

1. This exam includes 5 questions.
 2. Time for this exam is **100 minutes**.
 3. Do not forget to **write down your name and number on each page.**
 4. Each question is worth as many points as denoted and distributed at its top.
 5. This is an **open-book, open-notes exam.**
 6. Please answer the questions in empty spaces left below or aside the questions.
 7. **Use of internet or any communication with anyone throughout the exam are NOT allowed!**
 8. You may get partial points. **Work neatly and show all your work.**
 9. **Upload the signed statement of righteousness and solutions in a single pdf file (name_surname.pdf)!!!**
 10. Your **handwritten solutions** may be either on the printed exam papers I uploaded or on empty A4 sheets.
 11. **Make sure that your answers are readable and the signed statement of righteousness is there!!!**
- Good luck!!! ☺

!!! SIGN THE STATEMENT OF RIGHTEOUSNESS BELOW !!!

Karadut

Karadutum, çatal karam, çingenem
Nar tanem, nur tanem, bir tanem.
Ağaç isem dalımsın salkım saçak
Petek isem balımsın ağılüm
Günahımsın, vebalımsın.

Dili mercan, dizi mercan, dişi mercan
Yoluna bir can koyduğum
Gökte ararken yerde olduğum.
Karadutum, çatal karam, çingenem
Daha nem olacaktın bir tanem.
Gülen ayvım, ağlayan narımsın.
Kadınım, kırsığım karımsın.

Bedri Rahmi Eyuboğlu

Saklambaç

Geminin yarısından çoğu
Suyun içinde gizli.
Ağacın kökleri toprağın içinde
Boyundan büyük!!!... Gizli!...
Geceler gündüzlerden uzun
Karanlık... Zifiri... Gizli!...
Sır tutan kişi namuslu
Namuslu kişi..... Gizli!...
Alan gizli... Veren gizli...

Desene saklambaç oynamağa gelmişiz
bu dünyaya!!!

Bedri Rahmi Eyuboğlu

Vatan İçin.....

Neler yapmadık şu vatan için!
Kimimiz öldük;
Kimimiz nutuk söyledik.

Orhan Veli Kanık

Hay Kay.....

Yosun kokusu
Ve bir tabak karides
Sandıkburnu'nda

Orhan Veli Kanık

Gemliğe doğru

Denizi göreceksin;
Sakın şaşırma!

Orhan Veli Kanık

Anadolu

Beşikler vermişim Nuh'a,
Salıncaklar, hamaklar;
Havva Anan dünkü çocuk sayılır,
Anadoluyum ben.
Tanıyor musun?

Ahmed Arif

Geçme namert köprüsünden,
Ko aparsın su seni.
Yatma tilki gölgesinde,
Ko yesin aslan seni.

Diyarbakırlı Sait Paşa

Statement of righteousness:

I hereby pledge that I have answered the questions in this exam based only upon my own knowledge, with no communication with any classmate or anyone else, no help from internet or else any other source whatsoever but the lecture notes, the textbooks I was allowed to use; that I did never violate any of the above rules of the exam or take any other inappropriate action that will require any disciplinary precaution.

Name:

Signature:

Q1. (10+15=25pts)**Attention:** The parts (a) and (b) of this question are independent of each other!!!

- a) Consider the recursive function below! Find, **in terms of m** , the recurrence (i.e., the difference equation) that represents the *execution time* $f(m)$ for `recfunc`.

```
void recfunc(int m) {  
    if (m < 9) {  
        return;  
    }  
    recfunc(m/3);  
    for (int j = 0; j < m; j++) printf("%d\n", j);  
    for (int j = 0; j < 3; j++) recfunc(m/9);  
    for (int j = 0; j < m; j++) printf("%d\n", j);  
    recfunc(m/3);  
    for (int j = 0; j < m; j++) printf("%d\n", j);  
}
```

 $f(m)=$

- b) Find the particular solution for the following recurrence with the given initial values! Also give the asymptotic upper bound ($O()$). **Show all your work!**

$$t(n) = t(n-2) + n, \quad t(0) = t(1) = 0;$$

Q2. (20 pts)

