



THẢO LUẬN ĐỒ ÁN MÔN HỌC:

# PHÂN TÍCH & THIẾT KẾ

# THUẬT TOÁN

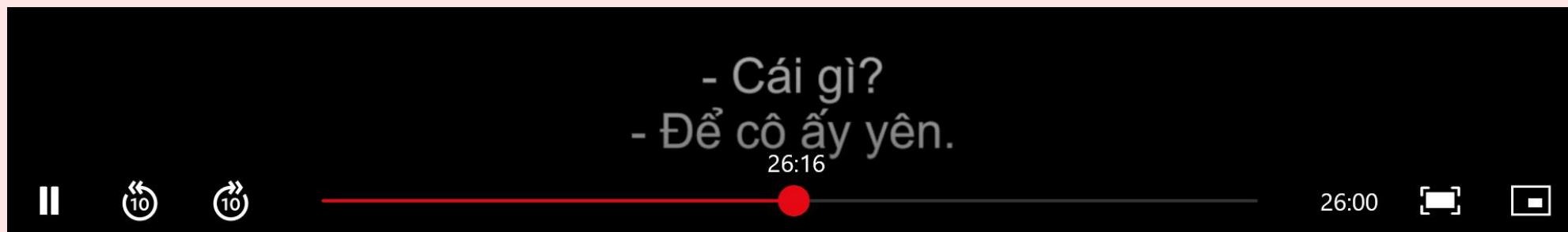
Hello ~

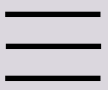


**NHÓM** | 19522065 – Nguyễn Thị Minh Phương  
**01** | 19522065 – Phan Nhật Minh



## ĐẶT VẤN ĐỀ



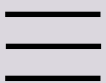


## CHỦ ĐỀ

# CHIA ĐỂ TRỊ!

## DIVIDE and CONQUER





- 1** Đặt vấn đề
- 2** Khái niệm
- 3** Các bước thực hiện
- 4** Phân tích độ phức tạp thuật toán
- 5** Các bài toán áp dụng
- 6** Ưu điểm & nhược điểm
- 7** Quizzes
- 8** Tổng kết & BTVN
- 9** Tài liệu tham khảo

# TABLE OF CONTENTS



## Khái niệm



# Chia để trị

Là phương pháp giải quyết một bài toán bằng cách *chia nhỏ* nó ra thành các bài toán con, ta sẽ tìm cách *giải quyết* các bài toán con này. Sau đó lời giải của các bài toán con được *tổng hợp* lại thành lời giải cho bài toán ban đầu.



01

Divide.

Chia bài toán ban đầu thành nhiều bài toán con nhỏ hơn.

02

Conquer.

Tìm phương án để giải quyết từng bài toán con một cách cụ thể.

