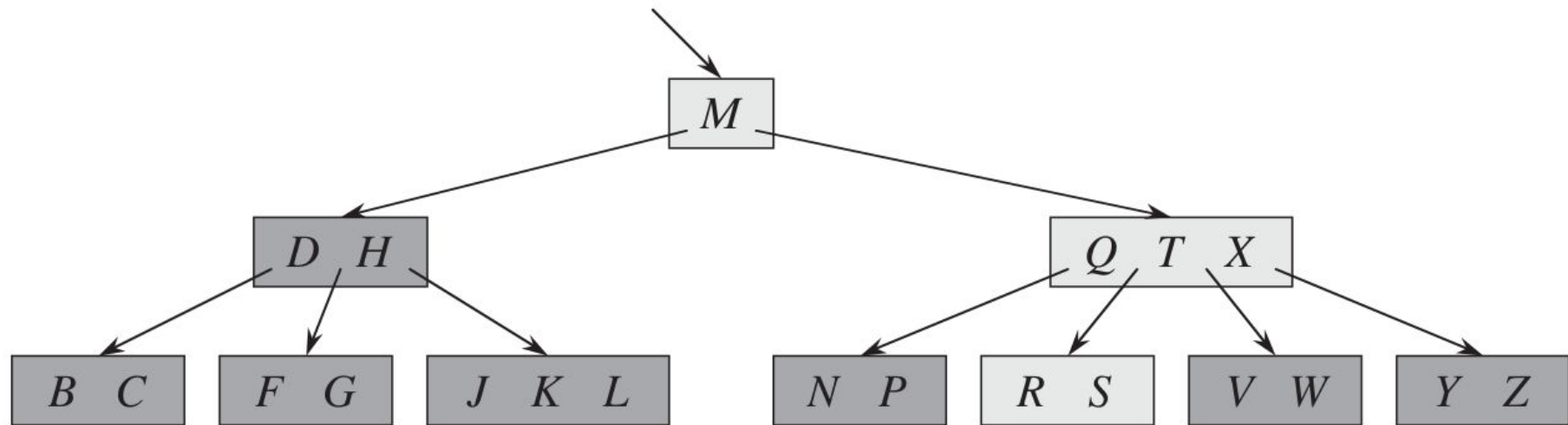
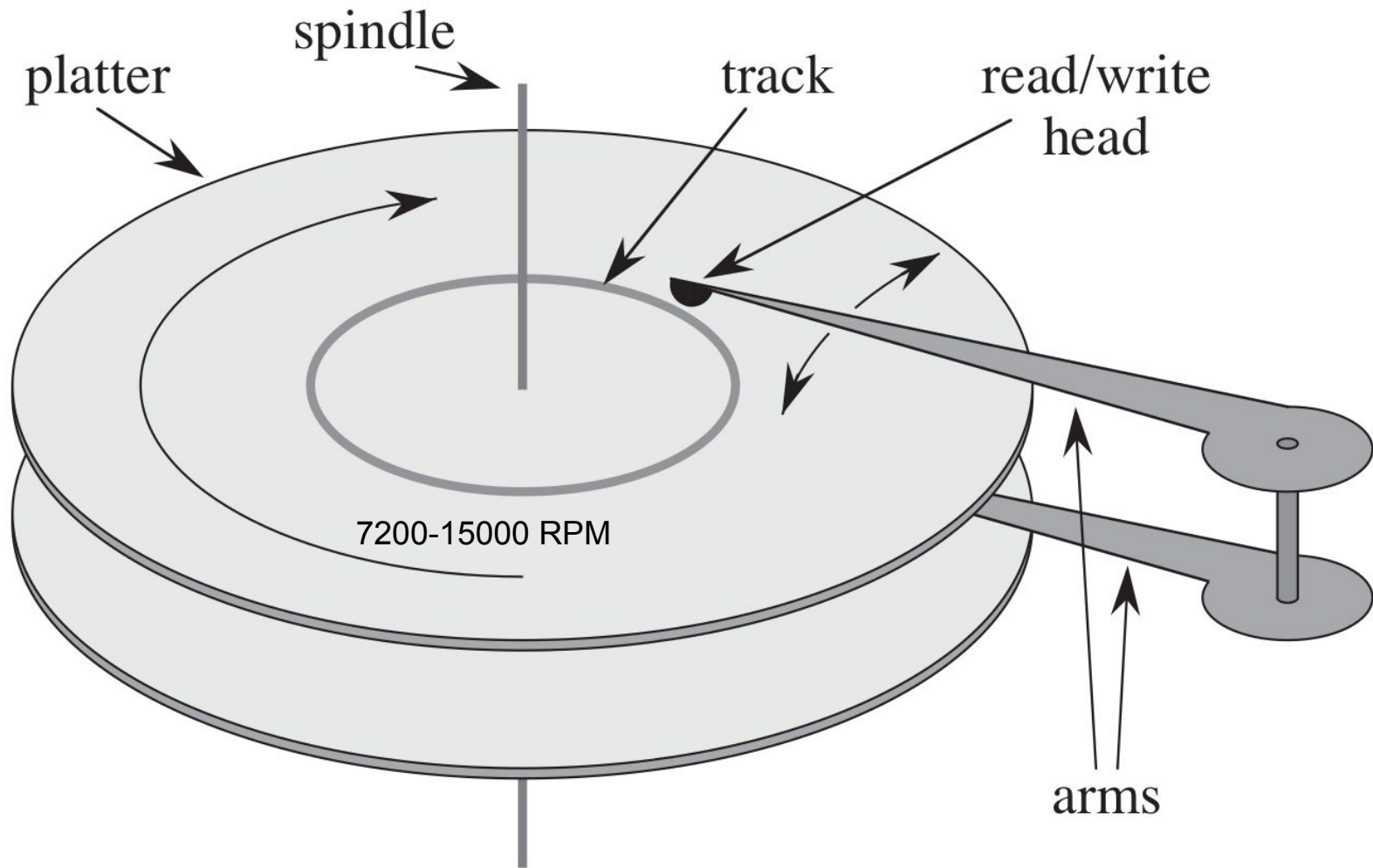
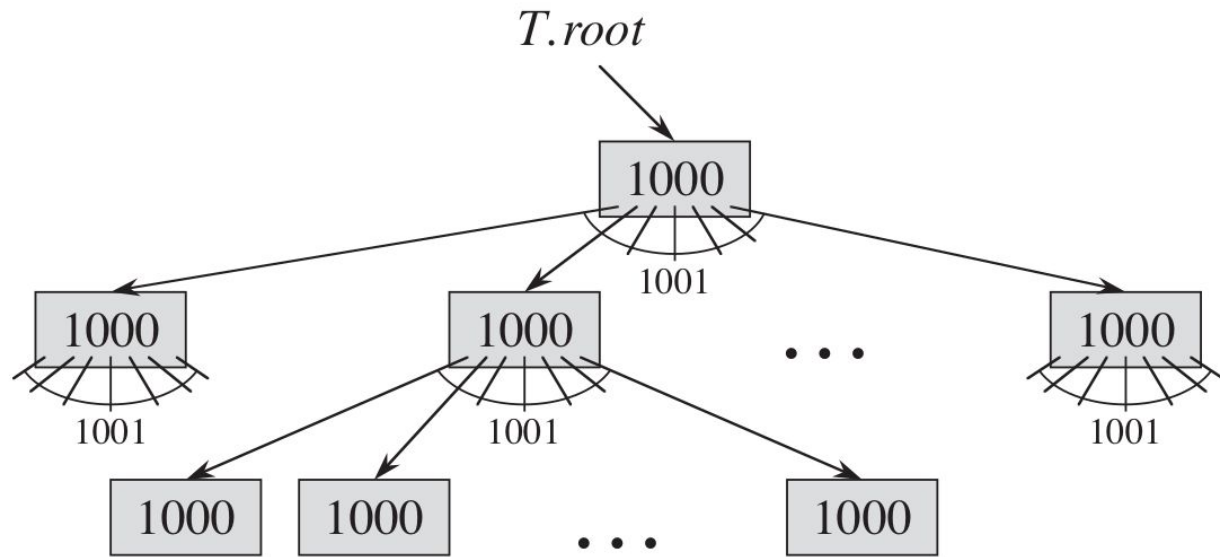


