Laser Domino Sorting

Introduction :

Sorting algorithms are one of the most widely used algorithms applicable to reducing time in solving a wide range of problems. With increase in data size, the usefulness of the algorithm becomes noticeable – so do the concerns of its time consumption.

O(n) sorting algorithms that are stable have been found, but the number of comparisons they make is either a multiple of n or involve comparisons with numbers that are just references not actually present in the data set. The main obstacle that hinders us from sorting by making just n comparisons is that *data elements are passive, not active.* The Laser Domino sorting algorithm aims at accomplishing this by simply using active, responsive data elements – *nanolasers powered by nano-photodiodes.*

Apparatus :

The data unit of our sorting algorithm is a nanolaser powered by a nano-photodiode. Suppose the range of numbers from which we expect our input array A[1..N] of N numbers to be from is 1 to MAX. We'll use MAX nanolasers *aligned* , which are labeled from 1 to MAX, thus representing the entire range.
Additionally, we use two nanolasers -

- *Trigger* - At 0^{th} position
- *Terminate* - At $(MAX + 1)^{th}$ position

Each nanolaser will be in either of the two states -

- Normal - the nanolaser's labeled value isn't present in A
- Excited – the nanolaser's labeled value is present in A

whereas Trigger and Terminate will always be in excited state.
Each nanolaser will be connected to a computer system by an electrically conductive line. The computer system will

accept the input array A and excite nanolasers corresponding to the numbers in A via signals through these lines. The same line will also carry a signal to the computer system from a *triggered* nanolaser; a nanolaser is triggered when its nano-photodiode generates current when another nanolaser's beam is incident on it. Upon triggering, the nanolaser's beam turns on.

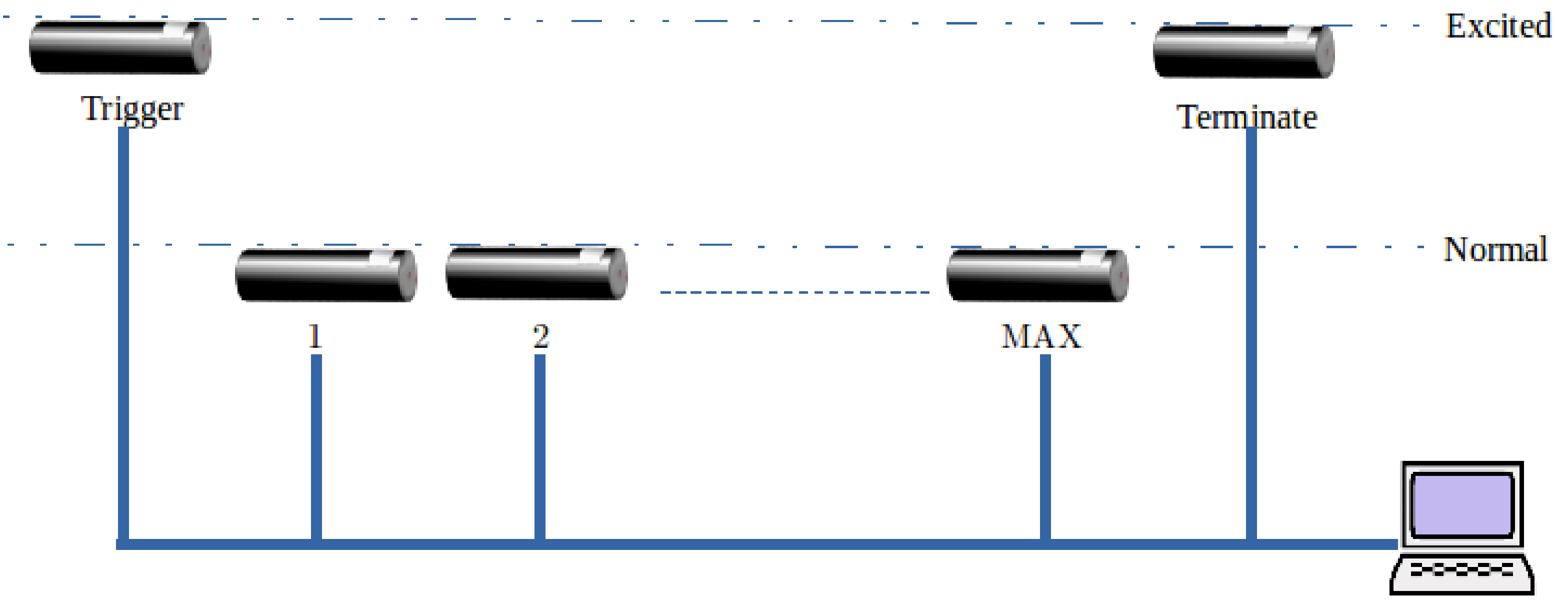

Fig. 1 : Initial state

<u>Working</u> :

(1) The computer system accepts input numbers A[1..N] and simultaneously excites the corresponding nanolasers. It should also keep a count of the repetition of the input numbers by hashing in the array C[1..MAX].

