

## 1 LSD Radix Sort

l = [751, 439, 860, 9, 123, 724, 423, 109, 144]

Columns

	0		1		2		3		4		5		6		7		8		9	
--	---	--	---	--	---	--	---	--	---	--	---	--	---	--	---	--	---	--	---	--

State 1:

	860		751				123		724										439	
							432		144										9	
																			109	

State 2:

	9				123		439		144		751		876							
	109				423															
					724															

State 3:

	9		144						439						741		860			
			123						432						724					
			109																	

## 2 Operations of a min-heap

None 6

None 2 6

None 2 6 7

None 2 4 7 6

None 4 6 7

None 4 5 7 6

None 4 5 7 6 9

None 5 6 7 9

None 3 5 7 9 6

None 1 5 3 9 6 7

None 3 5 7 9 6

### 3 Implementing MSD Radix Sort

Of

### 4 Comparing MSD Radix Sort With Other Sorting Algorithms

**MSD vs Comparative Sorting** In truth, the idea of a non-comparative sorting algorithm is icky icky. Conceptually, a comparative sort just makes more sense because in almost any way we can compare "things" and I don't think the complexity and the brain power required to come up with a non-comparative is worth the effort. I understand that there are somethings that aren't comparable like, "how do we organise these different coloured balls?". But then I would ask, are we really sorting anything at that point or are we just trying to organise them heuristically? If the former is so, we can just add some sort of comparable "attribute" to the concept. For example, just attached a number value to the colours, like how we might with the suits of a card. Clearly they aren't actual "things" with inherent comparison, but we associated them with a static hierarchy. So often we would assign diamonds a 0, clubs 1, hearts 2 and spades 3, giving us a quantitative and comparable value to associate with our suit concept. If it's the latter, then well, buckets and dictionaries. And they aren't even ordered anyways.

**MSD vs LSD** From a conceptual standpoint, I believe LSD to be easier to understand, and it particularly more efficient when we are dealing with items of fixed length sizes, strings or integers. MSD I feel requires a little bit more brain power, but lends itself to being more efficient for variable length items. However, this is a classic example of the trade-offs we often see when implementing algorithms, that trade-off being the readability and "simplicity" of our code and efficiency.