# CSCI-GA 1170 Homework 1

## Question 1

### Lower bound

According to ITB, $S(1000) \geq \lg(1000!)$. Stirling's approximation tells us

$$n! = (\frac{n}{e})^n \sqrt{2\pi n} e^{\alpha_n}$$

$\lg(n!) = n(\lg n - \lg e) + \frac{1}{2}\lg(2\pi n) + \alpha_n \lg e \approx 8527.46$

Therefore, we conclude that $S(1000) \geq 8528$

### Upper bound

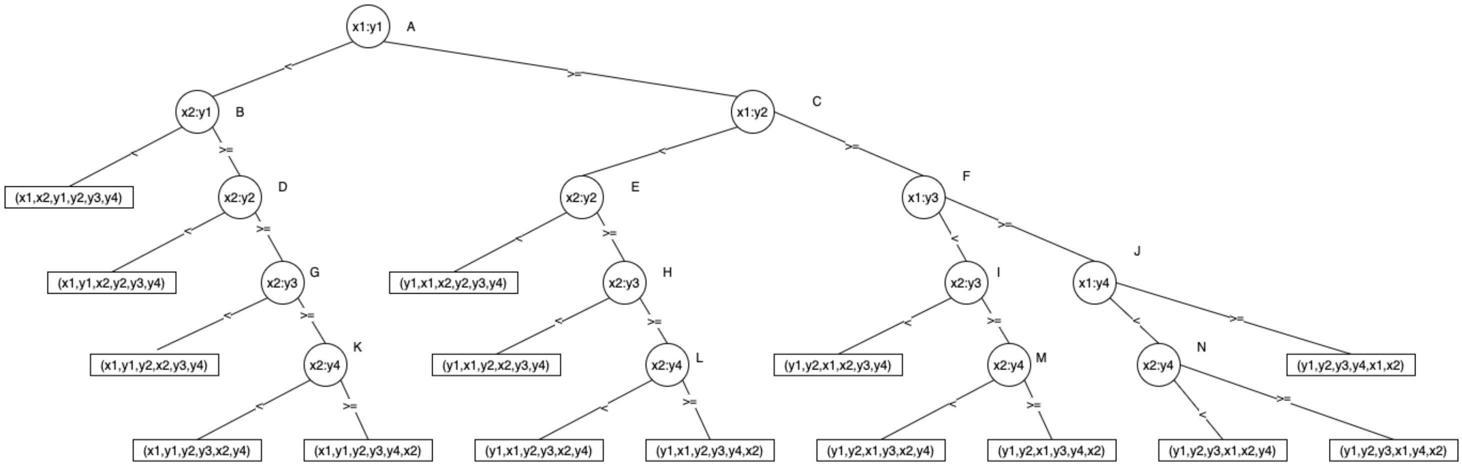
The recurrence in merge sort algorithm is $T(n) = 2T(n/2) + n$. We can get an upper bound as follows:

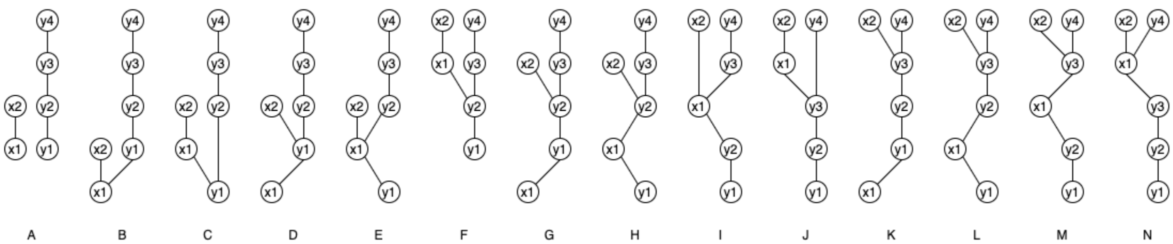
$$S(1000) < S(1024) \leq T(2^{10}) \leq \sum_{i=0}^{9} 2^{10-i} \times 2^i = 10240$$

