A close-up photograph of a person's hand typing on a laptop keyboard. The scene is dimly lit, with the primary light source being the backlit keys of the laptop, which cast a soft glow on the hand and the surrounding area. The background is dark and out of focus. The text 'Como um programa é executado internamente no computador?' is overlaid on the left side of the image in a large, white, sans-serif font.

**Como um programa é  
executado  
internamente no  
computador?**

# O processo de execução de um programa pelo computador

- O Ciclo de Vida de um Programa
- A Criação do Código
- A Compilação e Interpretação
- O Carregamento do Programa na Memória
- A Execução do Programa pela CPU
- A Interação com o Sistema Operacional e Recursos
- O Encerramento e Liberação de Recursos

# O CICLO DE VIDA DE UM PROGRAMA

Um programa é um conjunto de instruções escritas em uma linguagem de programação, que precisam ser executadas em uma máquina.

O seu ciclo de execução é desenvolvimento, tradução (compilação/interpretação), execução e encerramento.

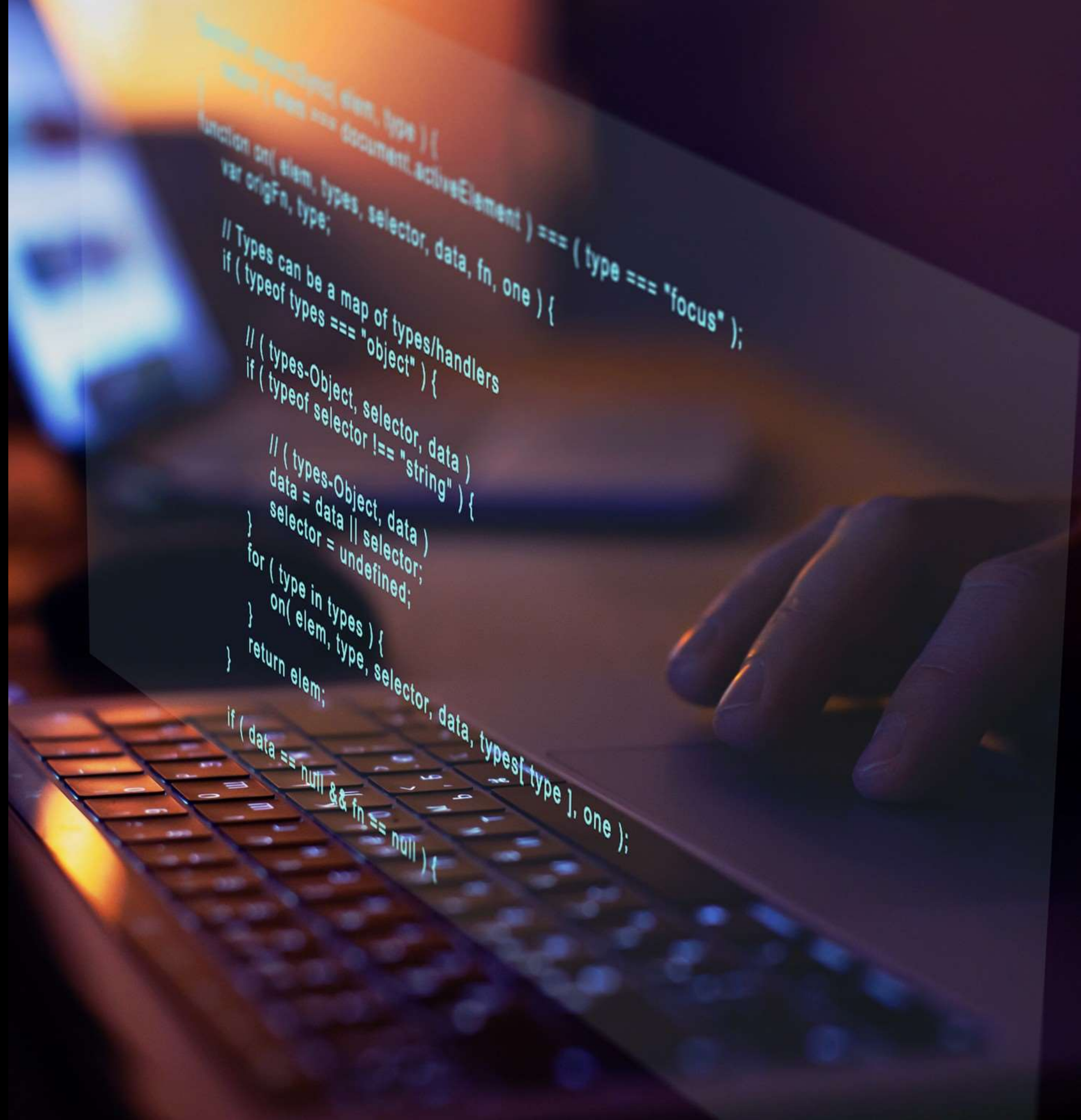




# CRIAÇÃO DO CÓDIGO

Linguagens de alto nível como C, Java, e Python são usadas para facilitar o desenvolvimento humano.

A conversão do código-fonte para linguagem de máquina é necessária, pois a CPU só entende instruções em código binário.



# COMPILAÇÃO E INTERPRETAÇÃO

- Compilação: Linguagens como C e C++ passam por um compilador que transforma o código-fonte em código executável.
- Interpretação: Linguagens como Python são interpretadas linha a linha, tornando o processo mais flexível, mas geralmente mais lento.
