|  |  |
| --- | --- |
| Paralllel computing exam  Mr Lars | Abstract    Fernando Nicolao Chikanda  [Course title] |

**Question 1**

**B**

The scenario is a metro station of a refugee camp in India for 900million.

The train only accommodates 1000 and there and only 4 rail ways and 10 trains going and returning, total trains 20.

A) Explain the computing concepts

And the parts to optimize

B) What will be the worst and best scenario?

**Best Case Scenario:**

1**. Efficient Queueing System:**

- Implementation of a digital ticketing system with time slots

- Clear demarcation of queues with barriers and signage

- Adequate staffing to manage crowds and assist vulnerable individuals

- Priority lanes for elderly, disabled, and families with young children

**2. Optimal Scheduling:**

- AI-powered scheduling algorithm that adapts to real-time demand

- Evenly distributed train departures to minimize peak congestion

- Quick turnaround times for trains at end stations

- Coordination with other transportation modes for onwards journeys

3. **Effective Resource Allocation:**

- All 20 trains operating at full capacity (1000 passengers each)

- Balanced utilization of all 4 railways

- Efficient loading and unloading procedures to minimize dwell time at stations

- Well-maintained trains and tracks to prevent breakdowns

4**. Clear Information Systems:**

- Multi-lingual digital displays and announcements

- Mobile app for real-time updates and personal queue management

- Help desks with translators for personalized assistance

- Clear signage and wayfinding throughout the station

5. **Maximized Throughput:**

- Assuming 30-minute round trips (optimistic but possible in this scenario)

- 20 trains x 1000 passengers x 24 hours / 0.5 hours = 960,000 passengers per day

- At this rate, it would take about 938 days (roughly 2.5 years) to transport all 900 million refugees

Factors contributing to best case:

- Adequate funding and resources

- Well-trained staff and volunteers

- Cooperation from refugees

- Good weather conditions

- Political stability and support

**Worst Case Scenario:**

1. Chaotic Queueing:

- No organized system for queuing or ticketing

- Overcrowding leads to pushing, stampedes, and injuries

- No priority for vulnerable individuals

- Lack of crowd control measures

2. **Inefficient Scheduling:**

- Random or poorly planned train departures

- Long waiting times between trains

- No coordination with other transportation modes

- Frequent delays due to poor management

3. **Poor Resource Allocation:**

- Uneven use of railways, with some overloaded and others underutilized

- Trains operating below capacity due to boarding inefficiencies

- Frequent breakdowns due to overuse and lack of maintenance

- Long dwell times at stations due to slow boarding/alighting

4. **Lack of Information:**

- No clear communication about schedules or procedures

- Language barriers causing confusion and frustration

- No system for updating refugees on wait times or changes

- Spread of misinformation leading to panic or unrest

5. **Minimal Throughput:**

- Assuming 2-hour round trips (due to inefficiencies and delays)

- Only 15 out of 20 trains operational due to breakdowns

- 15 trains x 800 passengers (below capacity) x 24 hours / 2 hours = 144,000 passengers per day

- At this rate, it would take about 6,250 days (roughly 17 years) to transport all 900 million refugees

**Factors contributing to worst case:**

- Lack of funding and resources

- Untrained or insufficient staff

- Uncooperative or desperate refugees leading to chaos

- Severe weather conditions disrupting operations

- Political instability or interference

- Infrastructure failures (e.g., power outages, track damage)

In reality, the actual scenario would likely fall somewhere between these extremes. The key to success would be identifying the most critical factors and focusing resources on optimizing those aspects of the operation.

2. Explain in computing concept the parts on a phone that determines the efficiency and optimization for data storage , data transfer and connectivity.

**Answers;**

A smartphone's efficiency in data storage, transfer, and connectivity is determined by a combination of hardware components and software optimizations. The main areas of focus are:

1. Storage: SSD/eMMC, file systems, and compression

2. Data Transfer: Bus architecture, processor, and I/O controllers

3. Connectivity: Modem, Wi-Fi, Bluetooth, and antenna design

Key optimizations include power management, cache optimization, data compression, and parallel processing.

**Detailed Elaboration:**

1. Storage Efficiency:

- Modern smartphones use NAND flash memory in the form of UFS (Universal Flash Storage) or NVMe.

- UFS 3.1 can achieve read speeds up to 2100 MB/s and write speeds up to 1200 MB/s.

- File systems like F2FS (Flash-Friendly File System) are optimized for NAND flash, reducing write amplification and improving longevity.

- Compression algorithms like zlib for general data and HEIF for images can significantly reduce storage requirements.