Therefore, we conclude that $S(1000) \leq 10240$

# Question 2

**The Hasse diagrams of the leaf nodes are linear and are given by the merge results.**



# Question 3

**TAPE SORTING ALGORITHM:**

TAPESORT($T_0, T_1, T_2$):

- tape_changed $\leftarrow$ True
- while tape_changed:
  - tape_changed $\leftarrow$ DISTRIBUTE($T_0, T_1, T_2$)
  - MERGE($T_0, T_1, T_2$)

DISTRIBUTE($T_0, T_1, T_2$):

- RESET($T_0$),RESET($T_1$),ERASE($T_1$),RESET($T_2$),ERASE($T_2$)
- is_finished $\leftarrow$ False
- last $\leftarrow -\infty$
- change_tape $\leftarrow$ False
- tape_changed $\leftarrow$ False
- if EOT($T_0$):
  - is_finished $\leftarrow$ True
- else:
  - READ($T_0$, $x$)
- while (! is_finished):
  - if x < last:
    - change_tape = ! change_tape
    - tape_changed $\leftarrow$ True
  - if ! change_tape:
    - WRITE($T_1$, $x$)
  - else:
    - WRITE($T_2$, $x$)
  - if EOT($T_0$):
    - is_fininshed = True
  - else:
    - last $\leftarrow$ x
    - READ($T_0$,$x$)
- return tape_changed

MERGE($T_0, T_1, T_2$):

- RESET($T_0$),ERASE($T_0$),RESET($T_1$),RESET($T_2$)
- t1_finished $\leftarrow$ t2_finished $\leftarrow$ False
- t1_last $\leftarrow$ t2_last $\leftarrow -\infty$
- t1_new $\leftarrow$ t2_new $\leftarrow$ False
- if EOT($T_1$): t1_finished $\leftarrow$ True else: READ($T_1, x_1$)
- if EOT($T_2$): t2_finished $\leftarrow$ True else: READ($T_2, x_2$)
- while ! t1_finished and ! t2_finished:
    - if $x_1 \leq x_2$:
        - WRITE($T_0, x_1$)
        - if EOT($T_1$):
            - t1_finished $\leftarrow$ True
            - t1_new $\leftarrow$ True
        - else:
            - t1_last $\leftarrow x_1$
            - READ($T_1, x_1$)
            - if $x_1$ < t1_last:
                - t1_new $\leftarrow$ True
    - else:
        - WRITE($T_0, x_2$)
        - if EOT($T_2$):
            - t2_finished $\leftarrow$ True
            - t2_new $\leftarrow$ True
        - else:
            - t2_last $\leftarrow x_2$
            - READ($T_2, x_2$)
            - if $x_2$ < t2_last:
                - t2_new $\leftarrow$ True
    - while t1_new:
        - WRITE($T_0, b_2$)
        - if EOT($T_2$):
            - t1_new $\leftarrow$ False
            - t2_finished $\leftarrow$ True
        - else:
            - t2_last $\leftarrow x_2$
            - READ($T_2, x_2$)
            - if $x_2$ < t2_last:

- t1_new ⟵ False
- while t2_new:
  - **Repeat the previous while loop for $T_1$**
- while ! t2_finished:
  - WRITE($T_0, b_2$)
  - if EOT($T_2$):
    - t2_finished ⟵ True
  - else:
    - READ($T_2, x_2$)
- while ! t1_finished:
  - **Repeat the previous while loop for $T_1$**

# Question 4

By definition, if $g \in \theta(f) = o(f) \cap \omega(f)$, we have $(\forall C > 0)[Cf \geq g \geq f/C \text{ (ev.)}]$. Clearly, the definition of $\theta(\cdot)$ is not meaningful, because if $f > 0 \text{ (ev.)}, \theta(f) = \phi \text{ (empty set)}$.

# Question 5

## (i) [Counter example]

$f(x) = (1 + \sin(\frac{\pi}{2}x))x$

Note that when $x = 4k + 1, k \in \mathbb{N}$, we have $f(x) = 2x$. Therefore, $f(x)$ is unbounded. However, when $x = 4k + 3, k \in \mathbb{N}$, we have $f(x) = 0$. Thus, "$f \succ\succ 1$" does not hold.