(2) After accepting all input numbers, the Trigger is turned on.

(3) The Trigger's beam is incident on the nano-photodiode of the 1^{st} nanolaser (say labeled as x) in its path, thus triggering it. The nanolaser sends a signal to the computer system which looks upon C[x] and inserts x C[x] times in the empty array B. Simultaneously, the nanolaser's beam is turned on.

(4) In a similar way, the following excited nanolasers get triggered, each causing its label (say l) to be appended C[l] times to B.

(5) The final excited nanolaser lastly triggers Terminate, whose signal could be used to turn off the beam of Trigger (thus turning off all other nanolasers) and bring all excited nanolasers (except Trigger and Terminate) back to the normal state.

Thus, we finally have a sorted output array B[1..N] for given input array A[1..N].

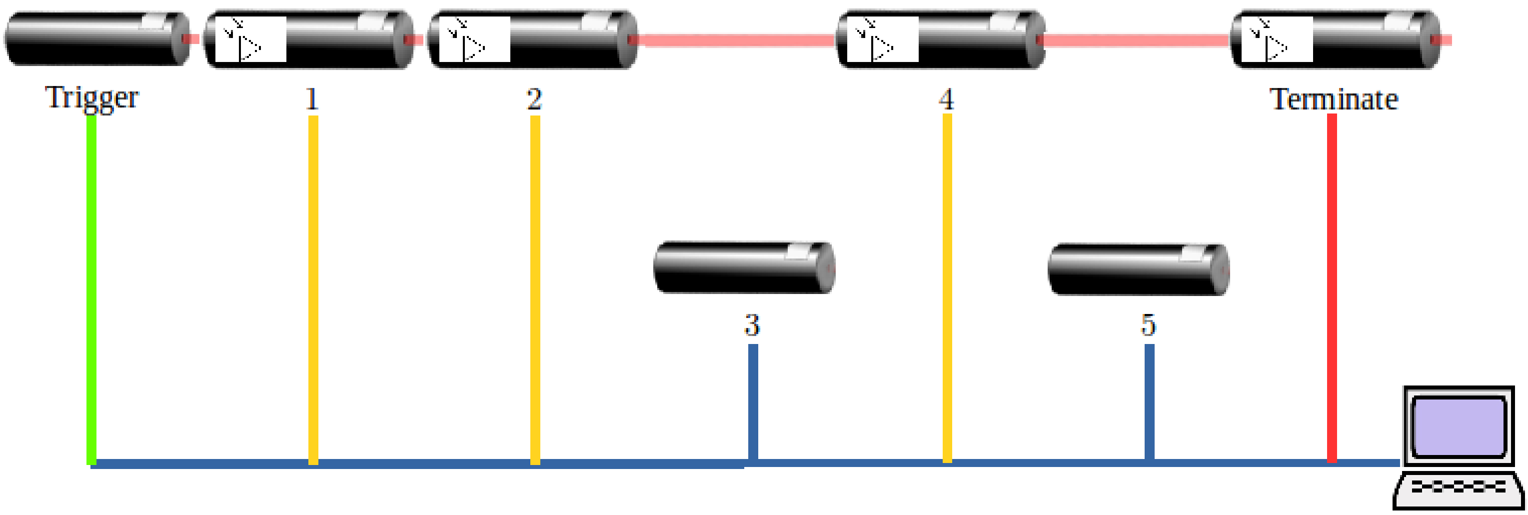

Fig. 2 : Working of LDS for A[1..4] = [4,1,2,4], MAX = 5

For above example, we get C[1..5] = [1,1,x,2,x], where x means "don't care". Nanolasers 1, 2 and 4 are excited. The green, yellow and red lines depict steps (2), (3)-(4) and (5) respectively.

<u>Analysis</u> :

(1) The algorithm involves no comparison operations. The only primitive operations involved are construction of arrays C and B. The running time of the algorithm is $\Theta(n)$ for any case.

(2) We can overwrite A instead of creating new array B for the sorted output array, but we need C. So our algorithm is out-of-place.

(3) By some modification of C, we can extend its usage to ensure stable sorting of the input array.

Conclusion :

The Laser Domino Sorting algorithm is named so because it works on the principle of domino effect of nanolasers.