**Assignment**

(2018btecs00212)

**Q.3. Given an array A of n elements, each of which is an integer in the range [1, n2]. How do we sort the array in O(n) time?**

The complexity of the merge and quick sort is O(nlogn), so we don’t use that we can use the radix sort the complexity of radix sort is given by O(d\*(n+b)) where d is the digits present in the number and b is base of number system.

So as according to our question d = logbn

So overall time complexity is O((n+b)\*O(logb(n)). Which is more than the time complexity of comparison based sorting algorithms for a large k. The idea is to change base b. If we set b as n, the value of O(logb(n)) becomes O(1) and overall time complexity becomes O(n).