

# Algoritmo de Impressão de Livro

Luis Gustavo Arseno Santana  
Thiago Carreira A. Nascimento  
Faculdade de Informática - PUCRS  
90619-900 - Porto Alegre - Brasil

28 de novembro de 2016

## Resumo

*O presente trabalho aborda a implementação de um algoritmo para impressão de livros. Basicamente, dado como entrada um arquivo contendo descrições de um livro (nome, seções, subseções, parágrafos), o algoritmo deve organizar a entrada numa estrutura de dados do tipo Árvore, gerando um arquivo de saída com a estrutura gerada. Tal implementação pode resumidamente ser caracterizada pelo seguinte: para cada item lido do arquivo, verifica sua relação estrutural com o antecessor (se filho do nodo raiz ou de um subnodo), alocando-o em sua respectiva posição hierárquica. Considera-se a raiz como título do livro, os capítulos como filhos da raiz, as seções como filhos dos capítulos, e, por fim, os parágrafos e subseções como filhos das seções. A implementação do algoritmo se faz em observação à estrutura de dados do tipo Árvore.*

## Introdução

De acordo com Goodrich (2013, cap 7), a estrutura de dados do tipo *Árvore* representa uma estrutura não-linear de computação, devido a sua estrutura organizacional, que contempla formas de relacionamento entre objetos para além do “antes” e “depois” das listas. Nesse sentido, a principal característica de tal estrutura de dados reside no seu tipo de relacionamento hierárquico, que é classificado por meio de noções como “acima” e “abaixo” ou “pai” e “filho”.

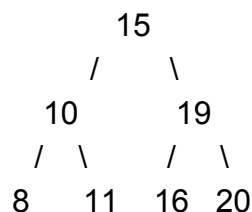
Talvez o exemplo mais simples de uma estrutura do tipo *Árvore* seja a representação da relação de descendência por meio de uma árvore genealógica: o primeiro progenitor é a *raiz*, e a partir desta são estabelecidas relações do tipo ‘pai-filho’. Tal *Árvore* poderia ser classificada como do tipo *genérica*, uma vez que, por exemplo, não há restrições em relação à sua forma organizacional, seja em termos da quantidade de filhos que cada *nodo* deve possuir ou em relação ao método de inserção dos dados. Como um exemplo alternativo, poder-se-ia citar uma

Árvore do tipo *Binária de Pesquisa*, na qual cada *nodo* possui no máximo dois filhos, sendo que a *raiz* “divide” entre “esquerda” e “direita” os elementos que estão numa ordem de precedência ou procedência. Assim, por exemplo, se se trata de uma Árvore de números inteiros que tem como *nodo* raiz o número 15, todos os números menores do que 15 serão alocados na parte “esquerda” da Árvore (nos *nodos* filhos da raiz que representam galhos) e os maiores do que 15 à sua “direita”. O mesmo princípio pode ser aplicado, por exemplo, a organização de nomes ou qualquer outro tipo de dado elementar.

Exemplo de uma Árvore genérica:



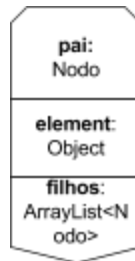
Exemplo de uma Árvore Binária de Pesquisa:



O algoritmo apresentado neste trabalho representa uma *Árvore Genérica*, uma vez que cada *nodo* (Capítulo, Seção, etc) pode ter vários filhos.

O método de inserção pode ser em linhas gerais descrito da seguinte forma: identifica-se o *Título* do livro, e o insere como *nodo raiz* da árvore; identifica-se o *Capítulo* do livro, e o insere como *filho* da raiz; identifica, para cada Capítulo, sua *Seção*, inserindo-a como *filho* do Capítulo, por fim, identifica-se cada *Parágrafo* (com sua respectiva quantidade de linhas) e *Subseção* inserindo-os como filhos da *Seção*.

A estrutura de dados da Árvore tem a seguinte configuração:



Ou seja, o pai de todo elemento é um Nodo, porém, o(s) filho(s) é um Lista de Nodos.

### **Requisitos**

Os requisitos estabelecidos para o presente trabalho são os seguintes:

1. Cada livro possui um título (que ocupa uma página inteira);
2. Cada livro pode possuir 0 ou mais capítulos;
3. Cada capítulo possui um título (que ocupa exatamente uma linha);
4. Cada capítulo inicia obrigatoriamente em uma nova página, sendo que a primeira linha corresponde ao título;
5. Cada capítulo pode ter 0 ou mais seções;
6. Cada capítulo também possui 0 ou mais parágrafos;
7. Cada seção possui um título (que ocupa exatamente uma linha);
8. Cada seção pode ter 0 ou mais subseções;
9. Cada seção possui 0 ou mais parágrafos;
10. Cada subseção possui um título (que ocupa exatamente uma linha);
11. Cada subseção também possui 0 ou mais parágrafos;
12. Cada parágrafo ocupa um determinado número de linhas, conforme especificado.

