Master Theorem

- The Master Theorem applies to:

- $T(n) = a\,T(n / b) + f(n)$
- Where $a >= 1$ and $b > 1$

Master Theorem

- $T(n) = a T(n / b) + f(n)$
- Where $a >= 1$ and $b > 1$

- Note 1: if $b < 1$, the problem would get bigger, not smaller
- Note 2: if $b = 1$, the size of the problem would stay the same

Master Theorem Intuition

- $T(n) = a\,T(n / b) + f(n)$
- Where $a >= 1$ and $b > 1$

- Let’s first assume that $f(n) = 0$
- Note: this is not very realistic (but will help understand what is going on with the running time better)

Master Theorem Intuition

- $T(n) = a\, T(n/b)$
- Let’s drill down and iterate

- $T(n) = a\,(a\, T(n/b^2))$
- $T(n) = a^2\, T(n/b^2)$

Master Theorem Intuition

- $T(n) = a\,T(n / b)$
- $T(n) = a^2\,T(n / b^2)$

- $T(n) = a^2\,(a\,T(n / b^3))$
- $T(n) = a^3\,T(n / b^3)$

Master Theorem Intuition

- $T(n) = a\,T(n/b)$
- $T(n) = a^2\,T(n/b^2)$
- $T(n) = a^3\,T(n/b^3)$

- $T(n) = a^3\,(a\,T(n/b^4))$
- $T(n) = a^4\,T(n/b^4)$
- Do we see a pattern?

Master Theorem Intuition

- $T(n) = a^{k} T(n / b^{k})$
- When do we reach the base case?

Master Theorem Intuition

- $T(n) = a^k T(n / b^k)$
- When do we reach the base case?

- When $n / b^k = 1$, i.e. $n = b^k$
- Then, $T(n) = a^k T(1)$

- What is the value of a^k?
- (considering that $n = b^k$)

Master Theorem Intuition

- $T(n) = a^k T(n / b^k)$
- What is the value of a^k?
- $n = b^k$

- Consider $c = \log_b a$

Master Theorem Intuition

- Consider $c = \log_b a$
- b to the power and log in base b are inverse functions of each other
- We take b to the power of each side, we get:
- $b^c = b^{\log_b a} = a$

Master Theorem Intuition

- $T(n) = a^{k} T(1)$
- $n = b^{k}$
- $b^{c} = a$
- $a^{k} = (b^{c})^{k} = b^{c\,k} = b^{k\,c} = (b^{k})^{c} = n^{c}$

- $T(n) = n^{c} T(1)$

Master Theorem Intuition

- $T(n) = n^{c}\, T(1)$
- Where $c = \log_b a$

- $T(n) = \Theta(n^{c})$ where $c = \log_b a$
- .. for $f(n) = 0$ .. which is not very realistic

Master Theorem Intuition

- $T(n) = \Theta(n^c)$ where $c = \log_b a$
- .. for $f(n) = 0$ .. which is not very realistic

- If $b = a$, then $T(n) = \Theta(n)$
- If $b < a$, then $c > 1$ (worse then $\Theta(n)$)
- If $b > a$, then $c < 1$ (better than $\Theta(n)$)

Master Theorem Intuition

- $T(n) = \Theta(n^c)$ where $c = \log_b a$
- .. for $f(n) = 0$ .. which is not very realistic

- If $b < a$, then $c > 1$ (worse than $\Theta(n)$)
- Example: $T(n) = 3\,T(n / 2)$
- Problem size is divided by 2 but we have 3 recursive calls

Master Theorem Intuition

- $T(n) = \Theta(n^c)$ where $c = \log_b a$
- .. for $f(n) = 0$ .. which is not very realistic

- If $b > a$, then $c < 1$ (better than $\Theta(n)$)
- Example: $T(n) = 2\,T(n / 3)$
- 2 recursive calls but problem size is divided by 3

Master Theorem Intuition

- $T(n) = \Theta(n^c)$ where $c = \log_b a$
- .. for $f(n) = 0$ .. which is not very realistic
- If $b > a$, then $c < 1$ (better than $\Theta(n)$)

- Another example: $T(n) = T(n / 2)$
- 1 recursive call but problem size is divided by 2 (Sounds similar to ??)

Master Theorem Intuition

- Now let's look at scenarios where f(n) is NOT 0

- If f(n) is smaller than $\Theta(n^c)$
- Then $\Theta(n^c)$ is going to be the dominant factor and T(n) is going to be $\Theta(n^c)$

Master Theorem Intuition

- If $f(n)$ is $\Theta(n^c)$ then
- A logarithmic factor will be introduced and
- $T(n)$ is going to be $\Theta(n^c \log n)$

Master Theorem Intuition

- If f(n) is (strictly, from a Ω standpoint) bigger than $\Theta(n^c)$ then
- f(n) is going to be the dominant term and
- T(n) is going to be $\Theta(f(n))$