This exercise has been substantially done in Exercises 42, 43, and 44.

(a) The worst case data for bubble sort is a descending sequence (see solution to Exercise 43) and for selection sort is an ascending sequence terminated by a small value (see Exercise 44).

(c) The table can be derived from the data given in the solutions to Exercises 42, 43, and 44.

(d) For large n (say $n > 30$), selection sort has the best worst case performance followed by insertion sort. Bubble sort has the poorest worst case performance. For smaller n, selection sort and insertion sort are competitive.

(e) The overhead time included in the insertion sort and bubble sort measurements is the same. This is given in Figure 2.28 of the text. The selection sort overhead time is slightly larger (see below).

n	Repetitions	Total time	Overhead
0	37871	0.549451	0.000015
10	31676	0.549451	0.000017
20	26808	0.549451	0.000020
30	23185	0.549451	0.000024
40	20443	0.549451	0.000027
50	18174	0.549451	0.000030
60	16516	0.549451	0.000033
70	15039	0.549451	0.000037
80	13836	0.549451	0.000040
90	12797	0.549451	0.000043
100	11913	0.549451	0.000046
200	7026	0.549451	0.000078
300	4983	0.549451	0.000110
400	3860	0.549451	0.000142
500	3150	0.549451	0.000174
600	2660	0.549451	0.000207
700	2304	0.549451	0.000238
800	2027	0.549451	0.000271
900	1815	0.549451	0.000303
1000	1641	0.549451	0.000335

Times are in seconds

(f) The conclusions remain the same.

(g) Since the worst case complexity of each of the sort methods is $\Theta(n^2)$, we can estimate the worst case time needed to sort n numbers using that to sort 1000 numbers. $t(n)/t(1000) \sim n^2 / (1000)^2$. So $t(n) \sim n^2 t(1000)/10^6$. In particular, $t(2000) \sim 4t(1000)$, $t(4000) \sim 16t(1000)$ and $t(10000) \sim 100t(1000)$. Using these formulas and the measured values of $t(1000)$, we can estimate the required times.