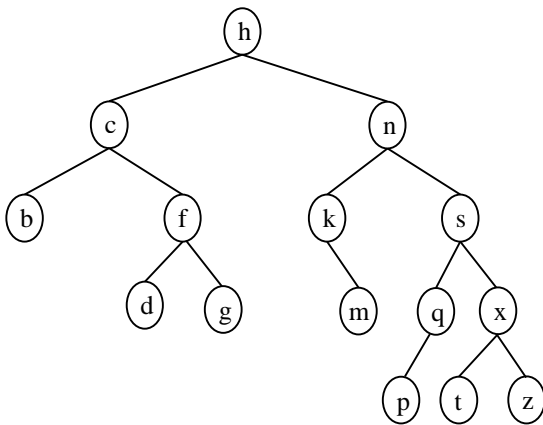
Suppose five algorithms $A_1 \dots A_5$ solving the same problem with execution times, $t_1(n), \dots, t_5(n)$, of the following asymptotic upper bounds. Put the algorithms in order regarding their execution speed from the fastest to the slowest!

- i. $O(2^{n+2})$
- ii. $O([n^n]^2)$
- iii. $O(4^n)$
- iv. $O(n^2!)$
- v. $O(\log n^n)$
- vi. $O(n^{[n^2]})$

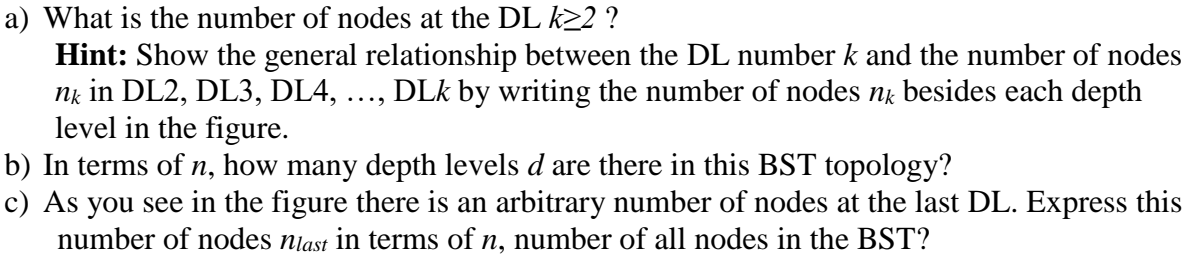
Q3. (10+10+10 = 30pts)

Given is the BST topology below. The **letters are the keys of the BST considering their alphabetic order**.

- a) Check whether it is an AVL tree, *showing the subtree heights for each node and clearly state your conclusion!* If any violation, state the α node and the type of violation and restore the AVL property!
- b) Insert the key *y* into the BST! Test for the new tree's AVL property by
 - i. showing the nodes to be checked in the order from the first to the last and, for each, its subtrees' heights;
 - ii. if any violation, stating the α node, the type of violation and the way to restore it;
 - iii. and drawing the topology with the AVL property restored using the letters.
- c) Insert the key *e* into the **new AVL tree from part (b)**! Test for the new tree's AVL property by going through the same procedure as in part (b) explained in (i)-(iii).



Assume that you have a BST with n nodes as shown in the figure below. This is *the best possible BST topology with the missing right subtree of the root's left child*. The dots over the last depth level (DL) (● ● ●) indicate an arbitrary number of depth levels all filled up. The only nodes that are missing compared with a complete binary tree are those that are a member of the right subtree of the root's left child.



Q5. (10 pts)

i/j	1	2	3	4	5	6	7
1	1	0	0	0	0	0	-47
2	0	12	0	0	0	-20	0
3	0	0	45	0	29	0	0
4	0	0	0	112	0	0	0
5	0	0	241	0	225	0	0
6	0	428	0	0	0	396	0
7	685	0	0	0	0	0	637

Given is the above matrix m . In order to represent m most efficiently (allocating minimum amount of memory space)) provide **a partial function formula** (NOT code!!!) in terms of i and j . You can use the partial function formula I provided for you below.

$$m_{i,j} = \begin{cases} & \text{for} \\ & \text{for} \end{cases}$$