KING SAUD UNIVERSITY

COLLEGE OF COMPUTER & INFORMATION SCIENCES
DEPT OF COMPUTER SCIENCE

CSC311 Computer Algorithms
First Semester 1437/1438 AH (Fall 2016)
Second Mid-term Examination: Tue 6.12.2016 C.E. (duration = 1.5 hours)
Instructor: Dr. Aqil Azmi

NAME: **ID:**

1. **[Marks 20=12+8]**
Solve the recurrence relation: $T(n) = 5T(n/6) + 3n$. Use **(a)** repeated substitution method; and **(b)** master table.

$$T(n) = 5T(n/6) + 3n$$

$$= 5[5T(n/6^2) + 3(n/6)] + 3n$$

$$= 5^2 T(n/6^2) + 5 \times 3(n/6) + 3n$$

$$= 5^2[5T(n/6^3) + 3(n/6^2)] + 5 \times 3(n/6) + 3n$$

$$= 5^3 T(n/6^3) + 5^2 \times 3(n/6^2) + 5 \times 3(n/6) + 3n$$

$$\vdots$$

$$= 5^k T(n/6^k) + 3n \sum_{i=0}^{k-1} \left(\frac{5}{6}\right)^i$$

$$\leq 5^k T(n/6^k) + 3n \cdot \frac{1}{1 - 5/6}$$

$$\Downarrow$$

$$k = \log_6 n$$

$$T(n) = 5^{\log_6 n} T(1) + 15n$$

$$= n^{\log_6 5} T(1) + 15n$$

$$= O(n)$$

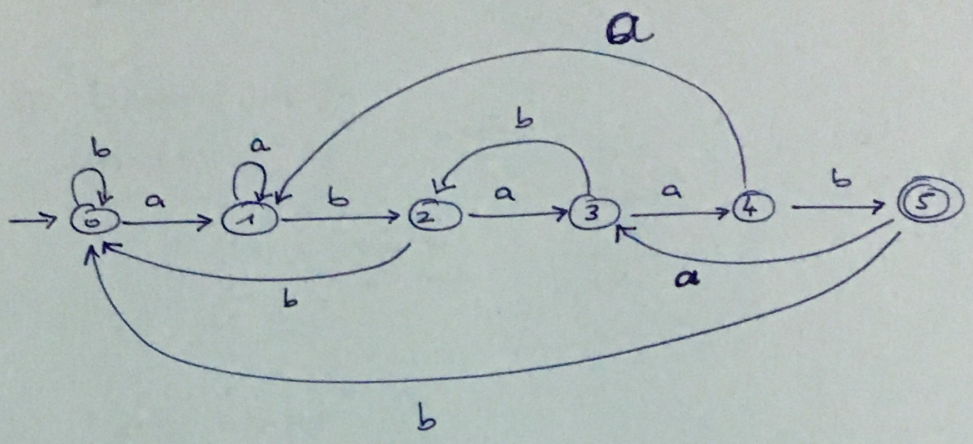
MT

$a = 5 \quad b = 6 \quad f(n) = 3n$

$$h(n) = \frac{f(n)}{n^{\log_b a}} = \frac{3n}{n^{\log_6 5}} = n^{>0}$$

$$T(n) = O(f(n))$$
$$= O(n)$$

2. [Marks 20=14+6]
Design **(a)** Finite-state Automaton to match the pattern $P = abaab$, and **(b)** show how the matching is done on text $T = babaaabaabaabb$.



T = b a b a a a b a a b a a b b

0 0 1 2 3 4 1 2 3 4 5 3 4 5 0

3. [Marks 30=5+10+5+10]

a. For binary heap, give formula for *parent* and the *child* of node *k*.
b. For heap (رباعي), give formula for *parent* and all 4 *children* of node *k*.
c. What is the *exact* height of a binary heap with $n = 100$ nodes.
d. What is the *exact* height of a رباعي heap with $n = 100$ nodes.

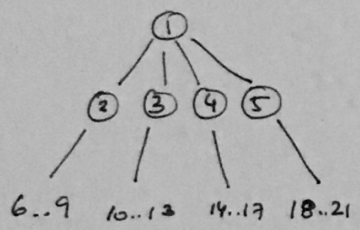
a) binary heap

node k $\xrightarrow{\text{children}}$ $2k$, $2k+1$

$\xrightarrow{\text{parent}}$ $\lfloor k/2 \rfloor$

b) رباعي heap

node k $\xrightarrow{\text{children}}$ $4k-2$, $4k-1$, $4k$, $4k+1$



$\xrightarrow{\text{parent}}$ $\left\lfloor \frac{k+2}{4} \right\rfloor$

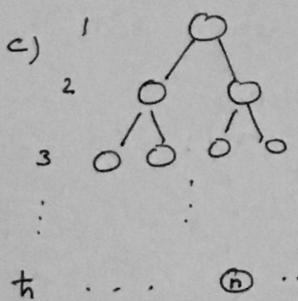
$k' \geq 4k-2$

$k'+2 \geq 4k$

$\frac{k'+2}{4} \geq k$

$k = \left\lfloor \frac{k+2}{4} \right\rfloor$

c)



1, 2, 2^2, ..., 2^{h-1}

$\frac{2^h - 1}{2^h - 1}$

$n \leq 2^h - 1$

$\Rightarrow 2^h \geq n+1$

$h = \lceil \log(n+1) \rceil$

for $n = 100 \Rightarrow$ height $h = \lceil \log 101 \rceil = 7$

d) Similarly $n \leq \sum_{i=0}^{h-1} 4^i = \frac{4^h - 1}{3}$

$3n \leq 4^h - 1 \Rightarrow$ height $h = \lceil \log_4(3n+1) \rceil$

for $n = 100 \Rightarrow$ height $h = \lceil \log_4(301) \rceil =$

4. [Marks 30=15+15]

a. Write the code of the binary search.

b. Write a *non-recursive* version of the binary search.

a)

```
procedure bs(a[], i, j, x)
{
    // search for x in a[i..j]
    if (i ≤ j)
    {
        m ← ⌊(i+j)/2⌋
        if (a[m] = x) return m
        else if (a[m] > x)
            return bs(a, i, m-1, x)
        else
            return bs(a, m+1, j, x)
    }
    return -1
}
```

b)

```
procedure bs(a[], x)
{
    i ← 1
    j ← n          // = size of a[]
    while (i ≤ j) do
    {
        m ← ⌊(i+j)/2⌋
        if (a[m] = x) return m
        else if (a[m] > x)
            j ← m-1
        else
            i ← m+1
    }
    return -1
}
```