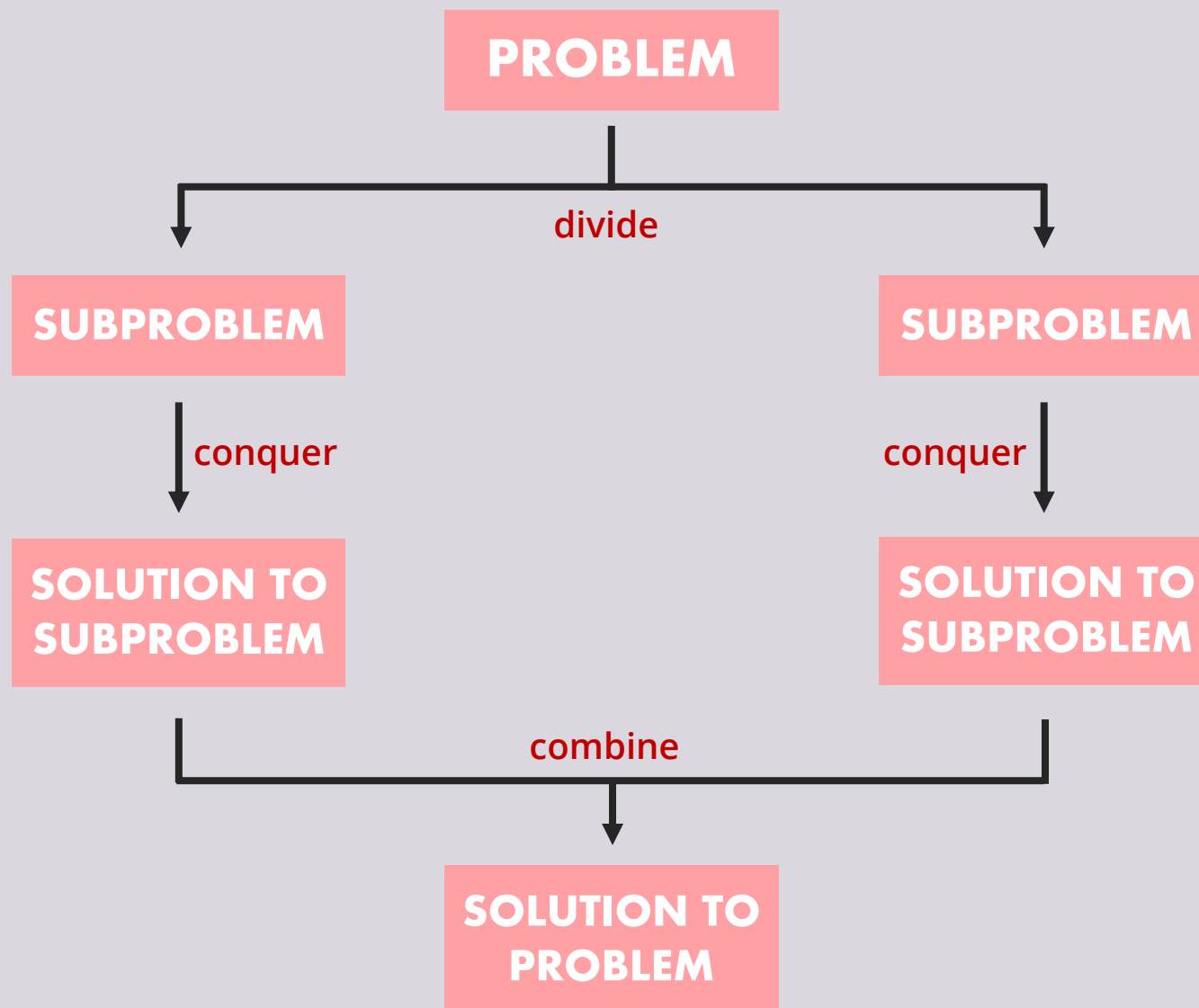
03

Combine.

Kết hợp lời giải của các bài toán con để suy ra kết quả của bài toán ban đầu cần tìm.

**Các bước  
thực hiện**





```
DAC(P) {  
    if(small(P)) {  
        Solution(P);  
    }  
    else {  
        divide P into P1, P2,..., Pk  
        apply DAC(P1), DAC(P2),..., DAC(Pk)  
        combine(DAC(P1), DAC(P2),..., DAC(Pk))  
    }  
}
```





# Recurrence Relation

$$T(n) = a * T(\frac{n}{b}) + f(n)$$

$f(n)$ : hàm tính thời gian chia bài toán kích thước  $n$  thành các bài toán có kích thước  $\frac{n}{b}$  và thời gian kết hợp các lời giải.



# Master Theorem

If  $f(n) \in \Theta(n^d)$  where  $d \geq 0$  in recurrence

$$T(n) \in \begin{cases} \Theta(n^d) & \text{if } a < b^d, \\ \Theta(n^d \log n) & \text{if } a = b^d, \\ \Theta(n^{\log_b a}) & \text{if } a > b^d. \end{cases}$$



# Master Theorem

If  $f(n) \in \Theta(n^d)$  where  $d \geq 0$  in recurrence

$$T(n) \in \begin{cases} \Theta(n^d) & \text{if } a < b^d, \\ \Theta(n^d \log n) & \text{if } a = b^d, \\ \Theta(n^{\log_b a}) & \text{if } a > b^d. \end{cases}$$

