

Seminários de Python

Computadores & Sistemas Operacionais

PhD Flavio Lichtenstein

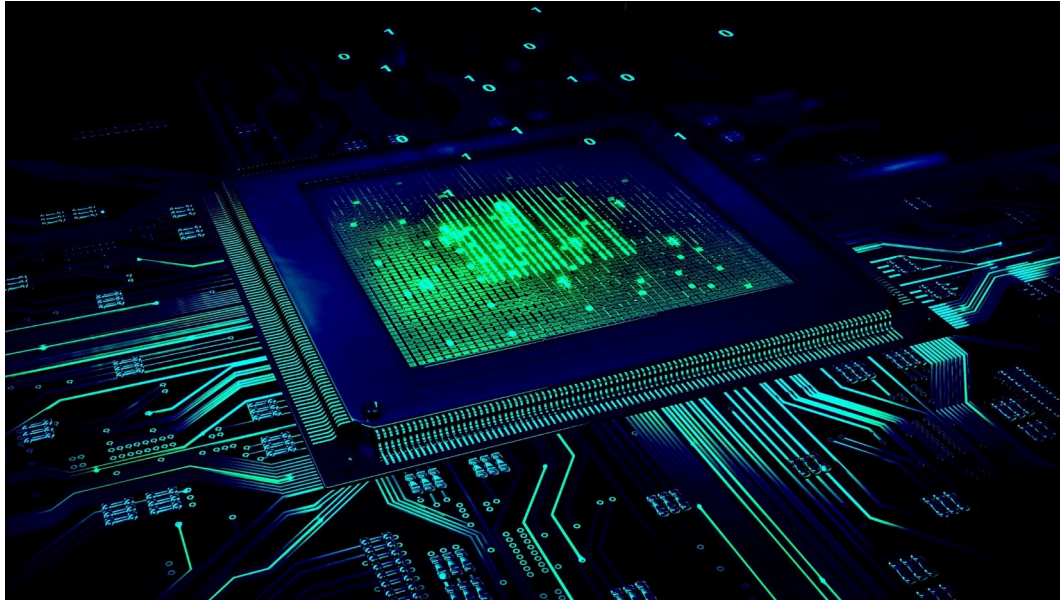
Bioinformatics, Systems Biology, and Biostatistics

CENTD – Instituto Butantan

setembro/2020

Estrutura de um computador

Imagem do chip principal de um computador
CPU: central processor unit



Hoje, possuí vários 'cores' ~ 8 CPU's em uma só
continuará evoluindo

Escalas importantes

1 Byte = 8 bits

1 bit = True / False, On / Off

1 Byte = 2^8 valores lógicos possíveis = 0 até 255

1 KB = 1 Kilo byte ~ 1024 bytes

1 MB = 1 Mega byte ~ 1.000.000 bytes = 1 milhão de bytes

1 GB = 1 Giga byte ~ 1.000.000.000 bytes = 1 bilhão de bytes

1 TB = 1 Tera byte ~ 10^{12} = 1 trila de bytes

Se uma página contém ~ 80 linhas x 100 char = 8 Kbytes (sem fotos)

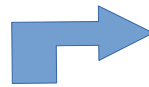
Se um livro tem 400 páginas ~ $400 * 8 \text{ Kbytes} = 3.2 \text{ Mbytes}$

Então um disco de 1 TB pode conter: $10^{12} / 3 * 10^6 \sim 3$ milhões de livros

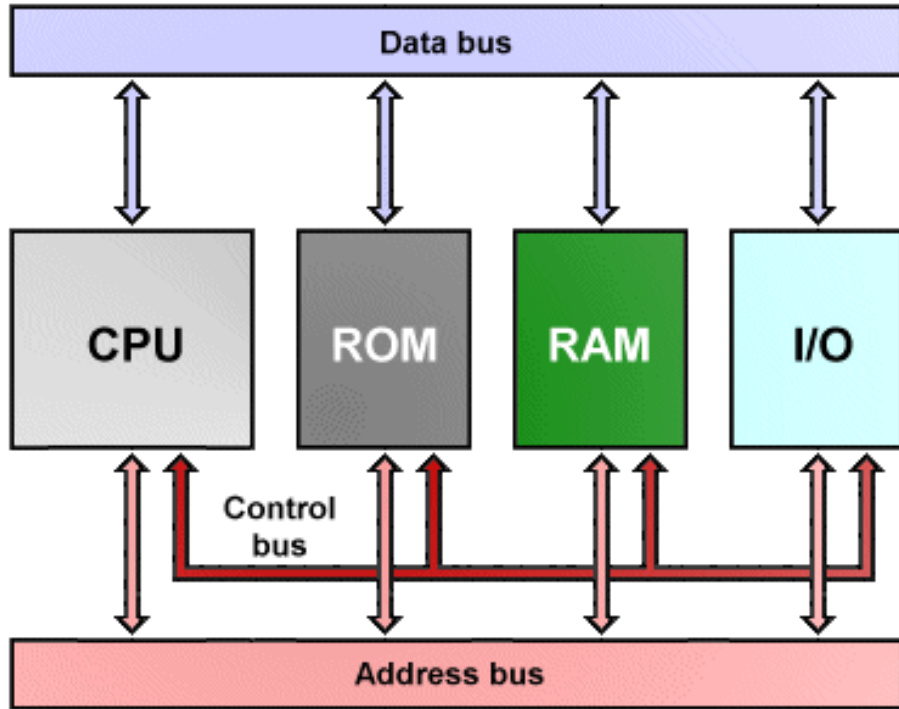
Principais elementos de um computador

Como é dividido um computador:

- CPU (CISC ou RISC) + cache
- Memória (Gbytes / Tbytes, tempo de acesso ~ 10 ns)
- Placas de vídeo (com GPUs – graphic processor units)
- Discos de armazenamento:
 - Magnéticos (móveis) – são os HD (hard drives)
 - SSD (solid state drives)
- Diversas interfaces como USB



As diversas partes conversando entre si ...



- Data bus: passagem de dados
- Address bus: passagem de endereços
- CPU – unidade central
- ROM – read-only memory (boot ~ liga)
- RAM – memória (randômica)
- I/O – interfaces de entrada e saída

RISC: Reduced Instruction Set Computer

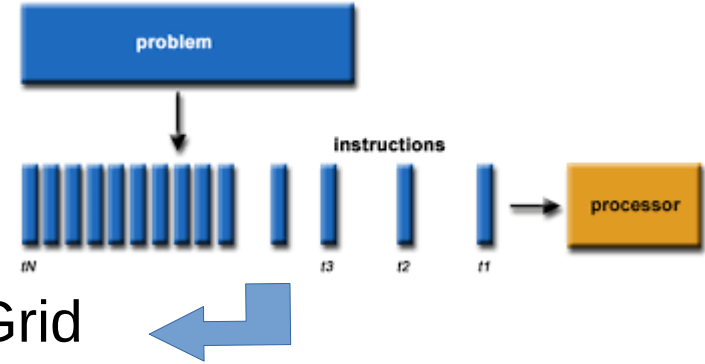
CISC: Complex Instruction Set Computer

CARACTERÍSTICAS	CISC	RISC
INSTRUÇÕES POR CICLOS	<i>Instruções complexas executadas em vários ciclos</i>	<i>Instruções simples executadas em um ciclo</i>
ACESSO À MEMÓRIA	<i>Qualquer instrução pode referenciar a memória</i>	<i>Apenas operações LOAD/STORE em memória</i>
PIPELINE	<i>Pouco ou nenhum pipeline</i>	<i>Uso Intenso de pipeline</i>
EXECUÇÃO DAS INSTRUÇÕES	<i>Instruções executadas pelo hardware</i>	<i>Instruções interpretadas pelo próprio programa</i>
FORMATO DAS INSTRUÇÕES	<i>Instruções com formato variável</i>	<i>Instruções com formato fixo</i>
QUANTIDADE DE INSTRUÇÕES E MODOS DE ENDEREÇAMENTO	<i>Várias instruções e modos de endereçamento</i>	<i>Poucas instruções e modos de endereçamento</i>
COMPLEXIDADES DO SISTEMA	<i>Complexidade está no microprograma</i>	<i>Complexidade está no compilador</i>
REGISTRADORES	<i>Conjunto de registradores único</i>	<i>Múltiplos conjuntos de registradores</i>