2. **Data Transfer Optimization:**

- Internal data transfer relies on high-speed buses like LPDDR5, which can achieve speeds up to 6400 Mbps.

- Modern smartphone SoCs (System on Chip) use big.LITTLE architecture, combining high-performance cores with energy-efficient cores for optimal performance and battery life.

- DMA (Direct Memory Access) controllers allow peripherals to transfer data without CPU intervention, improving efficiency.

3. **Connectivity Enhancements:**

- 5G modems can achieve theoretical speeds up to 20 Gbps, though real-world speeds are typically lower.

- Wi-Fi 6 (802.11ax) uses OFDMA and MU-MIMO to efficiently handle multiple devices, achieving speeds up to 9.6 Gbps in ideal conditions.

- Bluetooth 5.2 introduces LE Audio, improving audio quality and battery life for wireless audio devices.

- Beamforming technology in modern antennas can focus Wi-Fi signals towards specific devices, improving range and throughput.

4. **Power Management:**

- Dynamic Voltage and Frequency Scaling (DVFS) adjusts CPU clock speeds and voltage in real-time based on workload.

- Big.LITTLE architecture can achieve up to 75% energy savings compared to traditional architectures.

5. **Memory Management:**

- LPDDR5 RAM in modern smartphones can achieve data rates up to 6400 Mbps, significantly faster than previous generations.

- Intelligent memory compression techniques can effectively increase available RAM by 10-20%.

6. **Hardware Acceleration:**

- Neural Processing Units (NPUs) can perform AI tasks up to 100 times faster than CPUs while using less power.

- Image Signal Processors (ISPs) can process complex computational photography algorithms in real-time, enabling features like night mode and HDR.

7. **Software Optimization**:

- Just-In-Time (JIT) compilation in Android can improve app performance by 30-50% compared to interpreted code.

- iOS's LLVM compiler uses whole-program optimization to improve code efficiency across the entire operating system.

These advancements work in concert to provide the high performance and efficiency expected in modern smartphones. The continuous improvement in these areas drives the evolution of smartphone capabilities, enabling more complex applications and better user experiences while managing power consumption and heat generation.

3.In computing concepts explain how the hardware of a server and a phone can process 1 billion transactions and show the metrics.

**Answer:**

1. **Transaction Processing Capability:**

Server: Can process 1 billion transactions in approximately 15-20 minutes.

Smartphone: Would require several hours, potentially 8-12 hours or more.

Explanation: A high-end server with multiple CPU sockets, each containing 64+ cores running at 3.5+ GHz, can handle over 1,000,000 transactions per second (TPS). This is due to its massive parallelism and high clock speeds. In contrast, a smartphone's SoC, typically with 8 cores in a big.LITTLE configuration running at up to 3 GHz, might manage 10,000-50,000 TPS. The server's superior cooling allows it to maintain peak performance, while the phone would throttle due to heat buildup.

2. **Memory Handling:**

Server: Utilizes 1-4 TB of ECC DDR4/DDR5 RAM with bandwidth up to 200 GB/s per socket.

Smartphone: Uses 8-16 GB of LPDDR5 RAM with bandwidth around 30-60 GB/s.

**Explanation:** The server's vast memory capacity allows it to hold large datasets in RAM, drastically reducing the need for slower storage access. Its higher bandwidth enables faster data retrieval for processing. The phone's limited RAM necessitates more frequent access to slower flash storage, increasing latency.

3. **Storage Performance:**

Server: Achieves 1,000,000+ IOPS with NVMe SSDs in RAID configuration.

Smartphone: Reaches 100,000-200,000 IOPS with UFS 3.1 storage.

**Explanation**: The server's storage system is designed for high-volume, random access patterns typical in transaction processing. Its higher IOPS rating means it can handle more simultaneous storage operations, crucial for managing a large transaction volume. The phone's storage, while fast for a mobile device, is more limited in handling concurrent operations.

4. **Network Throughput:**

Server: Capable of 100 Gbps with high-end network interfaces.

Smartphone: Up to 20 Gbps with 5G, or 9.6 Gbps with Wi-Fi 6.

**Explanation:** In a real-world scenario processing 1 billion transactions, network capacity is crucial for data ingress/egress. The server's superior network throughput allows it to receive and send data much faster, which is essential when dealing with distributed systems or cloud-based transactions. The phone's lower throughput becomes a bottleneck for high-volume processing.

5. Power and Thermal Considerations:

Server: Consumes 200-500W per CPU, maintains performance at 70-80°C with active cooling.

Smartphone: Uses 2-5W for the entire SoC, throttles performance to stay under 45-50°C.

