Programming Project: Selection Problem

Tyler Do

1. **Problem definition:**

   The selection problem seeks to find the ith order statistic in a set A of n distinct numbers, where 1 <= I <= n. The ith order statistic is the ith smallest element in the set. Originally the measurements for the RT consisted of the following input sizes n = 10000, 20000, 30000, 40000, …., 200000. More specifically, n takes 20 values from 10k to 200k with increments of 10k. I scaled down to n = 1000, 2000, 3000, 4000, …., 20000. This helped execute the experiment in a faster time frame. In this project, let us take I = 2n/3 for all the experiments. First algorithm will use Insertion sort, the second algorithm will use Heapsort, and the third algorithm will use Randomized-Select.

2. **Algorithms and RT analysis:**

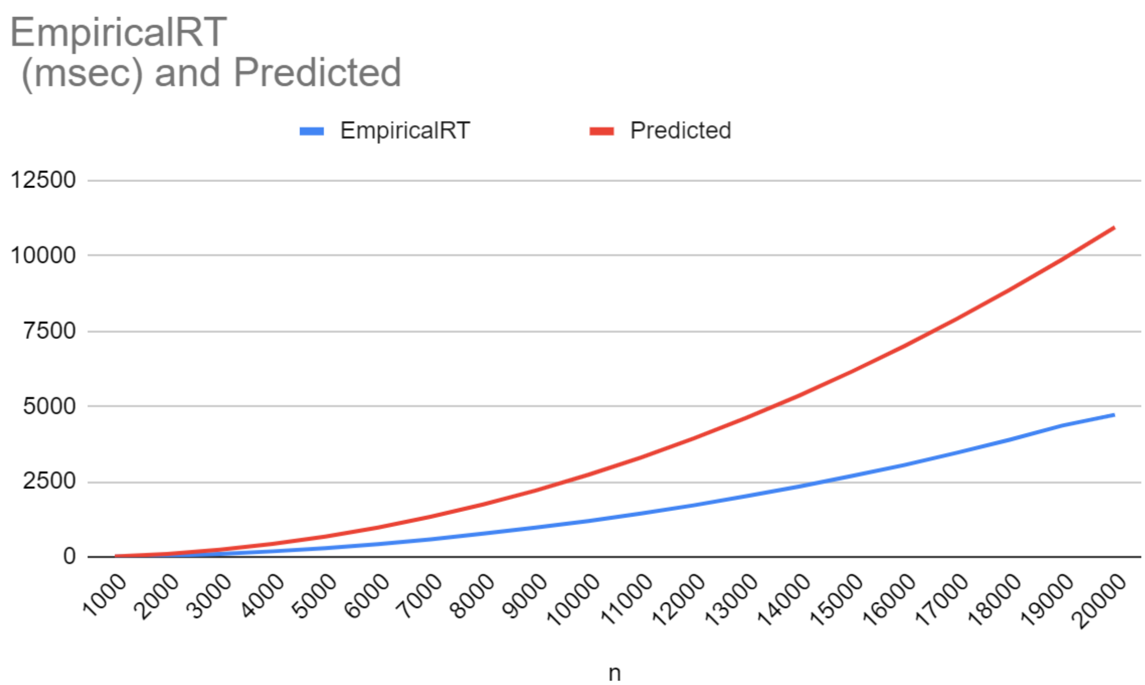
<u>ALG1(v,n,i)</u>	**<u>ALG2(v,n,i)</u>**	**<u>ALG3(v,n,i)</u>**
INSERTION-SORT(v,n)	HEAPSORT(v,n)	x = RANDOMIZED-SELECT(v,l,n,i)
Print v[i]	Print v[i]	Print x
RT = $O(n^2)$	RT = O(nlgn)	RT = O(n)

