

SBVORIN: Organização e Recuperação da Informação

Aula 12: Algoritmos de Coleta de Lixo (*Garbage Collection*)

Bacharelado em Ciência da Computação
Prof. Dr. David Buzatto



INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
SÃO PAULO
Campus São João da Boa Vista

Introdução

- A Coleta de Lixo (*Garbage Collection* (GC)) consiste em uma forma automática para o gerenciamento de memória de um programa, livrando, de certa forma, o programador de ter que realizar essa tarefa explicitamente;
- A coleta de lixo normalmente usará uma porção significativa do tempo de processamento de um programa;
- Algumas linguagens que possuem coleta de lixo:
 - Java;
 - C#;
 - D;
 - Go;

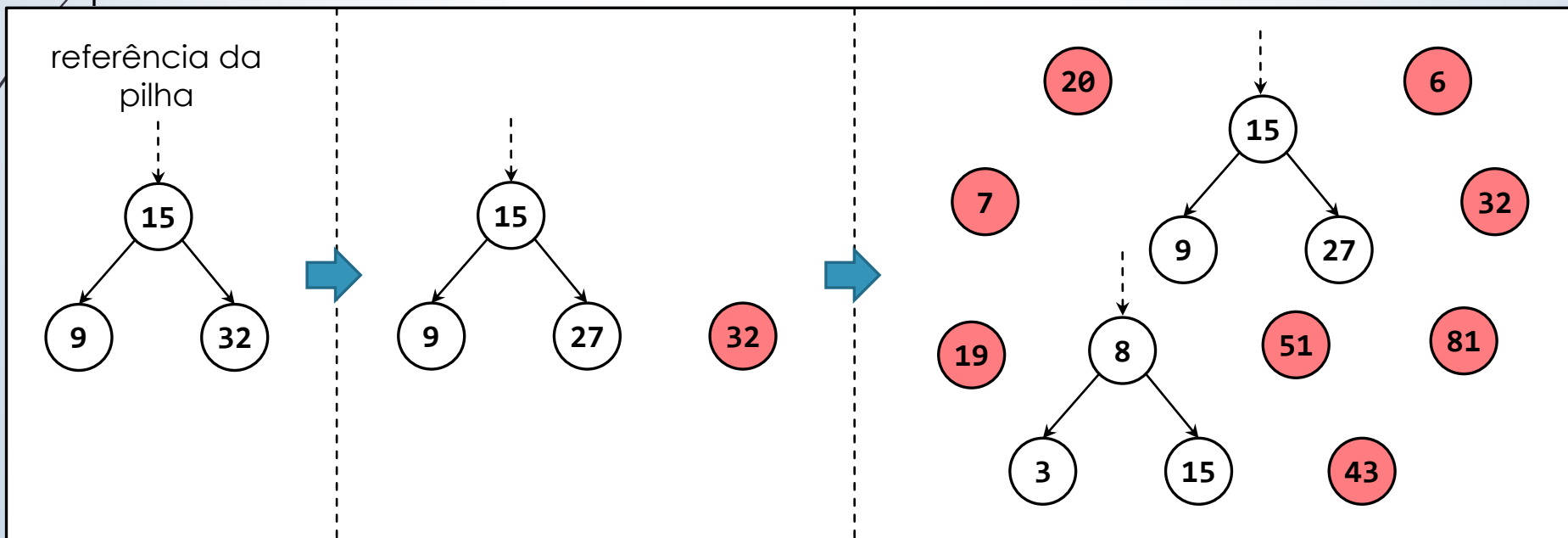
- ▶ Existem diversas abordagens/algoritmos para a execução da coleta de lixo. No contexto da plataforma Java, pode-se citar:
 - ▶ Serial Garbage Collector (Java 5 e 6);
 - ▶ Parallel Garbage Collector (Java 8);
 - ▶ Concurrent Mark Sweep (CMS) Collector;
 - ▶ G1 (Garbage First) Garbage Collector (Java 7);
 - ▶ Shenandoah Garbage Collector (Java 17 em diante com backport para Java 8 e 11).

- No contexto da linguagem de programação Java, o coletor de lixo liberará o espaço alocado à objetos que não possuam mais referências da pilha que apontem para os mesmos, ou seja, os objetos que ficam “soltos” no Heap são elegíveis para a coleta de lixo;

Coleta de Lixo *Mark & Sweep*

- A ideia básica do algoritmo *mark & sweep* (marcar e varrer) consiste em inicialmente percorrer todas as referências que não estão no heap, alcançando a partir dessas todos os objetos no heap e marcando-os. Após a passagem de marcação, percorre-se todos os objetos contidos no heap e todos aqueles que não foram marcados anteriormente são desalocados (varridos).

heap



➤ Garbage Collection:

➤ <https://www.youtube.com/watch?v=c32zXYAK7CI>

➤ Rust and RAII Memory Management:

➤ <https://www.youtube.com/watch?v=pTMvh6VzDI8>

SEDGEWICK, R.; WAYNE, K. **Algorithms**. 4. ed. Boston: Pearson Education, 2011. 955 p.

GOODRICH M. T.; TAMASSIA, R. **Estruturas de Dados & Algoritmos em Java**. Porto Alegre: Bookman, 2013. 700 p.

CORMEN, T. H.; LEISERSON, C. E.; RIVEST, R. L.; STEIN, C. **Algoritmos – Teoria e Prática**. 3. ed. São Paulo: GEN LTC, 2012. 1292 p.