# Examples

$$T(n) \in \Theta(?)$$

$$T(n) = 4T\left(\frac{n}{2}\right) + n$$

$$d = 1, a = 4, b = 2 \Rightarrow a > b^d (4 > 2^1) \Rightarrow T(n) \in \Theta(n^2)$$

$$T(n) = 4T\left(\frac{n}{2}\right) + n^2$$

$$d = 2, a = 4, b = 2 \Rightarrow a = b^d (4 = 2^2) \Rightarrow T(n) \in \Theta(n^2 \log(n))$$

$$T(n) = 4T\left(\frac{n}{2}\right) + n^3$$

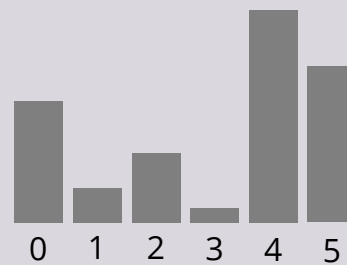
$$d = 3, a = 4, b = 2 \Rightarrow a < b^d (4 < 2^3) \Rightarrow T(n) \in \Theta(n^3)$$

# Bài tập áp dụng

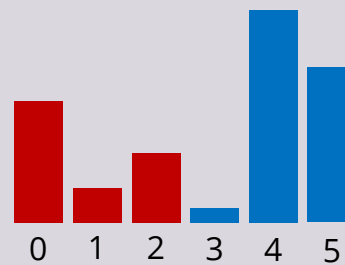




## MERGE SORT



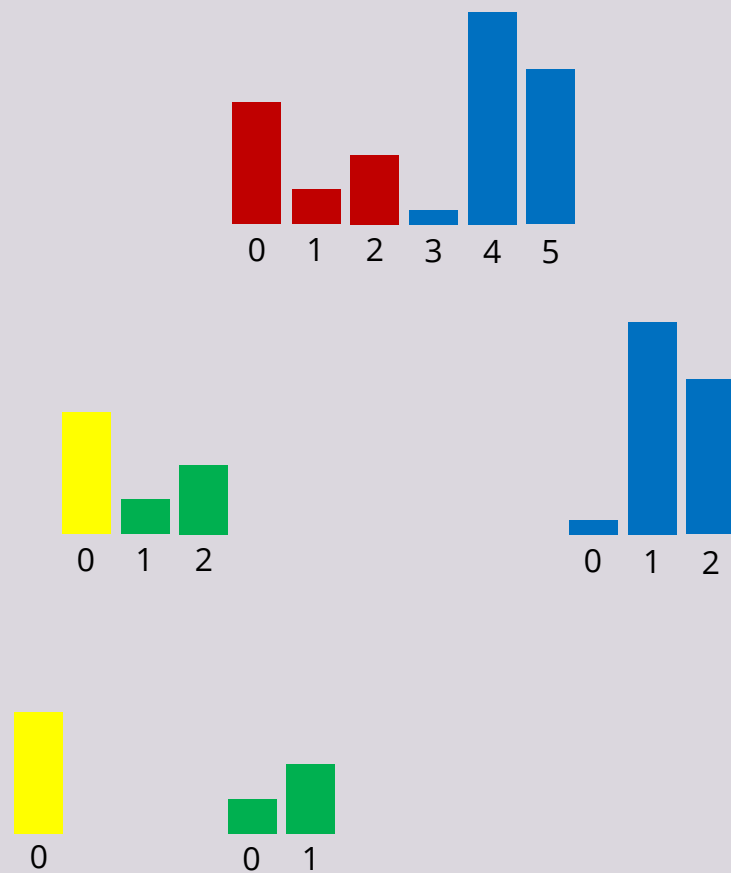
## MERGE SORT



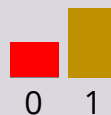
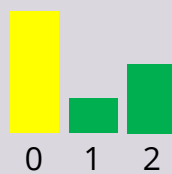




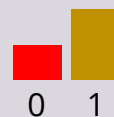
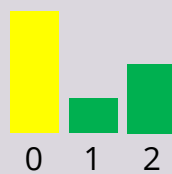




Number of Children	Frequency
0	3
1	1
2	2
3	0.5
4	4
5	3



Number of Children	Frequency
0	4
1	2
2	3
3	1
4	6
5	5















The figure displays three bar charts, each representing a different category. The top chart has six bars (0-5) with heights approximately 3.5, 1.5, 2.5, 0.5, 5.5, and 4.5. The bottom-left chart has three bars (0-2) with heights approximately 1.0, 1.5, and 2.0. The bottom-right chart has three bars (0-2) with heights approximately 0.5, 3.5, and 3.0. The bars are colored red, blue, and green respectively.

