Name: MUHAMAD SYUKRI BIN ISHAK

Class: CS230C

Test 1 CSC645

QUESTION 1

- An algorithm is a sequence of ambiguous instructions for solving a problem. (F)
- An algorithm must be rigorously and ambiguously specified. (F)
- Euclid algorithm is the algorithm for Euclidean distance. (F)
- Brute Force design technique always yields efficient and optimum solutions. (F)
- Asymptotic analysis is the analysis based on empirical perspective. (F)
- Exhaustive search falls under Brute force design technique for approximation problems. (T)
- Big Oh relates to upper bound whereas Big Omega relates to lower bound. (T)
- Divide and Conquer design technique typically divide a problem into two subproblems

and solve both subproblems accordingly. (T)

- Greedy design technique solves a problem piece by piece through a sequence of

choices. (T)

- Simplicity, generality and efficiency are the desirable characteristics of an algorithmic. (T)

Question 2

a)

Question 2

a. $i = 1$, $i <= n/2$, $i++$

$n = 8 \qquad n/2 = 8/2 = 4/2 = 2/2 = 1$

$n/2 = 2^k/2$

$n = 2^k$

$\log_2 n = \log_2 2^k$

$= k \log_2 2$

$= O(\log_2 n)$

$j = 1$, $j <= n/2$, $j++$

$n = 8 \qquad n/2 = 8/2 = 4/2 = 2/2 = 1$

$n = 2^k$

$\log_2 n = \log_2 2^k$

$= O(\log_2 n)$

inner loop × outer loop

$O(\log_2 n) \times O(\log_2 n)$

$= O(\log_2 n^2)$

Question 2

b)

b. Question 2

$i = 0, \; i < n, \; i = i + 2$

$\cancel{0}, \cancel{2}, \cancel{4}, \cancel{6}$

$$\sum_{i=0}^{n} i = 0 + 2 + 4 + \ldots\ldots n$$

$$= \frac{n(n+1)}{n}$$

$$= \frac{1}{2} n^2$$

$$= O(n^2)$$

Question 3

a)

Question 3

a) $T(n) = 2T(n/2) + n \quad n > 1 \quad T(1) = 0$

$$T(n) = 2T(n/2) + n$$

$$= 2[2T(n/2^2) + n/2] + n$$

$$= 2^2 T(n/2^2) + n + n$$

$$= 2^2 [2T(n/2^3) + n/2^2] + 2n$$

$$= 2^3 T(n/2^3) + 3n$$

$$T(n) = 2^k T(n/2^k) + kn$$

$$T(n) = 2^k T(1) + kn$$

$$\log_2 n = \log_2 2^k$$

$$\log_2 n = k \log_2 2$$

$$n/2^k = n$$

$$n \times \log_2 n$$

$$= O(n \log_2 n)$$

Question 3

b)

b) $T(n) = T(n-1) + n/2 \quad n > 1 \quad T(0) = 1$

$= T(n-1) + n/2$

$= T(n-1-1) + n/2 + n/2$

$= T(n-1-1-1) + n/2 + n/2 + n/2$

$= \sum_{i=1}^{n} i = 0 + 1/2 + 2/2 + 3/2 \ldots\ldots n/2$

$= \frac{n(n+1)}{2}$

$= \frac{1}{2} n^2$

$= O(n^2)$

Question 3

c)

Question 3

c. $T(n) = T(n-1) + 100 \quad n > 1 \quad T(0) = 1$

$= T(n-1-1) + 100 + 100$

$= T(n-2-1) + 100 + 100 + 100$

$= \sum_{i=1}^{n} i = 100 + 100 + 100 \ldots\ldots n$

$= n - 1 + 1$

$= n$

$= O(n)$

Question 4

a. For worst-case time efficiency of Sequential search

= O(n)

b. For worst-case time efficiency of Binary Search

= O(log n)

c. For best-case time efficiency of Sequential search

= O(1)

d. For best-case time efficiency of Binary Search

= O(1)

e. For the case where the search key is not found using Sequential search