

# Chapter 1

## Enumeration Sort

### 1.1 Objectives:

At the end of this lecture the learner will be able to:

- Understand the meaning of enumeration sort
- Apply enumeration sort algorithm to sort a list of numbers.

### 1.2 Definition of Enumeration Sort

First let us start our discussion with the definition of enumeration sort.

**Definition 1.** *Enumeration Sort: According to Knuth (1973), it is a method of finding the exact position of each element in a sorted list by comparing and finding the frequency of elements having smaller value. That is if  $p$  elements are smaller than  $a_q$ , then  $a_q$  occupies the  $(p+1)$ th position in the sorted list.*

### 1.3 Enumeration Sort Algorithm

Muller and Preparata (1975) proposed a non standard PRAM model to carry out enumeration sorting in logarithmic time. The algorithm consumes  $\theta(\log n)$  to spawn  $n^2$  processors and a constant time to sort.

#### 1.3.1 Pseudo Code

Contract: EnumSort: List  $\rightarrow$  List

Purpose: This algorithm is to sort a list of elements in increasing order.

Example: EnumSort([2,1,5,4,3])  $\rightarrow$  [1,2,3,4,5]

Procedure EnumSort(numList[0..n-1])

begin

spawn  $n^2$  processors denoted by  $P_{i,j}$  where  $i,j$  ranges from 0 to  $n-1$

for all processors  $P_{i,j}$  where  $i,j$  ranges from 0 to  $n-1$  do

begin

Initialize Position[i] to 0

if numList[i] < numList[j] or numList[i] = numList[j] and  $i < j$  then

Position[i] is set to 1

endif

```

end
for all processors  $P_{i,0}$  where  $i$  ranges from 0 to  $n-1$  do
begin
SortedList[( $n-1$ )-Position[ $i$ ]] is set to numList[ $i$ ]
end
end

```

**Dry Run:** Let us consider numList=[5,2,3]. Here  $n=3$ .

Table 1.1: Dry Run of the Enumeration Sort Algorithm-Finding the final Position of an element

P00	P01	P02	P10	P11	P12	P20	P21	P22
Pos[0]=0	Pos[0]=0	Pos[0]=0	Pos[1]=1	Pos[1]=1	Pos[1]=2	Pos[2]=1	Pos[2]=1	Pos[2]=1

Table 1.2: Dry Run of the Enumeration Sort Algorithm-Determining the Sorted List

P00	P10	P20
Sorted[(3-1)-Pos[0]]	Sorted[(3-1)-Pos[1]]	Sorted[(3-1)-Pos[2]]
=Sorted[2]	=Sorted[0]	=Sorted[1]
=a[0]	=a[1]	=a[2]
=5	=2	=3

Table 1.3: Dry Run of the Enumeration Sort Algorithm-Final Sorted List

0	1	2
2	3	5