Category	Count
0	3.5
1	1.5
2	2.5
3	0.5
4	5.5
5	4.5

Category	Count
0	1.0
1	1.5
2	2.0

Category	Count
0	0.5
1	3.5
2	3.0









The figure displays a 3x3 grid of bar charts, each representing the distribution of a variable across three categories (0, 1, 2) for a specific group (red, blue, green). The distributions are as follows:

- Row 1:**
  - Red:** Category 0 has a value of 3, Category 1 has a value of 1, and Category 2 has a value of 2.
  - Blue:** Category 0 has a value of 0, Category 1 has a value of 0, and Category 2 has a value of 0.
  - Green:** Category 0 has a value of 0, Category 1 has a value of 4, and Category 2 has a value of 3.
- Row 2:**
  - Red:** Category 0 has a value of 1, Category 1 has a value of 2, and Category 2 has a value of 3.
  - Blue:** Category 0 has a value of 0, Category 1 has a value of 0, and Category 2 has a value of 0.
  - Green:** Category 0 has a value of 1, Category 1 has a value of 4, and Category 2 has a value of 3.
- Row 3:**
  - Red:** Category 0 has a value of 0, Category 1 has a value of 4, and Category 2 has a value of 3.
  - Blue:** Category 0 has a value of 0, Category 1 has a value of 0, and Category 2 has a value of 0.
  - Green:** Category 0 has a value of 0, Category 1 has a value of 0, and Category 2 has a value of 0.

