1. True/False

T/F: When selecting an algorithm, the scale of the problem is unimportant. Our selection of algorithm should take into account whether we are manipulating 25 items or 25 million.

Answer: False, our primary concern is simplicity. We should strive to make readable code rather than very complex, but slightly more efficient code. If we are dealing with only 25 items, a simple, yet mediocre, algorithm is perfectly acceptable; when the program will likely encounter 25 million items, then we should focus on selecting a more efficient but complicated algorithm. (Page 300)

1. Multiple Choice

Which of the following has the orders of f(n) listed in ascending order?

1. 1, n2, n3, n\*log(n), 2n
2. 1, log(n), n\*log(n), n2, 2n
3. n!, log(n), n, n3, n\*log(n)
4. 1, n, n\*log(n), n2, n3
5. Options B and D
6. Options A and C

Answer: E. Options B and D both have the orders of magnitude listed in the correct order. (Page 296)

Note: n! was not discussed in the book, but it is of even higher order than 2n.

1. Fill in the Blank

“Algorithm A is O( f(n) ) if constants k and n0 exist such that A requires no more than k\*f(n) time units to solve a problem of size \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_”

Answer: n ≥ n0 (Page 294)

1. Short Answer or Code

Show the step by step simplification of the following growth-rate functions to the f(n) that would appear in the Big O notation for the algorithm. List the properties of growth rates that allow you to make your simplifications as you do so.

As part of an algorithm, a program must first perform a task that is:

O( 5n\*log(4n) + 70n – 15)

and an additional second operation that is:

O( 2n + n3 – 4n2 + 50 )

Answer:

1. Low order terms can be ignored

O( 5n\*log(4n) + 70n – 15) 🡪 O(5n\*log(n) )

O( 2n + n3 – 4n2 + 50 ) 🡪 O( 2n )

1. Multiplicative constants in the high order terms can be ignored

O(5n\*log(n) ) 🡪 O( n\*log(n) )

O( 2n ) 🡪 O( 2n )

1. Growth rate functions can be combined

O( n\*log(n) ) + O( 2n ) 🡪 O( 2n + n\*log(n) ) 🡪 O( 2n )

(By #1)

(Page 297-8)