- Existem também linguagens híbridas, como Java e C# que são compiladas em uma primeira etapa para um bytecode, que é então interpretado por uma máquina virtual.

```
        'role_id' => $role_id,
        'resource_id' => $resource_id
    );
    if ( $this->rule_exists( $resource_details ) {
        if ( $access == false ) {
            // Remove the rule as there is c
            $details['access'] = !$access;
            $this->_sql->delete( 'acl_rules'
        } else {
            // Update the rule with the new
            $this->_sql->update( 'acl_rules'
        }
    }
    foreach( $this->rules as $key=>$rule ) {
        if ( $details['role_id'] == $rule['role_id'] ) {
            if ( $access == false ) {
                unset( $this->rules[ $key ]
            } else {
                $this->rules[ $key ]['ac
            }
        }
    }
}
```



# CARREGAMENTO DO PROGRAMA NA MEMÓRIA

O sistema operacional carrega o programa na memória RAM para que a CPU possa acessá-lo.

Um programa é geralmente dividido em segmentos, que facilitam a organização e a segurança dos dados e instruções, são eles:

- Segmento de Código : Contém as instruções de máquina que a CPU executará.
- Segmento de Dados: Armazena variáveis globais e estáticas. Essas variáveis são alocadas quando o programa inicia e permanecem na memória até que ele termine.
- Segmento de Heap: Uma área dinâmica de memória onde variáveis que requerem alocação durante o tempo de execução são armazenadas
- Segmento de Stack (Pilha): Usado para armazenar variáveis locais e manter o controle de chamadas de função.

