Lab #2 Independent Study

Course: CET3126C

Student: Gabriel Licona

Professor: Lester Suarez

RISC vs. CISC, Compiled vs. Interpreted, and 32-bit vs. 64-bit Systems

RISC vs. CISC

RISC (Reduced Instruction Set Computer):

- Uses a small, simple set of instructions.
- Most instructions are designed to execute in one clock cycle.
- Focuses on efficiency, pipelining, and parallel execution.
- Relies more on the compiler to break down complex tasks into smaller operations.
- Example architectures: MIPS, ARM, RISC-V.

CISC (Complex Instruction Set Computer):

- Provides a large set of complex instructions, some performing multiple steps in one instruction.
- Designed to reduce the number of instructions per program, but each may take multiple cycles.
- Pushes complexity into the hardware, making programming simpler but pipelining harder.
- Example architecture: x86 (Intel, AMD).

Summary: RISC → simpler hardware, faster pipelines. CISC → complex hardware, fewer but heavier instructions.

Compiled vs. Interpreted Languages

Compiled Languages:

- Source code is translated ahead of time into machine code by a compiler.
- Runs directly on hardware, resulting in faster execution.
- Examples: C, C++, Rust.

Interpreted Languages:

- Source code is translated line by line at runtime by an interpreter.
- Easier to debug and portable, but slower.
- Examples: Python, JavaScript.

Hybrid Approaches:

- Some languages (e.g., Java, C#) are compiled into bytecode that runs on a virtual machine.
- Just-in-time (JIT) compilation can combine the benefits of both methods.

Summary: Compiled → faster, optimized, less flexible. Interpreted → more flexible, slower execution.

32-bit vs. 64-bit Architectures

32-bit Systems:

- Registers, addresses, and data paths are 32 bits wide.
- Maximum directly addressable memory: 4 GB (2^{32} bytes).
- Common in older PCs and embedded systems.

64-bit Systems:

- Registers and addresses are 64 bits wide.
- Theoretical memory limit: 16 exabytes (2^{64} bytes), though OS limits are much smaller.
- Supports larger applications, improved performance, and wider arithmetic registers.

Comparison: 32-bit → limited to 4 GB RAM, smaller pointer size. 64-bit → vastly larger address space, more powerful but slightly higher memory usage.

Final Takeaway

- **RISC vs. CISC**: hardware design philosophy (simple vs. complex instructions).
- **Compiled vs. Interpreted**: how source code is executed (ahead of time vs. runtime).
- **32-bit vs. 64-bit**: system word size (memory and performance differences).