**Explanation:** The server's higher power consumption allows for sustained high performance, crucial for processing large transaction volumes quickly. Its sophisticated cooling systems (often liquid cooling in high-performance scenarios) maintain optimal operating temperatures. The phone's strict power and thermal limits force it to reduce performance over time to prevent overheating, significantly impacting its ability to process transactions consistently.

6. **Real-world Application:**

In practice, processing 1 billion transactions would typically be distributed across multiple servers in a data center or cloud environment. This distribution allows for even higher throughput and redundancy. Smartphones, due to their hardware limitations, would generally act as endpoints for initiating or displaying the results of transactions, rather than processing them entirely on-device.

The phone might handle local caching, user authentication, and data encryption/decryption for a subset of transactions, while offloading the bulk of the processing to server infrastructure. This architecture leverages the strengths of both devices: the phone's mobility and user interface capabilities, and the server's raw processing power and ability to handle high-volume data operations.

4. Based on such question 3, taking into account mobile transactions, how can a phone / server process 1 billion transactions and show the metrics of how it will process such

**Answer;**

Scenario: Processing 1 billion mobile payment transactions

1. Phone's Role (Transaction Initiation):

The phone's primary function is to initiate transactions, handle user interaction, and manage local security.

Metrics for Phone:

a) Transaction Initiation Rate:

- Peak: 100 transactions per second (TPS)

- Sustained: 50 TPS

b) Local Processing Time:

- User authentication: 100-300 ms

- Encryption: 5-10 ms per transaction

- UI rendering: 16-33 ms (60-120 fps)

c) Network Performance:

- 5G latency: 10-20 ms

- 5G throughput: Up to 1 Gbps (realistically 100-300 Mbps)

d) Battery Impact:

- Screen on time: 300-400 mAh per hour

- Network usage: 100-150 mAh per hour of active use

e) Storage I/O:

- Read speed: 1000-1500 MB/s (UFS 3.1)

- Write speed: 600-800 MB/s (UFS 3.1)

f) Memory Usage:

- Active app memory: 150-300 MB

- Background services: 50-100 MB

2. Server's Role (Transaction Processing):

The server handles the bulk of transaction processing, including validation, database operations, and inter-bank communications.

Metrics for Server (Cluster):

a) Transaction Processing Rate:

- Peak: 1,000,000 TPS

- Sustained: 500,000 TPS

b) Processing Time per Transaction:

- Validation: 0.5-1 ms

- Database operation: 1-5 ms

- Inter-bank communication: 10-50 ms

c) Network Performance:

- Throughput: 100 Gbps

- Latency within data center: < 0.5 ms

d) Storage Performance:

- Read IOPS: 1,000,000

- Write IOPS: 500,000

- Latency: < 0.1 ms

e) Memory Operations:

- Bandwidth: 200 GB/s per socket

- Capacity: 2-4 TB per server

f) CPU Utilization:

- Average: 60-70%

- Peak: 85-95%

Processing 1 Billion Transactions:

1. Time to Complete:

- Total time: Approximately 33 minutes (at sustained 500,000 TPS)

2. Data Transfer:

- Assuming 1 KB per transaction:

Total data: 1 TB

- Server network utilization: 4-5% of 100 Gbps capacity

3. Server Cluster Size:

- Assuming 50,000 TPS per server:

20 servers to handle 1,000,000 peak TPS

4. Database Operations:

- 1 billion write operations

- Several billion read operations (for validation, balance checks, etc.)

5. Phone-side Metrics (per device):

- Assuming 100 transactions per user:

10 million unique users

- Average time per transaction on phone: 3-5 seconds

- Total active time per user: 5-8 minutes

6. Energy Consumption:

- Server cluster: Approximately 100-150 kWh

- Per phone: 1-2% battery life per 100 transactions

7. Scaling Considerations:

- Horizontal scaling: Add more servers to increase capacity

- Vertical scaling: Upgrade server hardware for better per-server performance

8. Reliability Metrics:

- Expected uptime: 99.999% (5 minutes downtime per year)

- Transaction success rate: 99.99% (100 failed transactions per million)

9. Security Overhead:

- Encryption/Decryption: 0.1-0.2 ms per transaction

- Fraud detection algorithms: 1-5 ms per transaction

10. Geographical Distribution:

- Assuming global distribution:

Avg. server to phone latency: 50-100 ms

- Content Delivery Network (CDN) usage for static content:

Reduces latency by 30-50% for UI elements

This scenario demonstrates the distributed nature of processing 1 billion mobile transactions. While the servers handle the bulk of the processing, the phones play a crucial role in initiating transactions and managing the user interface. The system's ability to handle this volume of transactions relies heavily on the server infrastructure, with phones acting as edge devices in the larger network.