

The background features a collection of colorful 3D blocks in various colors (purple, blue, yellow, green, orange, pink, grey) arranged in a scattered pattern. A large, faint key is visible in the background, with its handle pointing towards the bottom left and its head pointing towards the top right. The title text is overlaid on the left side of the image.

Leitura complementar

Introdução à Lógica

- ✓ Conceitos de Lógica;
- ✓ Programa;
- ✓ Tradutores;
- ✓ Tipos de programação.

Lógica

Lógica é a maneira de raciocinar particular a um indivíduo ou a um grupo, gerando uma sequência coerente, regular e necessária de acontecimentos ou métodos, com a finalidade de obter uma solução prática e eficaz para um problema.

Se observarmos o nosso dia a dia, usamos a lógica frequentemente, em nossas casas, no trabalho, nas compras, no trânsito, ou seja, em tudo. Como exemplo, podemos citar a lógica que fazemos desde a hora que acordamos até quando chegamos ao trabalho:

- **Acordar no horário programado**
- **Tomar banho**
- **Vestir a roupa adequada para trabalhar**
- **Tomar café**
- **Sair de casa**
- **Chegar ao local de trabalho dentro do horário previsto**

Como podemos observar, já acordamos pensando no que temos que fazer, portanto, já acordamos utilizando a lógica e, se a lógica inicial não for a ideal, nós a modificamos para que ela nos leve à melhor solução do problema. Por exemplo, se alguém já estiver tomando banho, então, podemos tomar café antes para que não nos atrasemos para o trabalho.

- **Acordar no horário programado**
- **Verificar se o banheiro está livre**
- **Se sim:**
 - Tomar banho
 - Vestir a roupa adequada para trabalhar
 - Tomar café

- **Se não:**
 - Tomar café
 - Tomar banho
 - Vestir a roupa adequada para trabalhar
- **Sair de casa**
- **Chegar ao local de trabalho dentro do horário previsto**

Existem diversos tipos de exercícios e jogos que servem para desenvolver o raciocínio lógico. Vamos começar com um exercício de lógica mais comum e depois focaremos a programação.

- **Exemplo**

Vamos supor que exista uma caixa com 15 bolas, sendo 5 verdes, 5 amarelas e 5 azuis. Quantas bolas devem ser retiradas da caixa, sem olhar, para termos certeza de que saíram 2 bolas de cor azul?

- **Resposta**
 - 5 bolas verdes
 - 5 bolas amarelas
 - 2 bolas azuis

Total: 12 bolas

- **Conclusão**

Temos que ter certeza que saíram 2 bolas azuis, portanto consideramos a pior hipótese no caso de retirar bolas aleatoriamente. Foram retiradas as 10 bolas entre verdes e amarelas e depois saíram as bolas azuis.

Programa

Programa é uma sequência lógica de instruções escritas em uma linguagem de programação, para serem executadas passo a passo, com a finalidade de atingir um determinado objetivo.

Tipos de linguagem de programação

Uma **linguagem de programação** é um método padronizado para comunicar instruções para um computador.

As linguagens de programação podem ser de **baixo nível** e de **alto nível**.

As linguagens de baixo nível são aquelas capazes de compreender a arquitetura do computador e que utilizam somente instruções do processador. Exemplos: Linguagem de máquina e Assembly.

As linguagens de alto nível são aquelas com a escrita mais próxima da linguagem humana. Exemplos: Objective-C, C++, C#, Delphi, Java, VB, MATLAB e ASP.

Tradutores

Os **tradutores** foram criados para tornar mais fácil a interface entre o usuário e a máquina. Como já vimos anteriormente, as linguagens são divididas em baixo e alto nível, cada uma refletindo uma proximidade com a linguagem natural do usuário.

O computador só executa instruções em linguagem de máquina, a qual é composta por dígitos binários. Logo, para que o computador execute instruções escritas em linguagens com estruturas diferentes, é preciso que essas instruções sejam traduzidas para a linguagem de máquina.

O tipo da tradução depende da complexidade da estrutura das linguagens.

Tipos de tradutores

Existem três tipos básicos de tradutores: **Montador**, **Interpretador** e **Compilador**.

O **Montador** traduz a linguagem Assembly para a linguagem de máquina. Sua estrutura é relativamente simples e depende diretamente do processador utilizado, pois cada processador tem seu set de instruções característico.

Os outros tradutores são mais complexos, pois necessitam fazer análises mais sofisticadas da estrutura da linguagem para realizar a tradução.

O **Interpretador** realiza a tradução e a execução simultaneamente, não gerando o código-objeto (linguagem de máquina) em disco.

O **Compilador** é um programa que traduz uma linguagem de programação de alto nível para linguagem de máquina, gerando um código-objeto independente.

Tipos de Programação

O ponto de partida do processo de codificação é a montagem de uma tela de interface com o usuário. Montamos uma tela arrastando controles para uma janela. Esses controles são chamados de objetos.