## Descrição da Solução

A solução apresentada consiste basicamente em gerar uma estrutura encadeada de *Árvore Genérica*, inserindo os elementos na ordem em que são lidos a partir do arquivo de entrada. O código foi implementado a partir de uma versão base disponibilizada no *Moodle* para este fim. Nele, constam métodos padrões para esse tipo de estrutura de dados, tal como adicionar elemento raiz, retornar o elemento raiz, retornar o pai de um determinado elemento, retornar o elemento de um determinado nodo, verificar se a árvore está vazia, bem retornar o tamanho e o nível da árvore.

O tipo de saída requerida para a estrutura de dados é similar ao tipo de leitura/encaminhamento da *Árvore* designado como pré fixado (cf. GOODRICH, 2013, *cap. 7.2.2*). Esse tipo de método organiza a estrutura de dados *Árvore* de acordo com o seguinte princípio: visita-se o *nodo* raiz, então seu filho mais à esquerda (no caso, o primeiro Capítulo), e os filhos destes respeitando a mesma sequência (da esquerda para direita). Visitado todos os *subnodos* no primeiro filho da raiz, segue-se o mesmo princípio, visitando os outros filhos. Assim, por exemplo, as *Árvores* apresentadas acima como exemplo teriam a seguinte estrutura de ordenação: *Genérica*: [pai, filho1, f1 do filho1, filho2, f1 de filho2, f2 de filho2, filho3, f1 do filho3], e a *Binária*: [15, 10, 8, 11, 19, 16, 20].

A impressão do livro segue os parâmetros da especificação: imprime-se uma capa, numa página única, com 15 linhas, e então, a partir da segunda página, passe-se a impressão dos *nodos* filhos da raiz. Pode acontecer de o livro não ter nenhum capítulo (portanto, nenhuma seção ou subseção), porém, mesmo assim conter parágrafos. Nesse caso, a capa é impressa normalmente, seguida dos parágrafos. Cada página possui 15 linhas, sendo que cada capítulo é impresso numa página distinta. Por fim, é apresentado um sumário geral do livro.

Juntamente com o código-fonte é disponibilizado um arquivo compilado executável em formato *.jar*, que pode ser inicializado a partir do seguinte comando:

```
$ java -jar App.jar entrada.txt saida.txt
```

Em que “*entrada.txt*” representa o arquivo de entrada do livro a ser lido e “*saida.txt*” representa o livro impresso. A execução do aplicativo gera um trace que tem o seguinte formato:

```
Carregando arquivo entrada.txt... ok
Gerando a árvore... ok
  Capítulos...: 2
  Seções.....: 5
  Subseções...: 2
  Parágrafos...: 16
Gerando o sumário... ok
Imprimindo o livro para o arquivo saida.txt... ok.
```

A seguir são apresentados dois exemplos de entrada e saída.

### **Casos de teste ilustrativos**

#### **1. Entrada: exemplo 1**

```
L Big Java
C Introducao
S Conceitos Basicos
P 3
P 2
S Preparacao do Ambiente
P 2
SS Requisitos Minimios
P 9
SS Instalacao
P 2
P 4
S Fontes de Consulta
P 1
P 3
```

## 2. Entrada: exemplo 2

L Teste Algoritmo  
C Introducao  
S Primeira Seção  
P 1  
P 3  
S Segunda Seção  
P 3  
SS Primeira Subseção  
P 10  
SS Segunda Subseção  
P 1  
P 3  
S Terceira Seção  
P 1  
P 3  
C Segundo Capítulo  
S Seção do Segundo Capítulo  
P 2  
P 3  
SS Subseção do Segundo Capítulo  
P 2  
P 3  
C Terceiro Capítulo  
S Seção do Terceiro Capítulo  
P 3  
P 4  
P 5

## 1. Saída: exemplo 1

1		
2		
3		
4		
5		
6		
7	Big Java	
8		
9		
10		
11		
12		
13		
14		
15		
----- Capa		
1	1. Introducao	
2	1.1 Conceitos Basicos	
3	Lorem Ipsum 1	
4	Lorem Ipsum 2	
5	Lorem Ipsum 3	
6	Lorem Ipsum 1	
7	Lorem Ipsum 2	
8	1.2 Preparacao do Ambiente	
9	Lorem Ipsum 1	
10	Lorem Ipsum 2	
11	1.2.1 Requisitos Minimios	
12	Lorem Ipsum 1	
13	Lorem Ipsum 2	
14	Lorem Ipsum 3	
15	Lorem Ipsum 4	
----- Pag.1		
1	Lorem Ipsum 5	
2	Lorem Ipsum 6	
3	Lorem Ipsum 7	
4	Lorem Ipsum 8	
5	Lorem Ipsum 9	
6	1.2.2 Instalacao	
7	Lorem Ipsum 1	
8	Lorem Ipsum 2	
9	Lorem Ipsum 1	
10	Lorem Ipsum 2	
11	Lorem Ipsum 3	
12	Lorem Ipsum 4	
13	1.3 Fontes de Consulta	
14	Lorem Ipsum 1	
15	Lorem Ipsum 1	
----- Pag.2		
1	Lorem Ipsum 2	
2	Lorem Ipsum 3	
-----		

