14-C1

排序

希尔排序：框架+实例

瓜熟蒂落，水到渠成

今有物不知其数，三三数之剩二；五五数之剩三；七七数之剩二。 问物几何

邓俊辉

deng@tsinghua.edu.cn

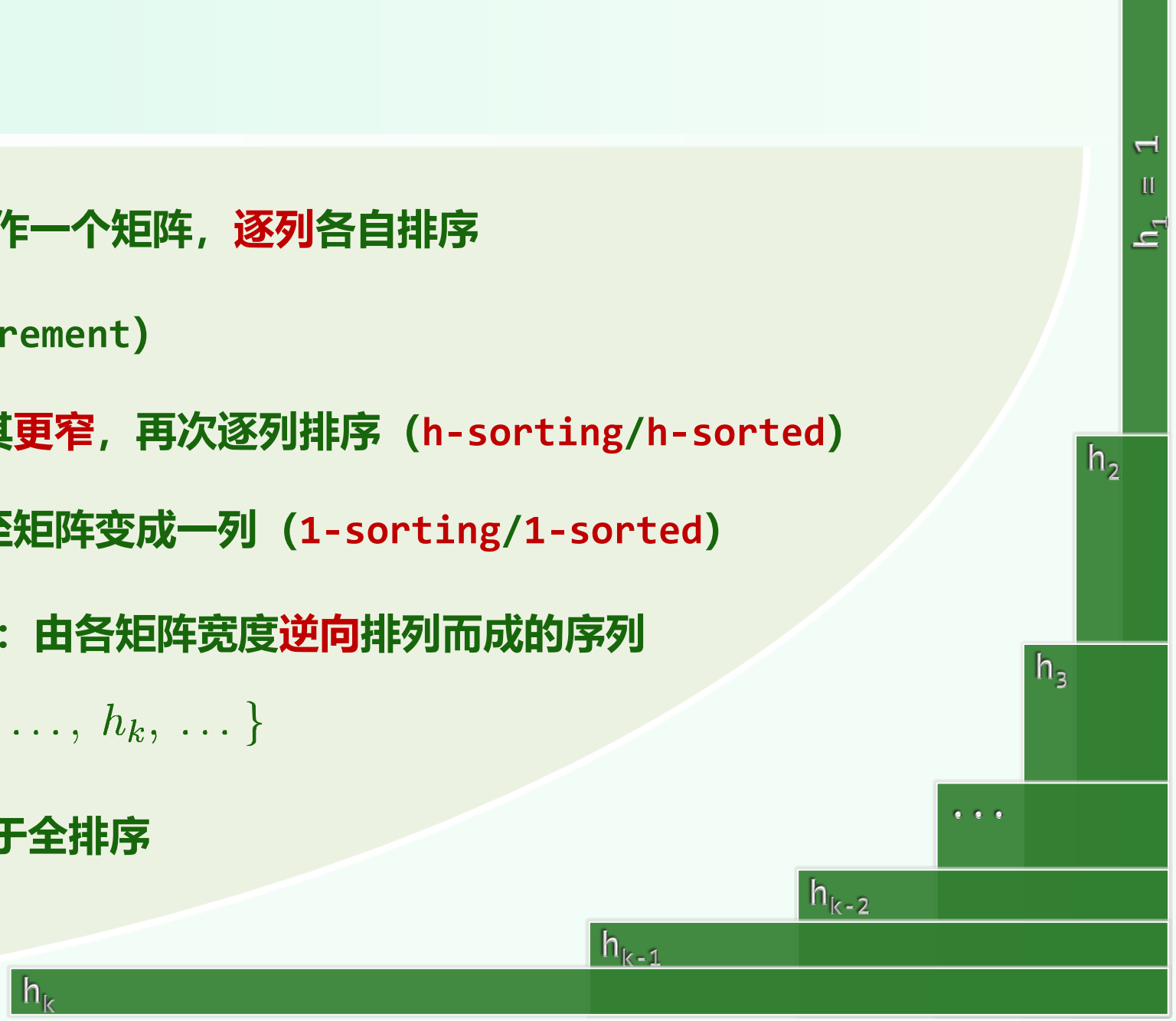
Shellsort

- D. L. Shell：将整个序列视作一个矩阵，逐列各自排序
- 递减增量（diminishing increment）
