and performance. Here's a comparison:

HDD vs. SSD: Key Differences

Feature	HDD	SSD
Technology	Uses spinning magnetic disks and a read/write head.	Uses flash memory with no moving parts.
Speed	Slower, with read/write speeds around 125 MB/s.	Faster, with speeds up to 550 MB/s or more.
Durability	Prone to damage due to moving parts.	More durable and resistant to physical shock.
Noise	Produces noise due to spinning disks.	Silent operation.
Cost	Cheaper per GB.	More expensive per GB.

Why SSDs Are Faster

SSDs provide faster data access because they use flash memory, which allows data to be read and written electronically rather than mechanically. Unlike HDDs, which rely on a spinning disk and a moving read/write head to locate data, SSDs can access any location on the drive instantly. This eliminates delays caused by physical movement.

Practical Example

- System with HDD: A laptop with an HDD might take 1-2 minutes to boot up and load the operating system. Opening large applications like Photoshop could take 20-30 seconds.
- System with SSD: A laptop with an SSD can boot up in 10-15 seconds and launch the same application in just 5-10 seconds. Tasks like file transfers and software installations are also significantly faster

Question 2

a. Software optimization is the process of improving the efficiency of a program or system to ensure it uses fewer resources (like memory, CPU, or storage) while maintaining or enhancing its performance. This can involve refining the code, optimizing algorithms, or improving the system's architecture.

Why Systems Slow Down Over Time

1. **Software Bloat:** Over time, updates to software can add unnecessary features or poorly optimized code, increasing resource usage.
2. **Fragmentation:** Files on traditional HDDs can become fragmented, slowing down data access. (This is less of an issue with SSDs.)
3. **Background Processes:** Accumulation of startup programs and background processes can consume resources, leaving less for active tasks.
4. **Cache and Temporary Files:** Excessive buildup of cache or temporary files can reduce available storage and slow down operations.
5. **Outdated Drivers or Software:** Older drivers or software may not be optimized for newer hardware or operating systems.

Examples of Well-Optimized vs. Poorly Optimized Programs

- Well-Optimized Program: A lightweight web browser like Google Chrome or Microsoft Edge that efficiently manages memory, loads pages quickly, and minimizes CPU usage.
- Poorly Optimized Program: A bulky application that consumes excessive RAM and CPU, causing the system to lag. For instance, an older version of a video editing tool that crashes frequently due to inefficient memory management.

Practical Impact

- A well-optimized program runs smoothly, even on older hardware, and allows multitasking without significant slowdowns.
- A poorly optimized program can cause the system to freeze or lag, especially when running alongside other applications.

b. Programming languages are broadly categorized into high-level and low-level languages based on their abstraction from machine code. Here's a breakdown:

High-Level vs. Low-Level Languages

- **High-Level Languages:** These are closer to human language, making them easier to read, write, and maintain. They are platform-independent and rely on compilers or interpreters to translate code into machine language. Examples include Python, Java, and C++.
- **Low-Level Languages:** These are closer to machine language, offering direct control over hardware. They are faster and more efficient but harder to write and debug. Examples include Assembly and Machine Code.

Why C++ is Used for Gaming

C++ is favored for gaming due to its:

- **Performance:** It provides direct control over hardware and memory, enabling developers to optimize resource usage for high-speed graphics rendering.
- **Object-Oriented Features:** C++ supports complex game architectures with reusable code through classes and inheritance.
- **Libraries:** It has robust libraries like DirectX and OpenGL for graphics and physics engines.

Why Python is Used for AI Development

Python is ideal for AI because of:

- **Ease of Use:** Its simple syntax allows developers to focus on algorithms rather than coding intricacies.
- **Libraries and Frameworks:** Python boasts powerful libraries like TensorFlow, PyTorch, and NumPy for machine learning and data processing.
- **Community Support:** A large community ensures extensive resources and troubleshooting.

Performance Comparison

1. **Gaming Graphics (C++):** A game developed in C++ can render complex 3D environments at high frame rates due to its ability to directly manage memory and optimize hardware usage.
    - **Example:** A C++ game engine like Unreal Engine can handle real-time physics and high-resolution textures seamlessly.
2. **AI Development (Python):** An AI model built in Python can process large datasets efficiently, leveraging libraries like TensorFlow for deep learning.
    - **Example:** Training a neural network in Python might take longer than in C++ due to Python's slower execution speed, but its simplicity and library support make development faster overall.

In summary, C++ excels in performance-critical applications like gaming, while Python shines in rapid development and algorithm-heavy tasks like AI.

c. When deciding between Python and JavaScript for web development, it's important to consider their efficiency, performance, and suitability for different types of web-based applications. Here's a comparison:

Python vs. JavaScript for Web Development

Aspect	**Python**	**JavaScript**
Efficiency	Python is known for its simplicity and readability, making it easier to write and debug the code.	JavaScript is highly efficient for dynamic and interactive web applications.
Performance	Python is slower in execution compared to JavaScript, as it is an interpreted language.	JavaScript is faster, especially for client-side tasks, due to its asynchronous capabilities.
Use cases	Ideal for back-end development with frameworks like Django and Flask.	Essential for front-end development and can also be used for back-end with Node.js..
Libraries/Framewors	Python has robust libraries for data processing and automation.	JavaScript has a vast ecosystem of frameworks like React, Angular, and Vue.js for building interactive UIs.
Learning curves	Easier for beginners due to its straightforward syntax.	Slightly steeper learning curve due to concepts like

		asynchronous programming and closures.

Advantages and Disadvantages

Python

- **Advantages:**
    - Beginner-friendly syntax.
    - Strong support for back-end development.
    - Extensive libraries for tasks like data analysis and machine learning.
- **Disadvantages:**
    - Slower execution speed for web applications.
    - Limited support for front-end development.

JavaScript

- **Advantages:**
    - Runs directly in browsers, making it essential for front-end development.
    - Supports asynchronous programming for real-time applications.
    - Versatile for full-stack development with Node.js..
- **Disadvantages:**
    - More complex syntax and behaviors.
    - Security risks due to its client-side nature.

Practical Example

- **Python:** A developer might use Python with Django to build a content-heavy website with complex back-end logic, such as an e-commerce platform.
- **JavaScript:** A developer might use JavaScript with React to create a highly interactive and dynamic single-page application, like a social media platform.

Ultimately, the choice depends on the project's requirements. If the focus is on interactivity and responsiveness, JavaScript is the better choice. For back-end-heavy applications or projects requiring data processing, Python is more suitable.