#9)

My input sequences were generated by creating a string for each algorithm that contains a set of positive and negative integers. The set contains 90 elements. The output shows what I believe is the correct representation of each of the algorithms. The times are in order of quickest to slowest, (linear, quadratic, and cubic). The runtimes can also be found below in the copied output text.

public static void main(String[] args) {

// TODO Auto-generated method stub

System.out.println("String = 500,-400,300,500,400,-300, 500, 400, 300, 500,400,-300,500,400,-300,-500,400,300,500,-400,300,500,-400, 300, 500,400,-300,500,400,300,500,400,-300,500,400,-300,-500,400,300,500,-400,300,500,400,300,500,-400,300,500,400,300,500,400,-300, 500, 400, 300,-500,400,300,500,-400,300,-10000,-50000,-20000,-4000,-50000,-7000, -9000,-600,-50,-30,-50,-70,-20,-70,-5,-5");

System.out.println();

System.out.println("Linear maximum contiguous subsequence sum algorithm :");

TestAlgorithms.maximumSubsequenceSum(new int[]{500,-400,300,500,400, 300,500,400,300,500,400,-300,500,400,-300,-500,400,300,500,-400,300, 500,-400,300,500,400,-300,500,400,300,500,400,-300,500,400,-300,-500,400,300,500,-400,300,500,400,300,500,-400, 300,500,400,300,500, 400,-300,500,400,300,-500,400,300,500,-400,300,-10000,-50000,-20000,-4000,-50000,-7000,-9000,-600,-50,-30,-50,-70,-20,-70,-5,-5});

System.out.println();

System.out.println("Quadratic maximum contiguous subsequence sum algorithm :");

TestAlgorithms.maxSubsequenceSum2(new int[]{500,-400,300,500,400,-300, 500,400,300,500,400,-300,500,400,-300,-500,400,300,500,-400,300,500,-400,300,500,400,-300,500,400,300,500,400,-300,500,400,-300,-500,400, 300,500,-400,300,500,400,300,500,-400,300,500,400,300,500,400,-300, 500,400,300,-500,400,300,500,-400,300,-10000,-50000,-20000,-4000,-50000,-7000,-9000,-600,-50,-30,-50,-70,-20,-70,-5,-5});

System.out.println();

System.out.println("Cubic maximum contiguous subsequence sum algorithm :");

TestAlgorithms.maxSubsequenceSum3(new int[]{500,-400,300,500,400,-300,500,400,300,500,400,-300,500,400,-300,-500,400,300,500,-400,300,

500,-400,300,500,400,-300,500,400,300,500,400,-300,500,400,-300,-500, 400,300,500,-400,300,500,400,300,500,-400,300,500,400,300,500,400,-300,500,400,300,-500,400,300,500,-400,300,-10000,-50000,-20000,-4000,-50000,-7000,-9000,-600,-50,-30,-50,-70,-20,-70,-5,-5});

Output:

String = 500,-400,300,500,400,-300,500,400,300,500,400,-300,500,400,-300,-500,400,300,500,-400,300,500,-400,300,500,400,-300,500,400,300, 500,400,-300,500,400,-300,-500,400,300,500,-400,300,500,400,300,500,-400,300,500,400,300,500,400,-300,500,400,300,-500,400,300,500,-400, 300,-10000,-50000,-20000,-4000,-50000,-7000,-9000,-600,-50,-30,-50,-70,-20,-70,-5,-5

Linear maximum contiguous subsequence sum algorithm :

Time in milliseconds :3732

Maxsum :13300

Quadratic maximum contiguous subsequence sum algorithm :

Time in milliseconds :71846

Maxsum :13300

Cubic maximum contiguous subsequence sum algorithm :

Time in milliseconds :936793

Maxsum :13300