  - 由粗到细：重排矩阵，使其更窄，再次逐列排序（h-sorting/h-sorted）
  - 逐步求精：如此往复，直至矩阵变成一列（1-sorting/1-sorted）
- 步长序列（step sequence）：由各矩阵宽度逆向排列而成的序列

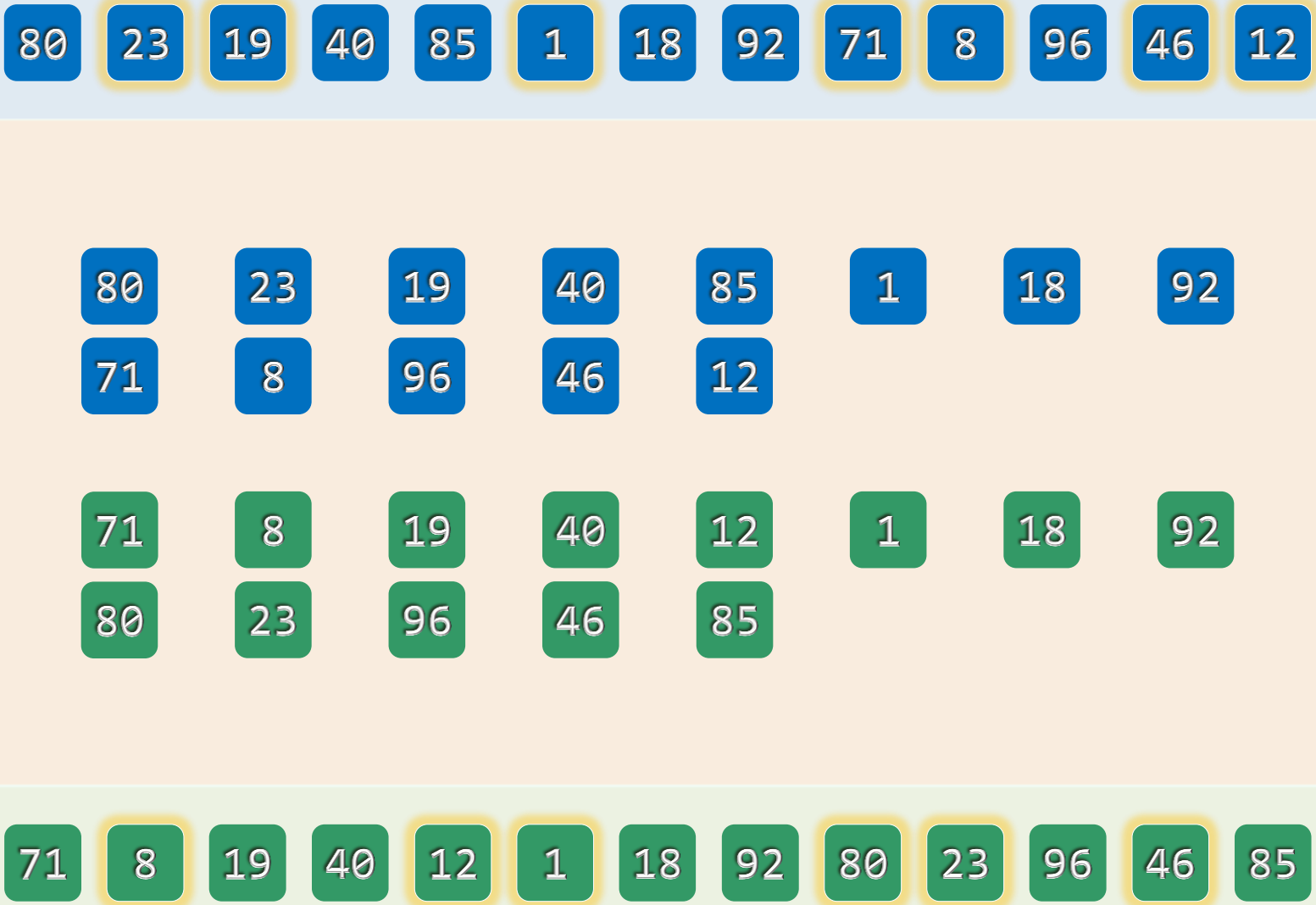
$$\mathcal{H} = \{ h_1 = 1, h_2, h_3, \ldots, h_k, \ldots \}$$