Hoje (2020) as arquiteturas trazem um híbrido entre RISC e CISC

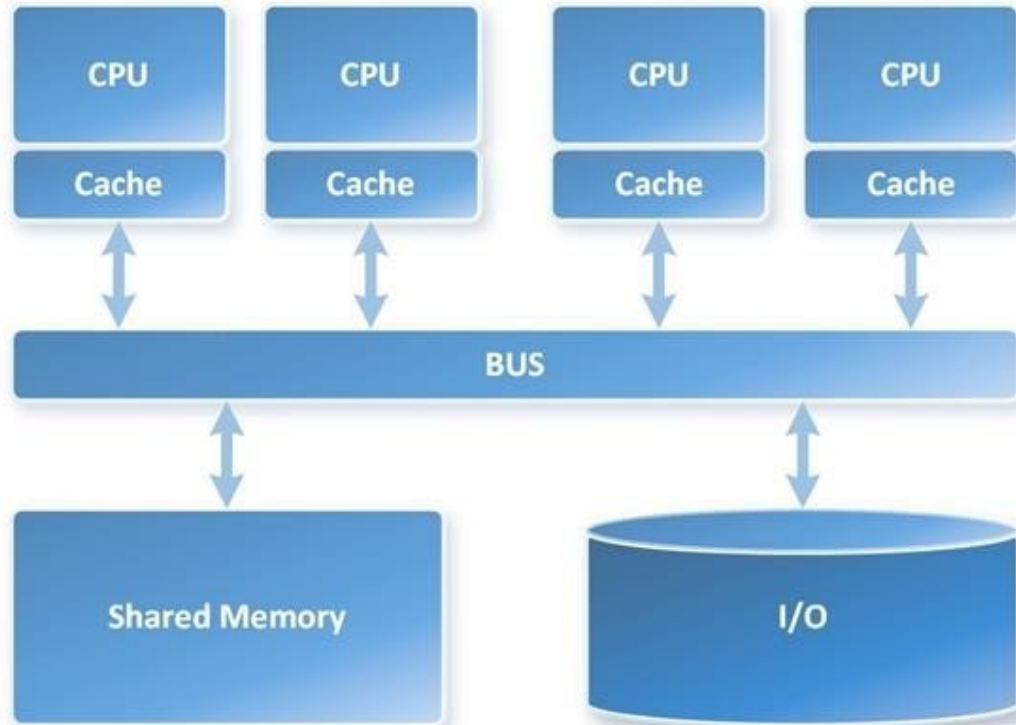
Limitações

Exemplos de Limitações



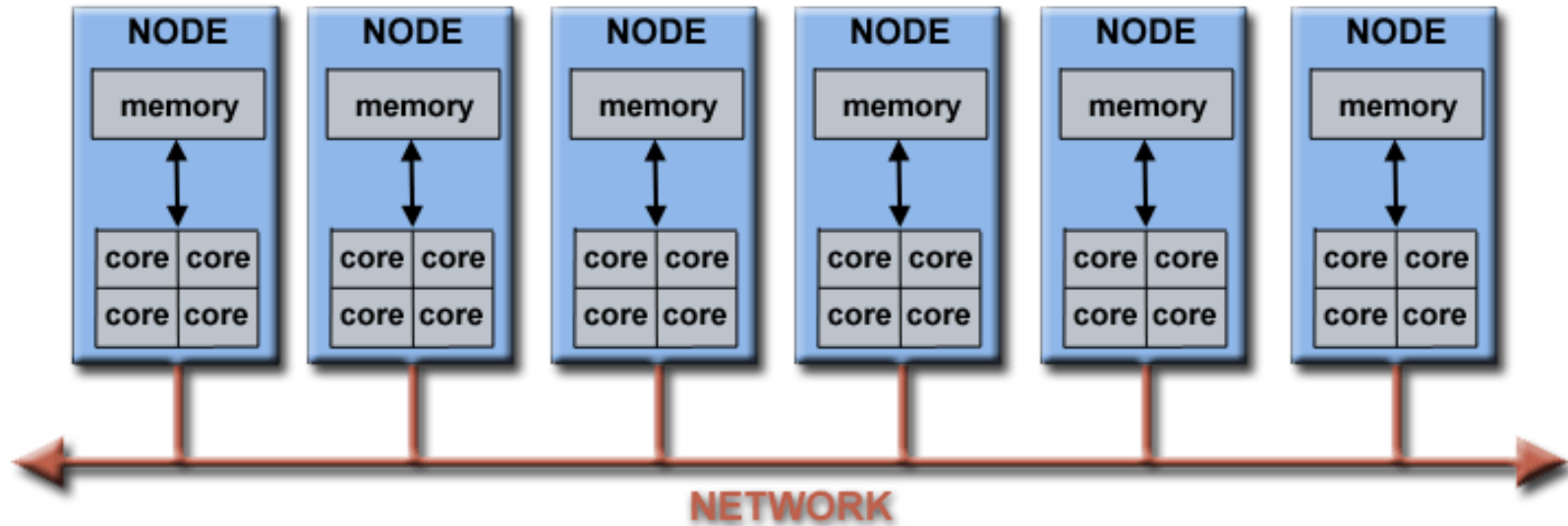
- Bottleneck (von Neuman) x Paralelismo x Grid
- Acesso a memória:
 - RAM: 10 ns
 - Disco rígido: 5 a 10 milisegundos
 - SSD: 35 a 100 microsegundos
- Espaço em memória
- Espaço em disco (big-data)

Servidor: muitas CPUs + GPUs



HPC: high performance computing

Google (GCP), Amazon (AWS), Microsoft (Azure)



GRID computing



Sistemas Operacionais

Principais sistemas operacionais

- Windows (antiga meta – podia clonar / ser mais fácil e barato)
- MacOS (antiga meta – melhor interface gráfica + agilidade)
- Linux (antigamente: difícil de aprender e comandar):
 - Muito fácil
 - Interface gráfica similar ao Mac e Windows
 - Mais rápido SO
 - A maioria das ferramentas são de livre acesso
 - Têm várias distribuições: Fedora, Ubuntu, Mint, openSuse, Red Hat ...



Obrigado

Dúvidas?

PhD Flavio Lichtenstein

Bioinformatics, Systems Biology, and Biostatistics

CENTD – Instituto Butantan

setembro/2020