高级搜索树

B-树：缓存

08-B2

To my mind the most interesting thing in art is the personality of artists;
and if that is singular, I am willing to excuse a thousand faults.

He has given signs of himself which are visible to those
who seek him, and not to those who do not seek him.

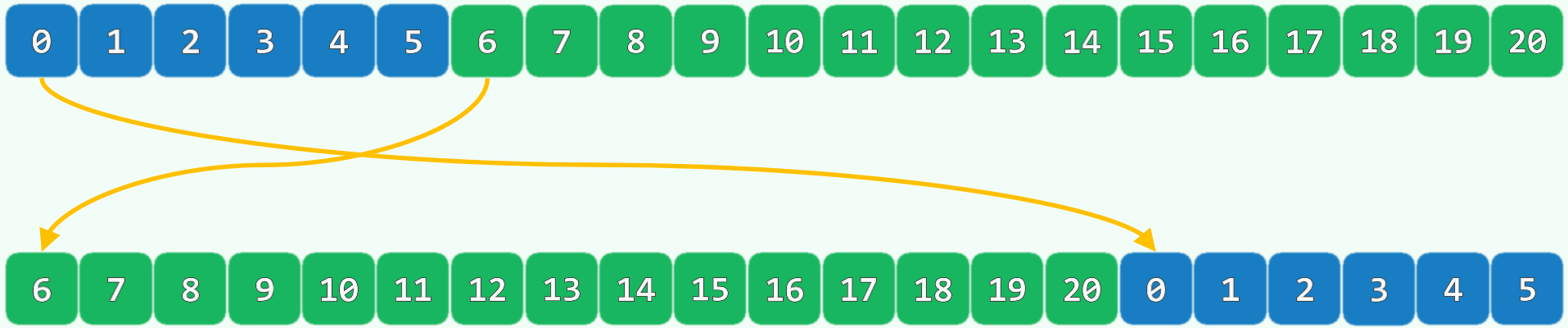
邓俊辉
deng@tsinghua.edu.cn

就地循环位移

❖ 仅用$O(1)$辅助空间，将数组A[0, n)中的元素向左循环移动k个单元

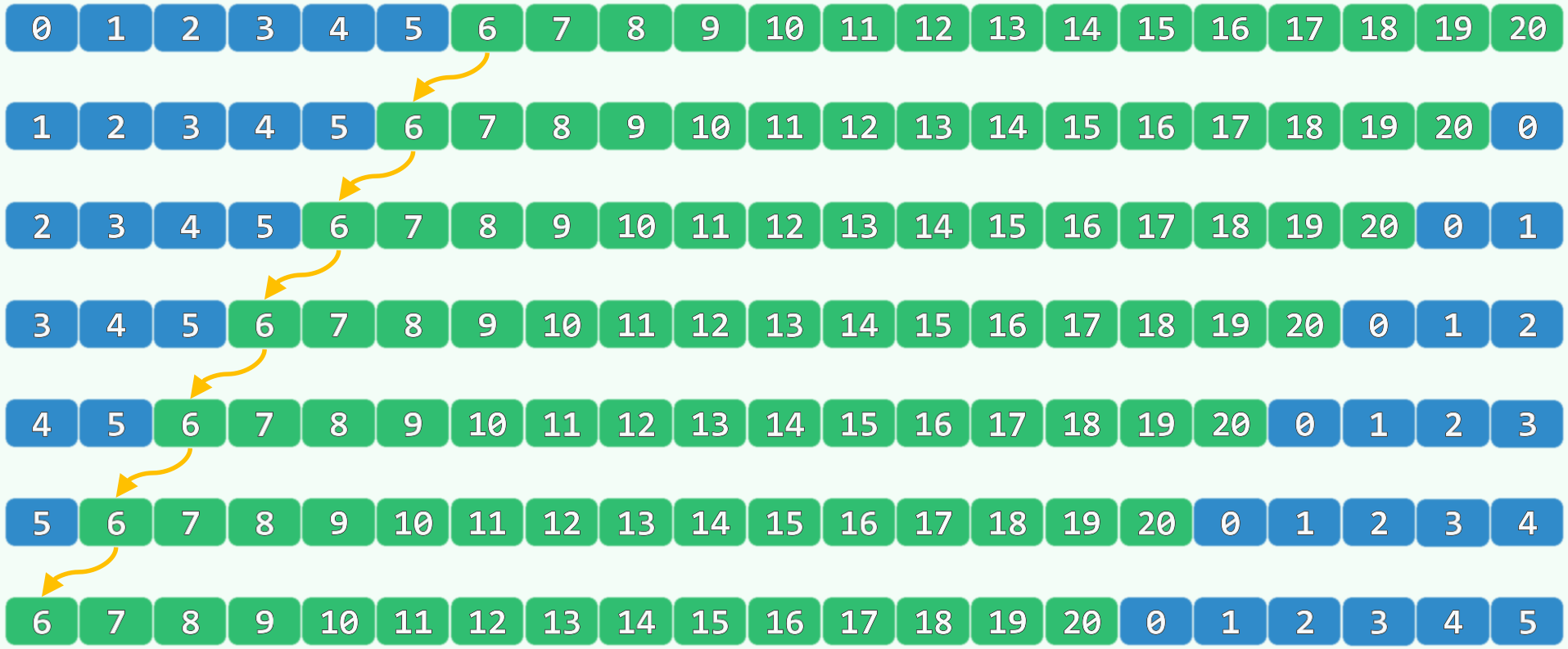
```
void shift( int * A, int n, int k );
```

❖ 比如：shift(A, 21, 6);