- 正确性：最后一次迭代，等同于全排序

1-sorted = ordered

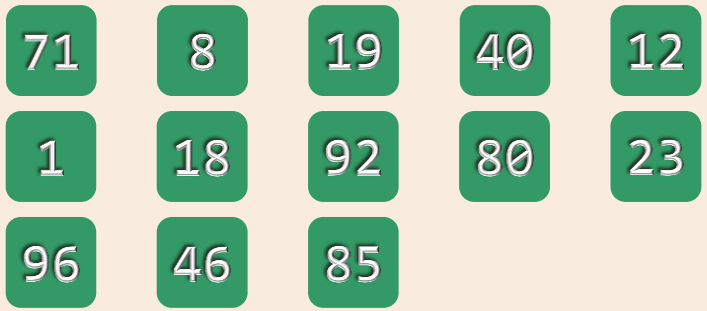


实例：$h_5 = 8$


80 23 19 40 85 1 18 92 71 8 96 46 12
80 23 19 40 85 1 18 92
71 8 96 46 12
71 8 19 40 12 1 18 92
80 23 96 46 85
71 8 19 40 12 1 18 92 80 23 96 46 85

实例：$h_4 = 5$

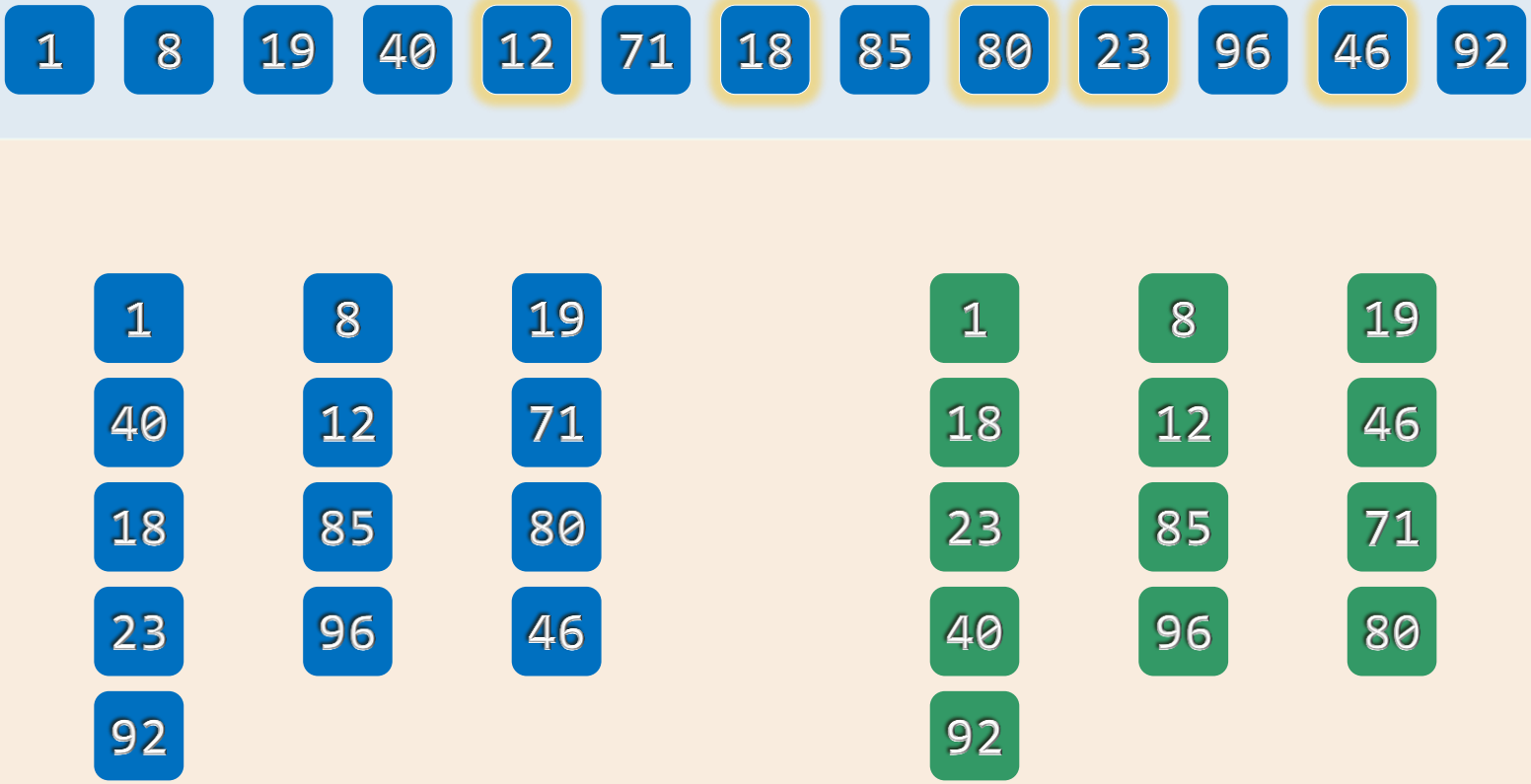

1 8 19 40 12 71 18 85 80 23 96 46 92


71 8 19 40 12
1 18 92 80 23
96 46 85

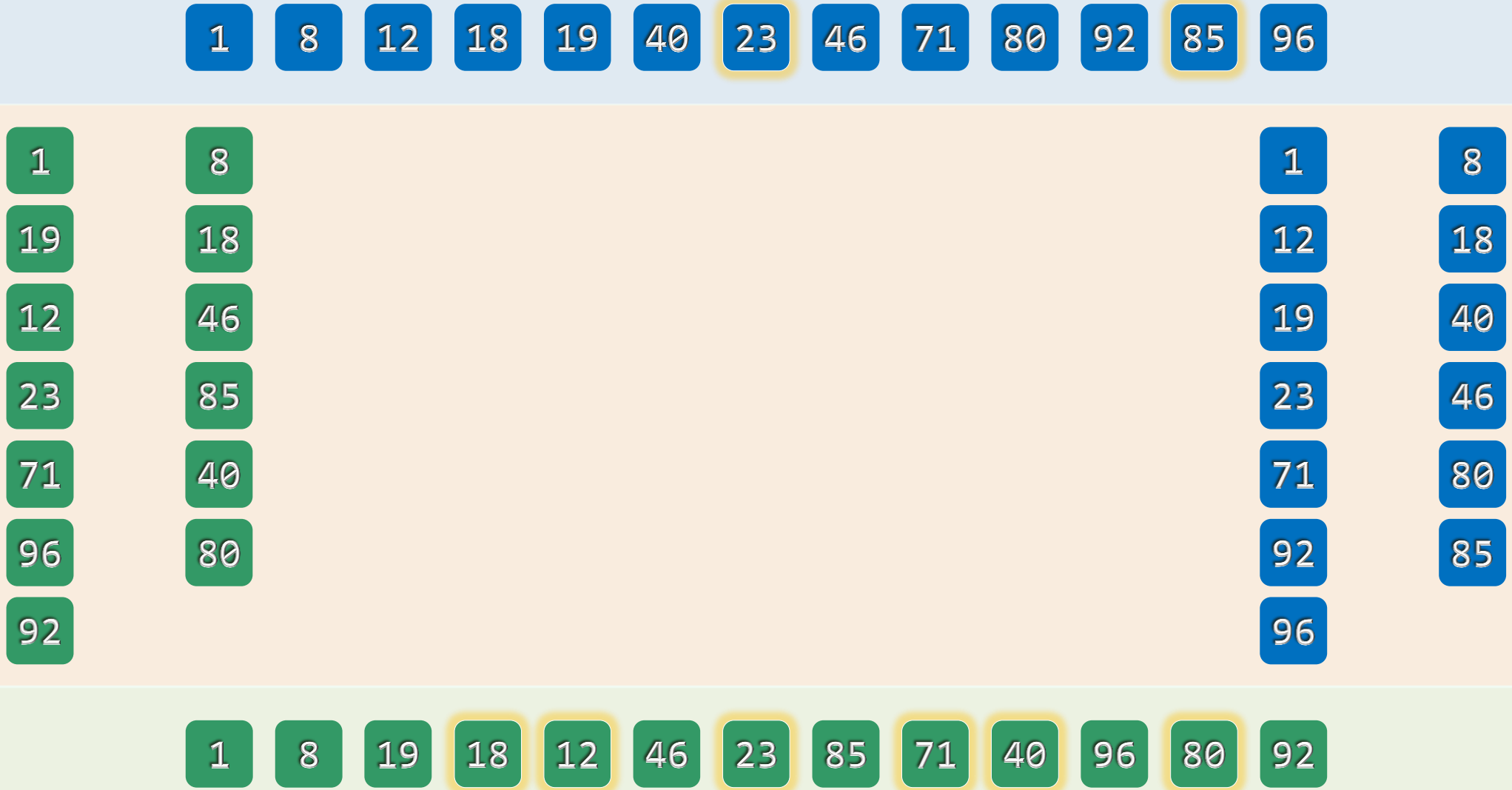

1 8 19 40 12