## (ii) [Proof]

By definition,

$$(\forall C > 0)[f > C \cdot g \text{ (i.o.)}] \equiv (\forall C > 0)(\forall x_0)(\exists x)[x > x_0 \wedge f(x) > C \cdot g(x)]$$

$$\begin{aligned}
\neg(f \preceq g) &\equiv \neg((\exists C > 0)[f \leq C \cdot g \text{ (ev.)}]) \\
&\equiv \neg((\exists C > 0)(\exists x_0)(\forall x)[x > x_0 \rightarrow f(x) \leq C \cdot g(x)]) \\
&\equiv \neg((\exists C > 0)(\exists x_0)(\forall x)[x \leq x_0 \vee f(x) \leq C \cdot g(x)]) \\
&\equiv (\forall C > 0)(\forall x_0)(\exists x)[x > x_0 \vee f(x) > C \cdot g(x)] \\
&\equiv (\forall C > 0)[f > C \cdot g \text{ (i.o.)}]
\end{aligned}$$

# Question 6

## (A)

(A-i) $f > 0 \text{ (ev.)}$

(A-ii) $f \succeq 0 \equiv f \geq 0 \text{ (ev.)}$

(A-iii) $f \succ 0 \equiv f \geq 0 \text{ (ev.)} \wedge f > 0 \text{ (i.o.)}$

(A-iv) $f \succ\succ 0 \equiv f \geq 0 \text{ (ev.)}$

In conclusion, (A-i)⇒(A-iii)⇒(A-ii)⇔(A-iv)

## (B)

(B-i) $f > 1 \text{ (ev.)}$

(B-ii) $f \succeq 1$

(B-iii) $f \succeq 1 \wedge (\forall C > 0)(f \geq C \text{ (i.o.)})$

(B-iv) $f \succ\succ 1$

In conclusion, (B-iv)⇒(B-iii)⇒(B-ii); (B-iv)⇒(B-i)⇒(B-ii)

# Question 7

- (a) counter-example $f(x) = x, g(x) = x^2 (x \in \mathbb{R})$
- (b) counter-example $f(x) = e^x (x \leq 0), g(x) = x + 1 (x \in \mathbb{R})$
- (c) counter-example $f(x) = 100, g(x) = 1 + \frac{1}{x}$
- (d) True
- (e) counter-example $f(x) = 2x, g(x) = x$
- (f) True
- (g) counter-example $f(x) = x^{-1}$
- (h) counter-example $f(x) = e^x (x \in R)$

# Question 8

## [Expansion]

$$\begin{aligned} T(n) &= 3T(n/2) + n \\ &= 9T(n/4) + (\frac{3}{2}n + n) \\ &= 27T(n/8) + (\frac{9}{4}n + \frac{5}{2}n + n) \end{aligned}$$

## [Guessing]

$$G_i = T(n) = 3^i T(n/2^i) + \sum_{j=0}^{i-1} (\frac{3}{2})^j \cdot n$$

## [Verification]

### [RB]

Note that T(n) is true for $i = 1$.

### [RI]

$$\begin{aligned} T(n) &= 3^i T(n/2^i) + \sum_{j=0}^{i-1} (\frac{3}{2})^j \cdot n \\ &= 3^i (3T(n/2^{i+1}) + \frac{n}{2^i}) + \sum_{j=0}^{i-1} (\frac{3}{2})^j \cdot n \\ &= 3^{i+1} T(n/2^{i+1}) + (\frac{3}{2})^i n + \sum_{j=0}^{i-1} (\frac{3}{2})^j \cdot n \\ &= 3^{i+1} T(n/2^{i+1}) + \sum_{j=0}^{i} (\frac{3}{2})^j \cdot n \end{aligned}$$

This confirms that $G_{i+1}$ holds. Then we have

$$T(n) = 3^i T(n/2^i) + (\frac{4}{3} \cdot (\frac{3}{2})^i - 1)n$$

## [Stop]

Choose $i = \lceil \lg n \rceil$. According to DIC, choose the initial condition to be $T(n) = 0$ for $0 < n \le 1$. This yields the exact solution for $n > 0$.

$$T(n) = \begin{cases} 0 & \text{if } n \le 1 \\ (\frac{4}{3} \cdot (\frac{3}{2})^{\lceil \lg n \rceil} - 1)n & \text{o.w.} \end{cases}$$

# Question 9

## (a)

Exponential type.
Let $C = 2$, then we have $2^{(x-1)^2+1} = 2^{x^2} \cdot 2^{2-2x} \leq 2^{x^2}$ (ev.)

