Week 4. Problem set

1. Compute asymptotic worst case time complexity of `solve` in terms of the size n of the input list:

   (i) Express the running time of the `solve` procedure $T(n)$ as a recurrence relation.

   (ii) Find the asymptotic complexity of $T(n)$ using the master theorem.

   (iii) Specify which case of the master theorem applies (if any).

   (iv) Write down conditions for this case that need to be checked (write down specialized version for a particular recurrence).

   (v) Prove that the conditions are satisfied. For asymptotic notation, each property used in a proof **must** be either proven explicitly or properly referenced (e.g. by citing [CLRS] or a particular Lecture/Tutorial slide).

```
1   /* L is a linked list */
2   solve(L):
3     if L is empty
4       return 1
5     else
6       s = 1
7       for j from 1 to 3 /* inclusive */
8         B = new linked list
9         i = 0
10        A = L.head
11        while A.next is not null
12          b = (i >> j) & 3   /* jth and (j+1)th least significant bits of i */
13          if b == 1
14            B.add(i / 2, A.value) /* add A.value at position (i/2) in B */
15          A = A.next
16          i = i + 1
17        s = s + solve(B)
```

2. Consider the following modification of the binary search algorithm [CLRS, Exercise 2.3-6]: given an input array A of size n, a parameter k, and an integer value x

   (a) if the array is empty — stop with failure (x not found)

   (b) if the size of the input array is 1 and $A[0] = x$ — stop with success (x is found)

   (c) otherwise, we create midpoints m_i (for all integer indices $1 \leq i \leq k$) *evenly* spaced between start index $m_0 = 0$ and end index $m_{k+1} = n - 1$

   (d) we then find consecutive indices m_i and m_{i+1} such that $A[m_i] \leq x \leq A[m_{i+1}]$ and repeat the search on the subarray $A[m_i : m_{i+1}]$

   Compute the time complexity of this algorithm in terms of n and k using two different methods:

   (i) Write down the running time $T(n)$ as a recurrent formula.

   (ii) Apply the master theorem and get closed form formula for $T(n)$ (assume k is a constant, the final answer may be independent of k):

      i. Which case of the master theorem applies?

      ii. Write down the conditions for this case that need to be checked (write down specialized version for this particular recurrence).

      iii. Provide asymptotic complexity for $T(n)$ using Θ-notation.

   (iii) Apply the substitution method and get closed form formula for $T(n, k)$ (treat k as a parameter, the final answer should depend on k).

(iv) Which value of k is optimal? Justify your answer (4–6 sentences).

3. For each of the following recurrences, apply the master theorem [CLRS, Theorem 4.1] yielding a closed form formula for the asymptotic complexity of $T(n)$. Assume that $T(n) = 1$ when $n < 10$ for all recurrences below. You must specify the following:

- Can the theorem be applied to the recurrence?
- If yes, then
    (i) Which case of the master theorem applies?
    (ii) Which conditions for this case that need to be checked (write down specialized version for a particular recurrence).
    (iii) Prove that the condition is satisfied. For asymptotic notation, each property used in a proof **must** be either proven explicitly or properly referenced (e.g. by citing [CLRS] or a particular Lecture/Tutorial slide).
    (iv) Provide asymptotic complexity for $T(n)$ using Θ-notation.
- Otherwise, provide an explicit justification, explaining why the theorem cannot be applied.

(a) $T(n) = 6T(n/36) + \sqrt{n} \cdot \log_2^2 n$
(b) $T(n) = 4T(3n/8) + n^2$
(c) $T(n) = 7T(n/5) + \log_2 (n!)$
(d) $T(n) = 3T(\sqrt[3]{n}) + n$
(e) $T(n) = \frac{1}{3}T(3n) + n \log_3 n$

4. **(+1% extra credit)** Consider the following recurrence with parameters $a > 1$ and $b > 1$:

$$T(n) = aT(n/b) + n^{4\log_a b} \ln^{a-2b} n$$

Specify **precise** conditions on a and b such that

(i) Case 1 of the master theorem applies to $T(n)$
(ii) Case 2 of the master theorem applies to $T(n)$
(iii) Case 3 of the master theorem applies to $T(n)$
(iv) The master theorem is not applicable to $T(n)$

Provide a brief justification in each case.

References

[CLRS]   Cormen, T.H., Leiserson, C.E., Rivest, R.L. and Stein, C., 2022. *Introduction to algorithms, Fourth Edition.* MIT press.