71 18 85 80 23
96 46 92


71 8 19 40 12 1 18 92 80 23 96 46 85

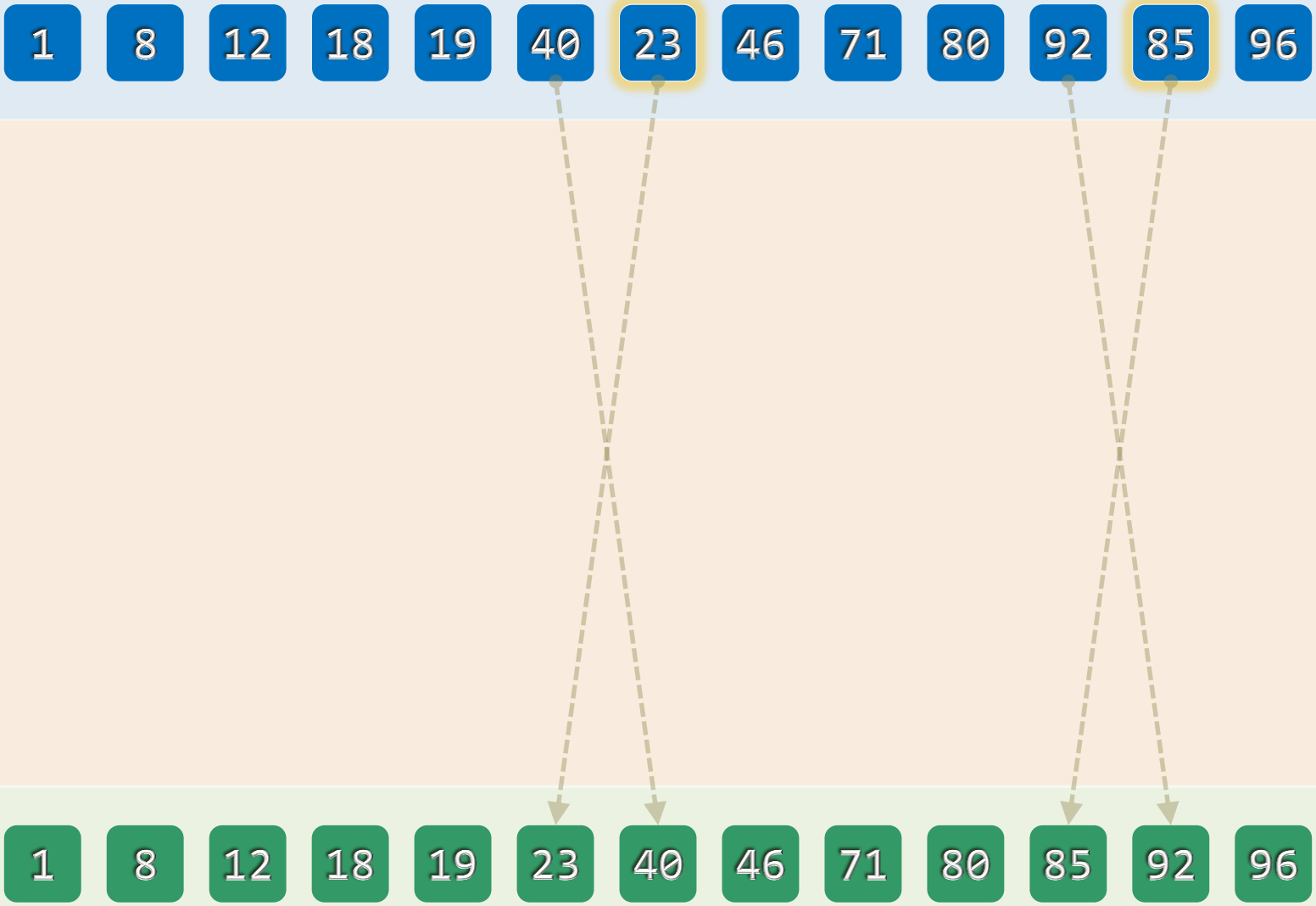
实例：$h_3 = 3$


1 8 19 40 12 71 18 85 80 23 96 46 92
1 40 18 23 92
8 12 85 96
19 71 80 46
1 18 23 40 92
8 12 85 96
19 46 71 80
1 8 19 18 12 46 23 85 71 40 96 80 92

实例：$h_2 = 2$


1 8 12 18 19 40 23 46 71 80 92 85 96
1 19 12 23 71 96 92
8 18 46 85 40 80
1 12 19 23 71 92 96
8 18 40 46 80 85
1 8 19 18 12 46 23 85 71 40 96 80 92