## (b)

Polynomial type.
Accoring to Lemma 8, since $f(x) = x$ is polynomial type and $\lg x$ is non-decreasing, $\lg x$ is polynomial type. Similarly, $\lg \lg x$ is also polynomial type. Since polynomial type functions are closed raising to any positive power, therefore $(\lg \lg x)^2$ is polynomial type.

## (c)

Polynomial type.
Let $f(x) = x/\lg x$ and $C = 2$, then $2f(x/2) = \frac{x}{\lg x - \lg 2} \geq \frac{x}{\lg x} = f(x)$ (ev.)

# Question 10

$T(n) = \sum_{i=0}^{\lg(n+1)-1} (i+1)^2 \cdot 2^i = \Theta(\lg^2(n+1) \cdot 2^{\lg(n+1)-1}) = \Theta(n \lg^2 n)$

# Question 11

By corollary 11, $T_1(n) = 6T_1(n/2) + n^3 = \Theta(n^3)$, and $T_2(n) = 8T_2(n/2) + n^2 = \Theta(n^3)$. Thus, they are growing at the same rate.

# Question 12

$T(n) = n + 3T(n/2) + 2T(n/3) \geq S(n) = n + 5T(n/3)$. According to Master's theorem, $S(n) = \Theta(n^{\log_3 5})$. Thus, we have $T(n) = \Omega(n^{\log_3 5})$. So, Joe's claim $T(n) = O(n)$ is wrong.

We use real induction to prove that Jane's claim $T(n) = O(n^2)$ is correct. We are going to show that given $x_0 \geq 1$ and $K > 0$, $x \geq x_0 \Rightarrow T(x) \leq Kx^2$.

**[Basic]**
By DIC, let $T(x_1) = C$, then for all $x_0 \leq x < x_1, T(x) < C$.

Let $K = \frac{C}{x_0^2} > 0$, then $x_0 \leq x < x_1 \Rightarrow T(x) < C = Kx_0^2 \leq Kx^2$

**[Induction]**
Let $x_1 = 3x_0, \delta = \frac{1}{2}x_1$. For all $x \geq x_1$, $x_0 = x_1/3 \leq x/3 < x/2 = x - x/2 \leq x - x_1/2 = x - \delta$.

Therefore,

$$\begin{aligned}
T(x) &= x + 3T(x/2) + 2T(x/3) \\
&\leq x + 3K(\frac{x}{2})^2 + 2K(\frac{x}{3})^2 \\
&= x + \frac{35}{36}Kx^2 \leq Kx^2 (K \geq 36)
\end{aligned}$$

This proves the real induction step.

Conclusion: Jane's claim is closest to the truth.

# Question 13

**(i)** $\omega = \log_b a = 1$, $f(n) = \log^{10} n = O(n^{1-\epsilon})$. Therefore, CASE(-) is applicable, and $T(n) = \Theta(n)$

**(ii)** $\omega = \log_b a = \log_{10} 100 = 2 < 10 = k$. Thus, by corollary 11, $T(n) = \Theta(n^{10})$

**(iii)**
$(\log n)^{\log\log n} = 2^{(\log\log n)^2} = O(n^{\log_{100} 10 - \epsilon}) = O(n^{\frac{1}{2}-\epsilon})$. Therefore, CASE(-) is applicable, and $T(n) = \Theta(n^{1/2})$

**(iv)**
$T(n) = 16T(n/4) + n^2$, and $\omega = \log_b a = \log_2 4 = 2 = k$. Therefore, by corollary 11, $T(n) = \Theta(n^2 \lg n)$