3. **Experimental Results:**

Table ALG1

n	**Theoretical RT N^2**	**EmpiricalRT (msec)**	**Ratio = (EmpiricalRT)/(TheoreticalRT)**	**Predicted RT(msec)**
10^3	10^6	12	$R_1 =$ 0.000012	27.35
$2*10^3$	$4*10^6$	48.6	$R_2 =$ 0.00001215	109.4
$3*10^3$	$9*10^6$	109.4	**$R_3 =$ 0.00002735 = c_1**	246.15
$4*10^3$	$16*10^6$	195.2	$R_4 =$ 0.0000122	437.6
$5*10^3$	$25*10^6$	302	$R_5 =$ 0.00001208	683.75
$6*10^3$	$36*10^6$	435	$R_6 =$ 0.00001	984.6
$7*10^3$	$49*10^6$	590.6	$R_7 =$ 0.00001	1340.15
$8*10^3$	$64*10^6$	782.2	$R_8 =$ 0.000012221	1750.4
$9*10^3$	$81*10^6$	987	$R_9 =$ 0.00001	2215
$10*10^3$	$100*10^6$	1202.8	$R_{10} =$ 0.000012028	2735
$11*10^3$	$121*10^6$	1449	$R_{11} =$ 0.00002	3309
$12*10^3$	$144*10^6$	1721.8	$R_{12} =$ 0.00001256	3938
$13*10^3$	$169*10^6$	2025.4	$R_{13} =$ 0.000011	4622
$14*10^3$	$196*10^6$	2344.2	$R_{14} =$ 0.00001	5360
$15*10^3$	$225*10^6$	2691	$R_{15} =$ 0.00001196	6153
$16*10^3$	$256*10^6$	3055.6	$R_{16} =$ 0.0000119359375	7001
$17*10^3$	$289*10^6$	3465.6	$R_{17} =$ 0.00001	7904
$18*10^3$	$324*10^6$	3895	$R_{18} =$ 0.0000131	8861.4
$19*10^3$	$361*10^6$	4365.6	$R_{19} =$ 0.000019902	9873
$20*10^3$	$400*10^6$	4724.2	$R_{20} =$ 0.0000118105	10940

ALG1: Runtime



I omitted r1 for c value because it seemed to be an outlier in the data.

Table ALG2

n	Theoretical RT $n\log_2 n$	EmpiricalRT (msec)	Ratio = (EmpiricalRT)/(TheoreticalRT)	Predicted RT(msec)
10^3	$10^3\log_2 10^3$	91.6	$R_1 =$ 0.0916	48.83234
$2*10^3$	$2*10^3\log_2 2*10^3$	101.6	$R_2 =$ 0.00463	107.46
$3*10^3$	$3*10^3\log_2 3*10^3$	170	**$R_3 =$ 0.00490 = c_2**	169.79
$4*10^3$	$4*10^3\log_2 4*10^3$	225.8	$R_4 =$ 0.00471	234
$5*10^3$	$5*10^3\log_2 5*10^3$	274	$R_5 =$ 0.00445	301
$6*10^3$	$6*10^3\log_2 6*10^3$	342.6	$R_6 =$ 0.00454	368
$7*10^3$	$7*10^3\log_2 7*10^3$	418.2	$R_7 =$ 0.00467	438
$8*10^3$	$8*10^3\log_2 8*10^3$	479.2	$R_8 =$ 0.00461	508
$9*10^3$	$9*10^3\log_2 9*10^3$	521	$R_9 =$ 0.00440	579
$10*10^3$	$10*10^3\log_2 10*10^3$	556	$R_{10} =$ 0.00418	651
$11*10^3$	$11*10^3\log_2 11*10^3$	587.8	$R_{11} =$ 0.00398	723
$12*10^3$	$12*10^3\log_2 12*10^3$	618.2	$R_{12} =$ 0.00380	796
$13*10^3$	$13*10^3\log_2 13*10^3$	655.2	$R_{13} =$ 0.00368	870
$14*10^3$	$14*10^3\log_2 14*10^3$	682.6	$R_{14} =$ 0.00354	944
$15*10^3$	$15*10^3\log_2 15*10^3$	705.4	$R_{15} =$ 0.00338	1019
$16*10^3$	$16*10^3\log_2 16*10^3$	731.4	$R_{16} =$ 0.00327	1094
$17*10^3$	$17*10^3\log_2 17*10^3$	753.2	$R_{17} =$ 0.00315	1170
$18*10^3$	$18*10^3\log_2 18*10^3$	776	$R_{18} =$ 0.00304	1246
$19*10^3$	$19*10^3\log_2 19*10^3$	804.2	$R_{19} =$ 0.00297	1323
$20*10^3$	$20*10^3\log_2 20*10^3$	846.4	$R_{20} =$ 0.00296	1400