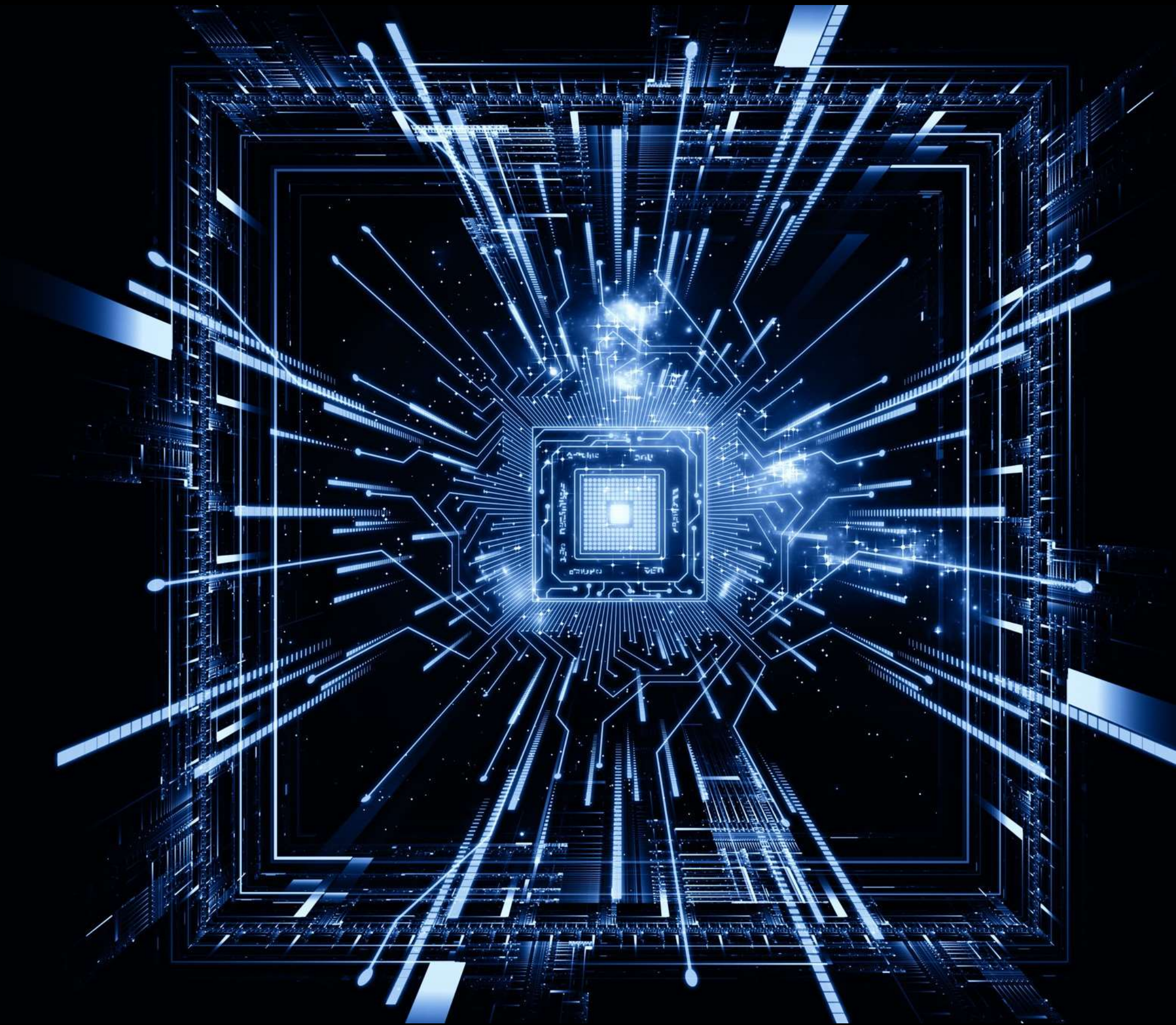




# EXECUÇÃO DO PROGRAMA PELA CPU

É realizado um ciclo de busca-decodificação-execução. Nele, a CPU busca uma instrução, decodifica seu significado e executa.

A Unidade Lógica e Aritmética (ALU) realiza operações matemáticas ou lógicas, enquanto a Unidade de Controle coordena todas as operações para garantir que cada etapa do processamento aconteça na ordem correta.





## ENCERRAMENTO E LIBERAÇÃO DE RECURSOS

Após o término, o sistema operacional libera memória, fecha arquivos e remove o programa da lista de processos. A CPU recebe uma instrução de término que indica ao sistema operacional que o processo chegou ao fim, desencadeando o processo de liberação dos recursos alocados.

Toda a memória alocada ao programa, tanto na RAM quanto na memória cache, é liberada para uso por outros programas e processos, assim como arquivos e conexões são fechados e os outros recursos do sistema são liberados para outros processos.





# CONCLUSÃO

A execução de um programa envolve várias etapas, desde a compilação ou interpretação até a execução e liberação de recursos, contando com a coordenação da CPU, memória e sistema operacional para transformar código em ações.

CPU, memória e sistema operacional trabalham juntos para garantir a eficiência, segurança e multitarefa, permitindo que vários programas operem simultaneamente sem interferências e compreender esse processo ajuda a otimizar o desenvolvimento de software e identificar problemas de desempenho.



# REFERÊNCIAS BIBLIOGRÁFICAS

<https://www.geeksforgeeks.org>

<https://www.khanacademy.org/computing/computer-programming>

[https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First\\_steps/What\\_is\\_JavaScript](https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/What_is_JavaScript)

<https://csfieldguide.org.nz>

<https://computer.howstuffworks.com/ram.htm>

<https://www.techopedia.com/definition/14862/fetch-decode-execute-cycle>

<https://developer.ibm.com/articles/l-system-calls-and-library-wrappers/>

<https://www.redhat.com/sysadmin/memory-management-linux>