**Counting and Optimisation Lab 1 Application of a Sliding window**

**Document Id 13**

**Questions**

1. Given an array of numbers {2, 3, 4, 2, 6, 2, 5, 1} and a sliding window with size w= 3

The output of max values for all possible windows is

1. {3, 4, 6, 6, 6, 5}
2. {4, 4, 6, 6, 5, 5}
3. {4, 4, 6, 6, 6, 5}
4. None of the above

2) For the values in Q1 the obvious(brute force) solution to find the minimum values for all

possible windows is as follows, Take the first w elements of the list

→ take the minimum of the first w elements

→ slide one position down the list

→ starting at the new position take the minimum of the first w elements

→ repeat

The time complexity for this approach would be

1. O(n)
2. O(wn)
3. O(n log n)
4. O(n^2)

3) The brute force solution in 2 is not satisfactory, is it possible to improve on this if so how

Data Science

https://en.wikipedia.org/wiki/Time-series\_segmentation

**Practical Given an arbitary time series array as shown below**

**int**[] timeSeries = **new int**[]{ 1, 3, 1, 4, 2, 3, 5, 4};

Find the index of a given maximum time as a sliding window problem

For the array above the correct answer is 6 as timeSeries[6] = 5