

MIDTERM

1. Determine the time complexity for the following code fragments in terms of the $\Theta()$ notation. Explain your justification for each case. (You do not need to give a formal proof, but you must explain your answers briefly.)

- a) `a = b + c`
`d = a + e`
- b) `sum = 0`
`for i=1 to 3 do`
 `for j=1 to n do`
 `sum = sum + 1`
 `endfor`
`endfor`
- c) `sum = 0`
`for i=1 to n*n do`
 `sum = sum + 1`
`endfor`
- d) (A[1:n] is an array; Random is a built-in function that takes constant time; module Sort takes $n \cdot \log n$ time.)
`for i=1 to n do`
 `for j=1 to n do`
 `A[i] = Random`
 `endfor`
 `Call Sort (A[1:n])`
`endfor`
- e) (A[1:n] is an array that contains a random permutation of integers from 1 to n.)
`sum = 0`
`for i=1 to n do`
 `j = 1`
 `while A[j] <> i do`
 `j = j + 1`
 `endwhile`
 `sum = sum + 1`
`endfor`
- f) (Even is a boolean function that takes constant time; n is an even number.)
`sum = 0`
`for i=1 to n do`
 `if even(i) then`
 `for j=1 to n do`
 `sum = sum + 1`
 `endfor`
 `else`
 `sum = sum + n`
 `endif`
`endfor`

2. (Assignment question) You are given the following function:

function Q1(X[1:n])

Input: X[1:n] array of elements with $X[i] \in \{0,1\}$, $1 \leq i \leq n$, n is even

Output: a (an integer)

```

b = 1
a = 0
for i = 1 to n do
    for j = i to n do
        a = a + X[i]
    endfor
    if X[i] ≠ X[n-i+1] then
        b = 0
    endif
endfor
if b = 1 then
    for i = 1 to n do
        for j = 1 to n do
            a = a + a * X[j]
        endfor
        for j = 1 to 2n/2 do
            a = a + 1
        endfor
    endfor
endif
end

```

Do a fine analysis of the function and express the worst-case $W(n)$ and average case $A(n)$ of the algorithm using Θ notation. You may consider all the assignment statements in the algorithm as the basic operation and the input as a binary string.

3. Solve the following recurrence relations:

a) $T(n) = T(9n/10) + n$, $T(1)=0$. Assume that $n=(10/9)^k$, $k \geq 1$. Use backward substitution.

b) $T(n) = 2 T(n/4) + \sqrt{n}$. Use Master Theorem.

c) $T(n) = T(\sqrt{n}) + 1$, $T(2)=0$. You may assume a suitable value for n . Use backward substitution.

4. Modify the binary search routine to support search in an array of infinite size. That is, you are given as input a sorted array and a key value X to search for. Assume that X occurs in the array. Call n the position of X in the array. Write an algorithm *in pseudocode* that can determine n in $O(\log n)$ time in the worst case. Analyze the complexity of your algorithm. (Note that n is not the data size; it is the position of the search element.)

Notes:

- Questions 1-4:25 points
- Time: 1:45 hours
- Close notes and books

Belkim bir kertenkeleydim
Piç edilmiş bir yağmurun serini
Bir güzelin çirkiniydim
Çirkinlerin en güzeli
Yeşil koşsa güneşlerin gölgesi
Ben en hızlı yeşiliydim
Kurbağa yarışlarında annemin

Çatal matal kaç çataldım kimbilir
Bin dereden bir kendimi getirdim
Haydan gelip huya giden bir huysuz
Heyheyler içinde bir heydim
Belkim yedi belkim sekiz belaydım

Düdük çalar hırsızlanmış polisler
Ben korkudan üstlerime işerdim
Üç yıldızlı bir albaydı gökyüzü
Karşısında önüm açık gezerdim
Ağzı bozuk meymenetsiz bir ozan

Rus cenginde çağanozdum bir zaman

İki gözüm iki koltuk-eviydi
Mavilerim bir miyobun koynunda
Kendi düşen köyler kentler ağlamaz
Sur dışında ben oturur ağlardım
Ekmek diye bağırsırdı bebeler
Elma derler ben ortaya çıkardım
Ağıtlarla kutlanırdı İsa - doğdu Gecesi
Fil dışından bir kuleydım yıktım kendimi

Bilmem hangi keloğlanın fesiydim
Bir püskülsüz sümbülteber tohumu
Fesleğenler yaprak dökmüş şerrimden
Bir naraydım kimse bilmez nereden
Ya yakından ya uçmaktan gelirdim
Belkim ince belkim kalın bir sestim
Belkilerin kol gezdiği saatta
Belkim belki bile değildim