ALG2: Runtime

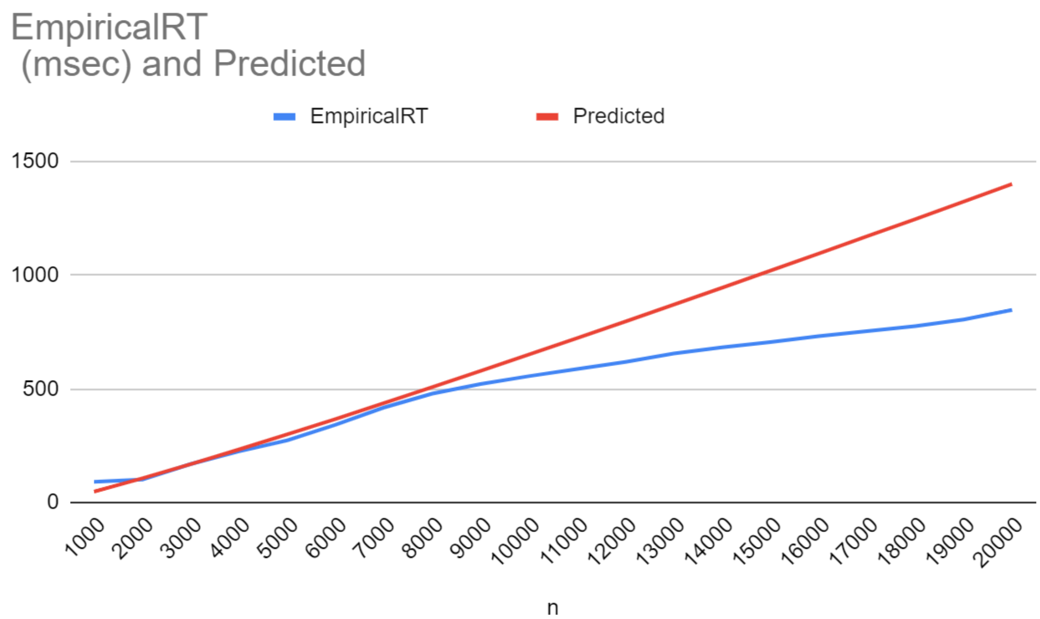
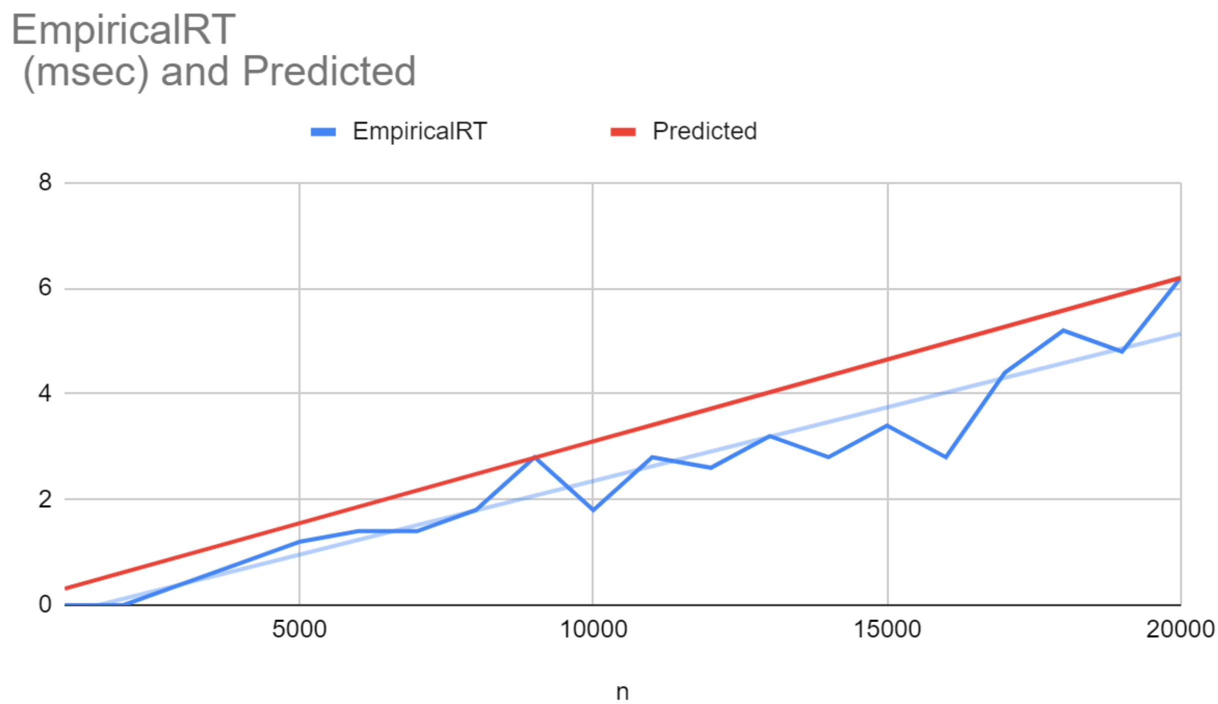