实例：$h_1 = 1$

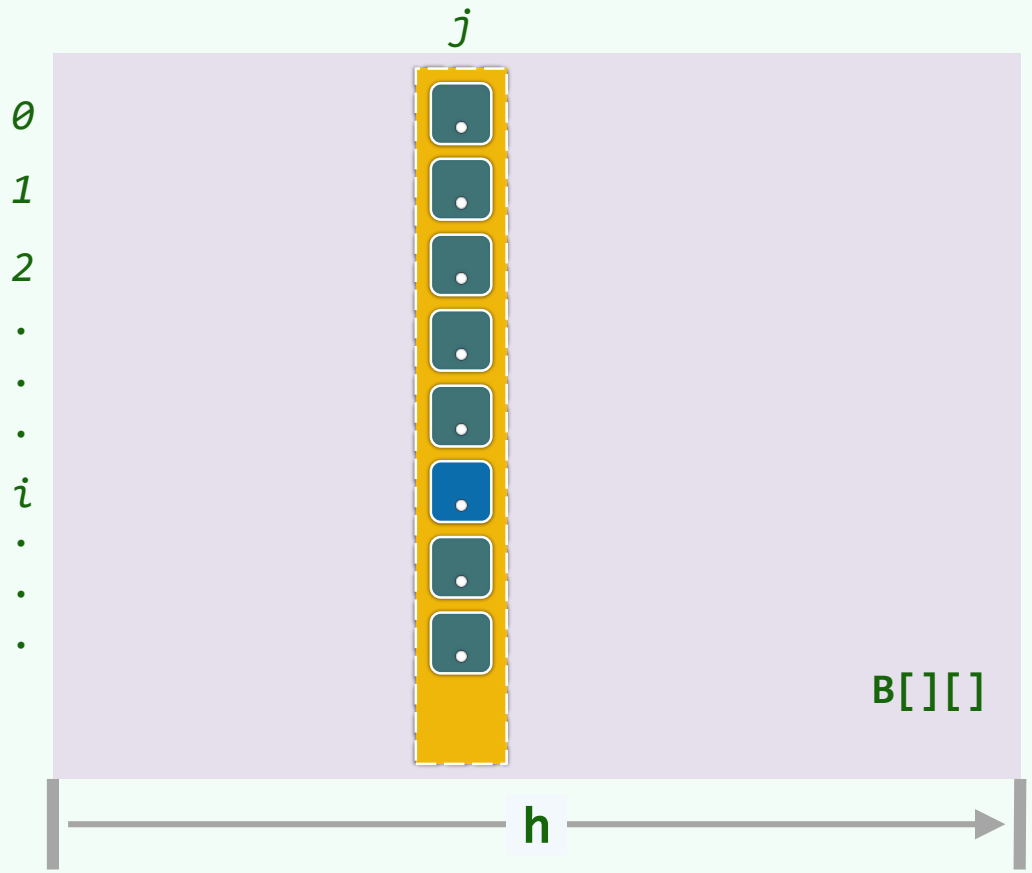


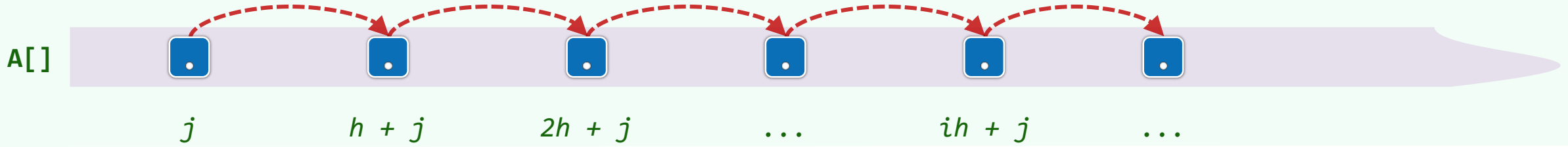
Call-by-rank

- 如何实现矩阵重排？莫非，需要使用二维向量？
- 实际上，借助一维向量足矣
- 在每步迭代中，若当前的矩阵宽度为h，则

$$B[i][j] = A[i \cdot h + j]$$

或 $A[k] = B[k/h][k\%h]$





实现

```
template <typename T> void Vector<T>::shellSort( Rank lo, Rank hi ) {
   for ( Rank d = 0x7FFFFFFF; 0 < d; d >>= 1 ) //PS Sequence: 1, 3, 7, 15, 31, ...
      for ( Rank j = lo + d; j < hi; j++ ) { //for each j in [lo+d, hi)
         T x = _elem[j]; Rank i = j; //within the prefix of the subsequence of [j]
         while ( (lo + d <= i) && (x < _elem[i-d]) ) //find the appropriate
            _elem[i] = _elem[i-d]; i -= d; //predecessor [i]
         _elem[i] = x; //where to insert [j]
      }
} //0 <= lo < hi <= size <= 2^31
```