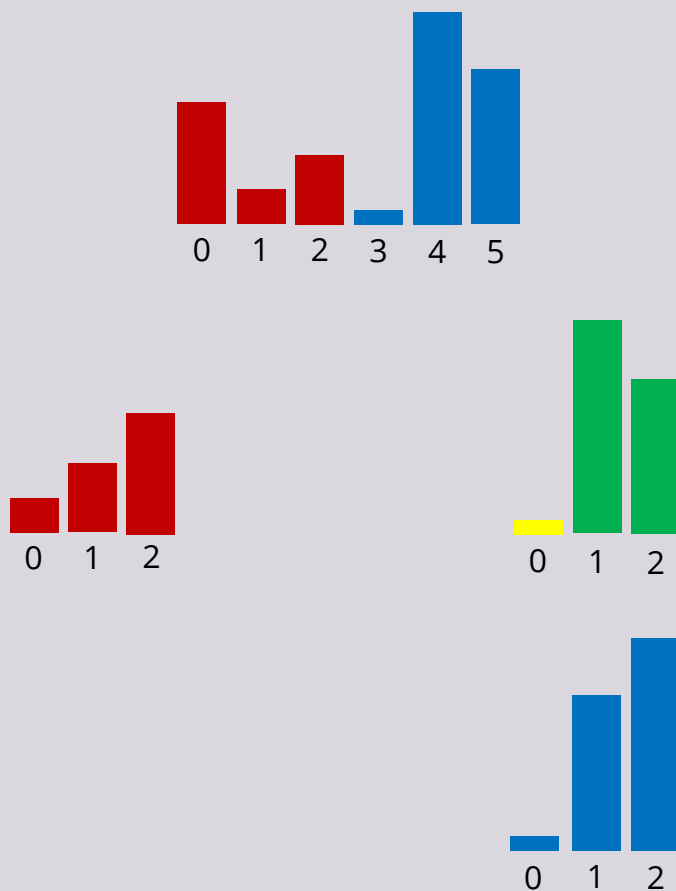


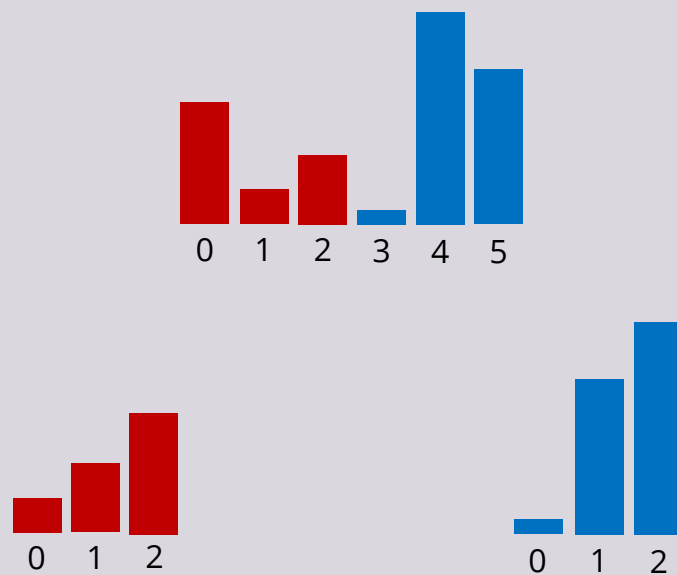


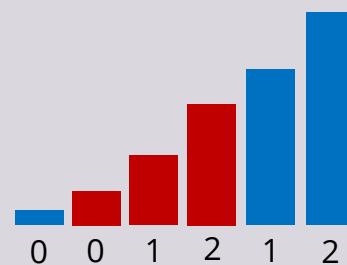




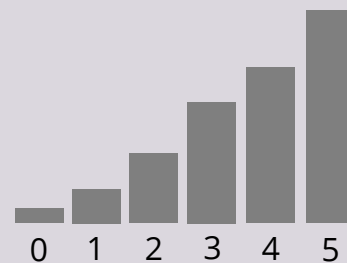
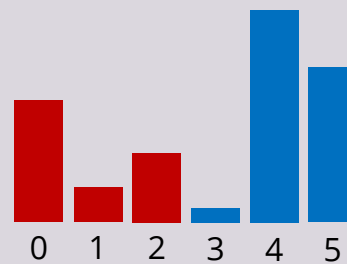




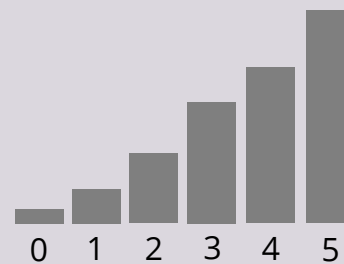




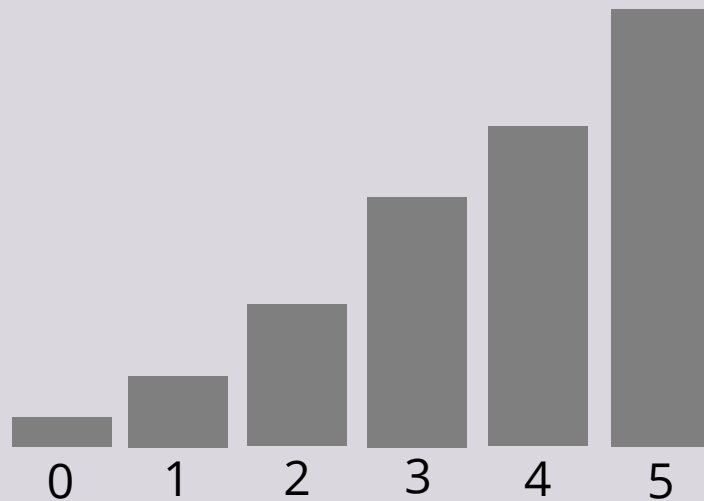
## MERGE SORT



## MERGE SORT



# MERGE SORT



$$T(n) \in \Theta(n \log(n))$$

## Code

[Code minh họa](#)







# Karatsuba's Algorithm

## Multiplication of Large Integers

### Recurrence Relation

$$A(n) = 3A(n/2) + cn \quad \text{for } n > 1, \quad A(1) = 1.$$

### Độ phức tạp

$$A(n) \in \Theta(n^{\log_2 3})$$

### Code

[Code minh họa](#)