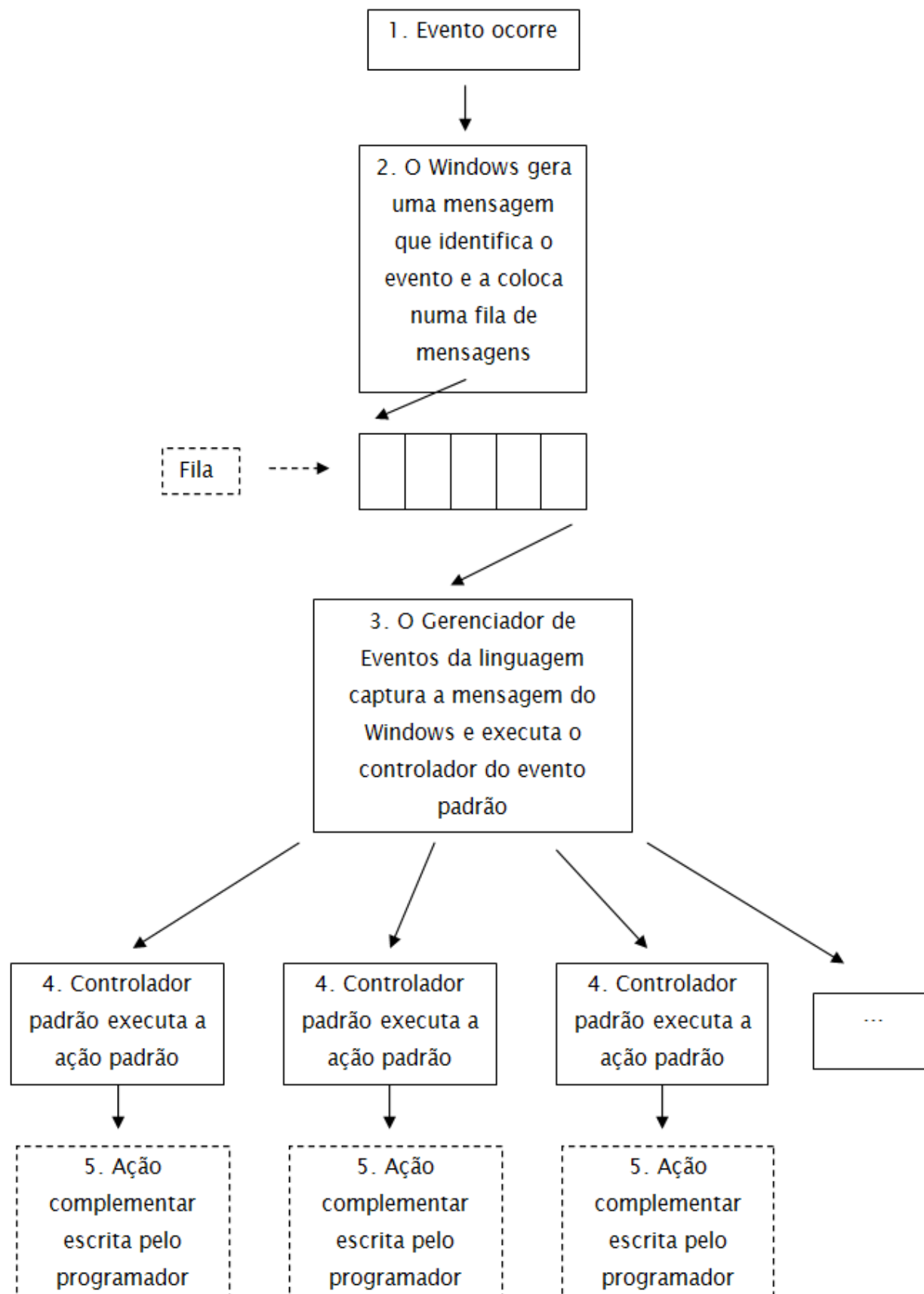
Em design-time (tempo de projeto), definimos as características (propriedades) iniciais desses objetos.

Em run-time (tempo de execução), além de alterarmos propriedades, podemos também chamar novas ações.

Realizamos tais tarefas na medida em que o usuário interage com a tela e seus objetos, disparando eventos. Capturamos tais eventos para que nosso código possa ser executado, por isso chamamos este estilo de **programação orientada a eventos**.

Na programação tradicional (**programação procedural**), toda a lógica é descrita em sequência e executada nesta ordem. As ações do usuário são “engessadas”, já que a ordem não pode ser alterada por ele.

Na **programação orientada a eventos**, nossa lógica é “quebrada” em vários módulos (procedimentos), que são executados na ordem determinada pelas ações do usuário.



A **programação orientada a objetos** se baseia na interação entre as unidades do software, chamadas de objetos. Nesse estilo de programação, são definidas diversas classes que determinam o comportamento e os estados de cada objeto.

Ocorrências de eventos

Um **evento** consiste em um fato que possa influenciar na execução de um programa. Sem a ocorrência de um evento, o programa que depende dele para ser executado ficaria em situação de espera eternamente.

Existem três tipos básicos de eventos:

- **Evento de Mouse**

Ocorre quando executamos qualquer tipo de ação com o mouse, como mover o mouse sobre um objeto, pressionar um dos botões do mouse sobre um objeto, soltar um dos botões do mouse, clicar (pressionar e soltar) com o botão esquerdo do mouse sobre um objeto ou aplicar um duplo-clique com o botão esquerdo do mouse sobre um objeto.

- **Evento de teclado**

Ocorre quando executamos qualquer tipo de ação com o teclado, como quando pressionamos uma tecla quando o foco está sobre um objeto, quando soltamos uma tecla quando o foco está sobre um objeto ou quando pressionamos e soltamos uma tecla quando o foco está sobre um objeto.

- **Evento de clock**

É gerado a pedido do programador, em intervalos de tempo regulares, cuja unidade de medida é milissegundos.