

Pontifícia Universidade Católica de Goiás  
Goiânia, 24 de setembro de 2017.  
Aluno: Wanderson César da Silva  
Matricula: 2017101200366

## **Organização de Computadores: Arquitetura Von Neumann**

Nos primórdios da Computação os programas eram desenvolvidos, executados e apresentavam os resultados imediatamente e não eram armazenados.

O conceito de programa armazenado surgiu da necessidade de armazenar programas em um computador, pois, até então, ainda não havia formas de armazenar programas em um computador. John Von Neumann e outros pesquisadores descobriram que, utilizando dispositivos de memória em formas de linha de retardo de mercúrio, poderiam armazenar instruções de programas.

O conceito de programa armazenado surgiu durante o projeto EDVAC e John Von Neumann acabou recebendo a maior parte dos créditos pela definição, mas a verdade é que várias pessoas participaram da construção. John Von Neumann ganhou mais destaque que os outros por ter escrito um relatório sobre o EDVAC, com 101 páginas, descrevendo formalmente o conceito, criando assim uma espécie de guia para a construção de computadores.

A proposta inicial de um computador de programa armazenado, relatada pelo próprio Von Neumann, era dividida em cinco partes:

- Central aritmética;
- Central de controle;
- Memória;
- Gravação;
- Entrada e saída;

A arquitetura é composta por três grandes sistemas de hardware: sistema central de processamento, sistema de memória e sistema de entrada/saída. A arquitetura tem a capacidade de executar instruções sequencialmente e possui apenas um caminho de dados entre a UCP e a memória principal, o que é chamado de Gargalo de von Neumann. Atualmente, é claro, isso não existe mais e sofreu uma grande evolução.

O ciclo de execução diz respeito à forma com que uma instrução é executada pela CPU, os caminhos que ela deve fazer e as unidades funcionais pelas quais ela passará, ou seja, descreve como o hardware trabalha com as instruções que serão executadas. O ciclo de execução de Von Neumann é simples e normalmente é citado como BUSCA-DECODIFICAÇÃO-EXECUÇÃO, pois é exatamente isto o que ocorre: busca uma instrução, a decodifica e a executa.