$T.root$







1 node,
1000 keys

1001 nodes,
1,001,000 keys

1,002,001 nodes,
1,002,001,000 keys

B-Tree là một loại cây tìm kiếm được thiết kế để lưu trữ trên ổ đĩa cứng (thiết bị lưu trữ thứ cấp).

1. Mỗi nút x có các thuộc tính cơ bản sau:
 - i. $x.n$ số lượng khóa đang được lưu trong nút x
 - ii. $x.n$ khóa theo trật tự
 $x.key_1 \leq x.key_2 \leq \dots \leq x.key_n$
 - iii. $x.leaf, true$: nếu x là nút lá và $false$ nếu ngược lại
2. Mỗi nút-nội x chứa $x.n + 1$ con trỏ ($x.c_1, x.c_2, \dots, x.c_{x.n}$) tới các nút con của nó. Nút lá không có nút con, hay nói cách khác con trỏ tới nút con của các nút lá có giá trị không xác định.
3. $x.n$ khóa chia giá trị của các khóa được lưu trong các cây con thành $x.n + 1$ miền giá trị
 $k_1 \leq x.key_1 \leq k_2 \leq x.key_2 \leq \dots \leq x.key_{x.n} \leq k_{x.n+1}$
4. Tất cả các nút lá có cùng độ sâu bằng chiều cao h của cây.
5. Các nút đều có giới hạn dưới và giới hạn trên cho số lượng khóa mà nút có thể lưu trữ. Đặt $t \geq 2$ là bậc tối thiểu của một B-Tree.
 - i. Mỗi nút trừ nút gốc có tối thiểu t khóa và tương ứng có tối thiểu t nút con. Nếu cây không rỗng thì nút gốc phải có tối thiểu 1 khóa.
 - ii. Mỗi nút có thể chứa tối đa $2t - 1$ khóa. Vì vậy, mỗi nút-nội x có tối đa $2t$ nút con. Nút được nói là đã đầy nếu có đủ $2t - 1$ khóa.

* Số lượng thao tác truy cập ổ đĩa tỉ lệ thuận với độ cao h của một B-Tree.

** Với một B-Tree bất kỳ với n khóa và bậc tối thiểu $t \geq 2$ thì $h \leq \log_t (n+1)/2$