SUMÁRIO

1.	Introducao	1	
1.1	Conceitos Basicos	1	
1.2	Preparacao do Ambiente		1
	1.2.1 Requisitos Minimios		1
	1.2.2 Instalacao	2	
1.3	Fontes de Consulta	2	

## 2. Saída: exemplo 2

1		
2		
3		
4		
5		
6		
7	Teste Algoritmo	
8		
9		
10		
11		
12		
13		
14		
15		
----- Capa		
1	1. Introducao	
2	1.1 Primeira Seção	
3	Lorem Ipsum 1	
4	Lorem Ipsum 1	
5	Lorem Ipsum 2	
6	Lorem Ipsum 3	
7	1.2 Segunda Seção	
8	Lorem Ipsum 1	
9	Lorem Ipsum 2	
10	Lorem Ipsum 3	
11	1.2.1 Primeira Subseção	
12	Lorem Ipsum 1	
13	Lorem Ipsum 2	
14	Lorem Ipsum 3	
15	Lorem Ipsum 4	
-----		
		Pag.1
1	Lorem Ipsum 5	
2	Lorem Ipsum 6	
3	Lorem Ipsum 7	
4	Lorem Ipsum 8	
5	Lorem Ipsum 9	
6	Lorem Ipsum 10	
7	1.2.2 Segunda Subseção	
8	Lorem Ipsum 1	
9	Lorem Ipsum 1	
10	Lorem Ipsum 2	
11	Lorem Ipsum 3	
12	1.3 Terceira Seção	



13 Lorem Ipsum 1  
 14 Lorem Ipsum 1  
 15 Lorem Ipsum 2

Pag.2

1 Lorem Ipsum 3

Pag.3

1 2. Segundo Capítulo  
 2 2.4 Seção do Segundo Capítulo  
 3 Lorem Ipsum 1  
 4 Lorem Ipsum 2  
 5 Lorem Ipsum 1  
 6 Lorem Ipsum 2  
 7 Lorem Ipsum 3  
 8 2.4.3 Subseção do Segundo Capítulo  
 9 Lorem Ipsum 1  
 10 Lorem Ipsum 2  
 11 Lorem Ipsum 1  
 12 Lorem Ipsum 2  
 13 Lorem Ipsum 3

Pag.4

1 3. Terceiro Capítulo  
 2 3.5 Seção do Terceiro Capítulo  
 3 Lorem Ipsum 1  
 4 Lorem Ipsum 2  
 5 Lorem Ipsum 3  
 6 Lorem Ipsum 1  
 7 Lorem Ipsum 2  
 8 Lorem Ipsum 3  
 9 Lorem Ipsum 4  
 10 Lorem Ipsum 1  
 11 Lorem Ipsum 2  
 12 Lorem Ipsum 3  
 13 Lorem Ipsum 4  
 14 Lorem Ipsum 5

## SUMÁRIO

1.	Introducao	1	
1.1	Primeira Seção	1	
1.2	Segunda Seção	1	
1.2.1	Primeira Subseção	1	
1.2.2	Segunda Subseção	2	
1.3	Terceira Seção	2	
2.	Segundo Capítulo	4	
2.4	Seção do Segundo Capítulo	4	
2.4.3	Subseção do Segundo Capítulo	4	4
3.	Terceiro Capítulo	5	
3.5	Seção do Terceiro Capítulo	5	

## Considerações Finais

Como mostrado por meio dos exemplos de entrada e saída 1 e 2, os requisitos propostos para o trabalhos são cumpridos:

O livro possui título, ocupando uma página inteira (1), bem como apresenta capítulo com título (2,3), considerando o caso em que não há capítulo. A cada novo capítulo uma nova página é iniciada (4), respeitando o número de seções (5) e parágrafos (6). As seções, por sua vez, devem ter título (7), subseções (8) e parágrafos (9). As subseções, por sua vez, devem ter título (10), podendo possuir parágrafos (11), estes, por fim, ocupando o número de linhas especificado (12). A impressão do livro foi feito conforme descrito e esperado nas especificações.

## Referências

GOODRICH, M. T., TAMASSIA, R. *Estrutura de Dados & Algoritmos em Java*. tradução: Bernardo Copstein. - 5 ed. - Porto Alegre: Bookman, 2013.