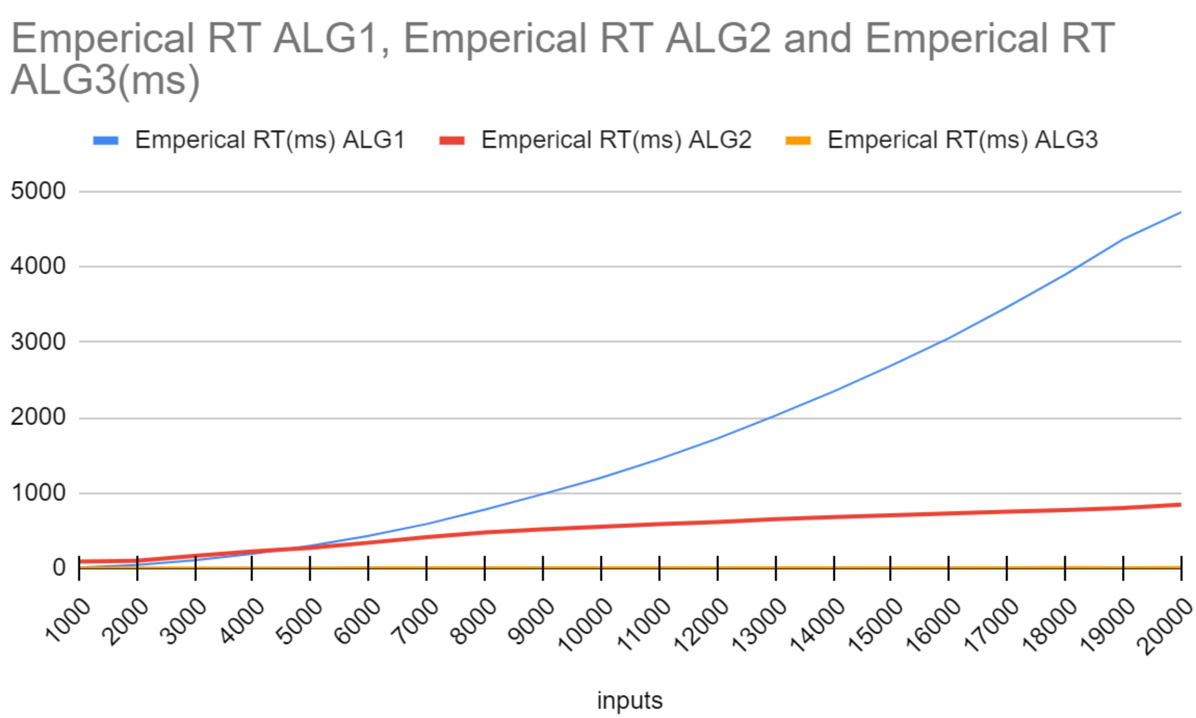
Table ALG3

n	**Theoretical RT N**	**EmpiricalRT (msec)**	**Ratio = (EmpiricalRT)/(TheoreticalRT)**	**Predicted RT(msec)**
10^3	10^3	0	$R_1 = 0$	0.31
$2*10^3$	$2*10^3$	0	$R_2 = 0$	0.62
$3*10^3$	$3*10^3$	0.4	$R_3 = 0.00013$	0.93
$4*10^3$	$4*10^3$	0.8	$R_4 = 0.0002$	1.24
$5*10^3$	$5*10^3$	1.2	$R_5 = 0.00024$	1.55
$6*10^3$	$6*10^3$	1.4	$R_6 = 0.00023$	1.86
$7*10^3$	$7*10^3$	1.4	$R_7 = 0.0002$	2.17
$8*10^3$	$8*10^3$	1.8	$R_8 = 0.00022$	2.48
$9*10^3$	$9*10^3$	2.8	$\mathbf{R_9 = 0.00031 = c_3}$	2.79
$10*10^3$	$10*10^3$	1.8	$R_{10} = 0.00018$	3.1
$11*10^3$	$11*10^3$	2.8	$R_{11} = 0.00025$	3.41
$12*10^3$	$12*10^3$	2.6	$R_{12} = 0.00021$	3.72
$13*10^3$	$13*10^3$	3.2	$R_{13} = 0.00024$	4.03
$14*10^3$	$14*10^3$	2.8	$R_{14} = 0.0002$	4.34
$15*10^3$	$15*10^3$	3.4	$R_{15} = 0.00022$	4.65
$16*10^3$	$16*10^3$	2.8	$R_{16} = 0.00017$	4.96
$17*10^3$	$17*10^3$	4.4	$R_{17} = 0.00025$	5.27
$18*10^3$	$18*10^3$	5.2	$R_{18} = 0.00028$	5.58
$19*10^3$	$19*10^3$	4.8	$R_{19} = 0.00025$	5.89
$20*10^3$	$20*10^3$	6.2	$R_{20} = 0.00031$	6.2

ALG3: Runtime



All Algorithms runtimes:



The speed for ALG3 is so fast compared to the others that it appears to be a straight line following the horizontal axis.

4. **Conclusions:**
   Through out the experiment we viewed the runtime for ALG1, ALG2, and ALG3. From the graphs we see the predicted worst-case scenarios. All the algorithms ran better than predicted, with ALG3 running the fastest and ALG1 running the slowest. This is to be expected because of the runtime for the sorting algorithms. With a smaller input ie. N = 1000, I was able to run the analysis. When the inputs ran up to 200000, the experiment would run for up to a few hours.
5. **References:**

   http://www.cplusplus.com/reference/
   Introduction to Algorithms, 3rd edition, by T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, The MIT Press, 2009, ISBN: 0262033844.