蛮力版

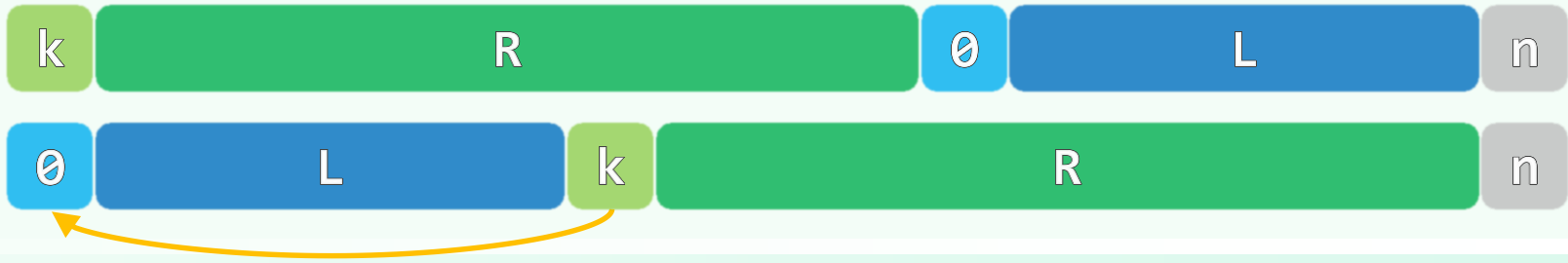
```
void shift0( int * A, int n, int k ) //反复以1为间距循环左移
{ while ( k-- ) shift( A, n, 0, 1 ); } //共迭代k次，O(n*k)
```



迭代版: Stride-k Reference Pattern (1/2)

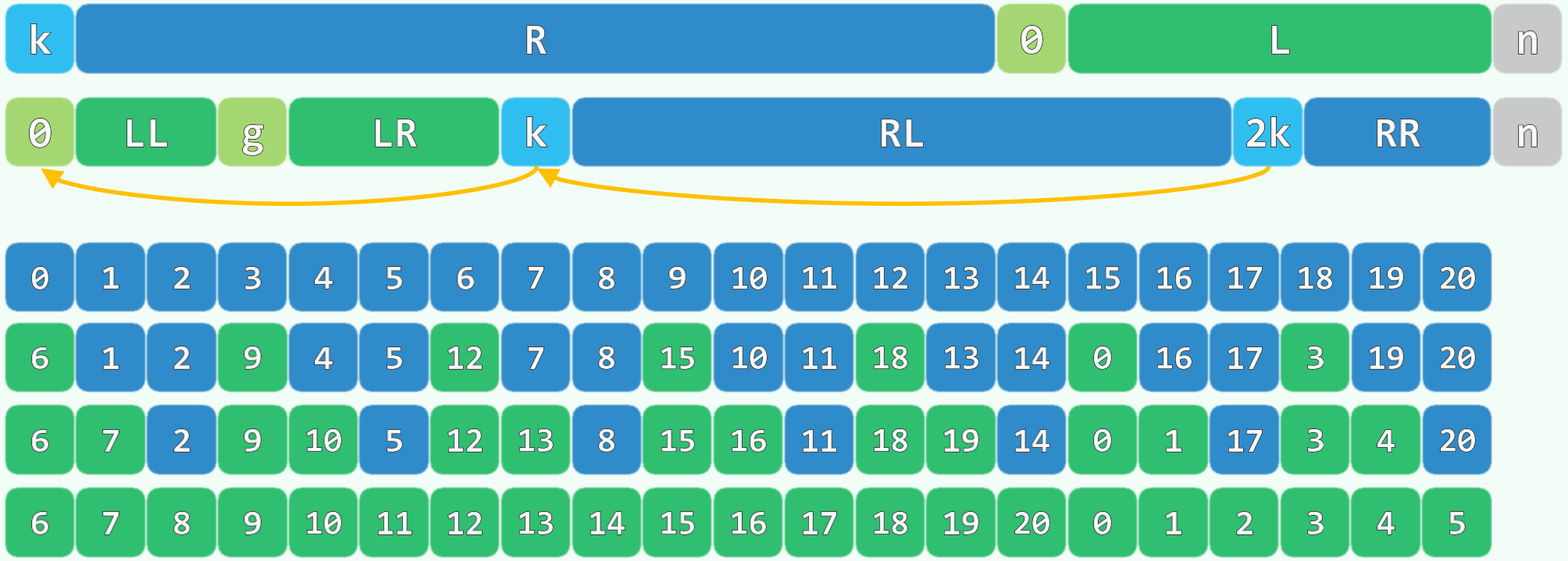
```
int shift( int * A, int n, int s, int k ) { // O( n / GCD(n, k) )
   int b = A[s]; int i = s, j = (s + k) % n; int mov = 0; //mov记录移动次数
   while ( s != j ) //从A[s]出发，以k为间隔，依次左移k位
      { A[i] = A[j]; i = j; j = (j + k) % n; mov++; }
   A[i] = b; return mov + 1; //最后，起始元素转入对应位置
} //[0, n)由关于k的g = GCD(n, k)个同余类组成，shift(s, k)能够且只能够使其中之一就位
```

其它的同余类呢...



迭代版: Stride-k Reference Pattern (2/2)

```
void shift1(int* A, int n, int k) { //经多轮迭代，实现数组循环左移k位，累计O(n+g)
   for (int s = 0, mov = 0; mov < n; s++) //O(g) = O(GCD(n, k))
      mov += shift(A, n, s, k);
}
```



倒置版: Stride-1 Reference Pattern

```
void shift2( int * A, int n, int k ) {

    reverse( A, k ); //O(3k/2)

    reverse( A + k, n - k ); //O(3(n-k)/2)

    reverse( A, n ); //O(3n/2)

} //O(3n)
```