# MỘT SỐ BÀI TOÁN KHÁC:

- Binary Search
- Quick Sort
- Binary Tree Traversals and Related Properties
- Strassen's Matrix Multiplication
- The Closest-Pair and Convex-Hull Problems by Divide-and-Conquer



## Ưu điểm & nhược điểm của kỹ thuật chia để trị

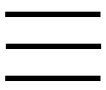
### Ưu điểm

- Giải quyết được bài toán lớn, phức tạp.
- Trực quan, dễ cài đặt.
- Độ phức tạp thường thấp hơn Brute Force.
- Tìm ra giải thuật hiệu quả.
- Sử dụng bộ nhớ đệm hiệu quả.
- Độ chính xác cao.
- Giải quyết các vấn đề con song song bằng đa luồng.

### Nhược điểm

- Độ quy thực thi chậm, dễ xảy ra lỗi khi cài đặt.
- Tốn thêm bộ nhớ.
- Khó khăn trong việc chọn điều kiện dừng.
- Trùng lặp vấn đề con khi chia nhỏ.
- Đôi khi sẽ phức tạp.





# Quizzes

- **Mỗi nhóm 1 đội**
- **Nickname: nhóm01, nhóm02,...**





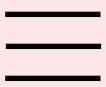
# TỔNG KẾT

## Bài tập

Tìm vị trí của phần tử lớn nhất trong mảng gồm  $n$  phần tử, sử dụng kỹ thuật chia để trị.

1. Viết chương trình thực hiện bài toán.
2. Giải recurrence relation của bài toán.
3. So sánh với thuật toán Brute Force.

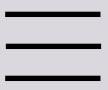




## Tài liệu tham khảo

- Anany Levition, Introduction to the Design and Analysis of Algorithms, 3<sup>rd</sup> edition, 2014
- [https://en.wikipedia.org/wiki/Divide-and-conquer\\_algorithm](https://en.wikipedia.org/wiki/Divide-and-conquer_algorithm)





# THANKS FOR LISTENING





# Trả lời các câu hỏi bổ sung sau buổi thảo luận

## 1. Bài toán nào có thể áp dụng kỹ thuật chia để trị?

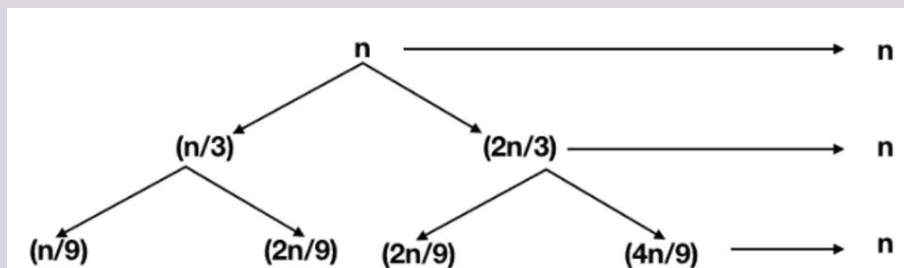
Những bài toán lớn có thể chia ra thành những bài toán con *độc lập, không gộp lên nhau*.

## 2. Làm sao để tính độ phức tạp khi chia bài toán ra thành các bài toán con không đều nhau?

Sử dụng phương pháp cây đệ quy.

Ví dụ:

$$T(n) = T\left(\frac{n}{3}\right) + T\left(\frac{2n}{3}\right) + n$$



## 3. Sự khác nhau giữa a và b trong công thức truy hồi tổng quát (Recurrence Relation)?

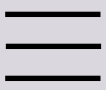
$$T(n) = a * T\left(\frac{n}{b}\right) + f(n)$$

b: tổng số vấn đề con sau khi chia nhỏ

a: số vấn đề con cần được *giải quyết*

Ví dụ:  $a^b = a^{\frac{b}{2}} \cdot a^{\frac{b}{2}}$

Ở đây  $b = 2$  và  $a = 1$  vì ta chỉ cần tính  $a^{\frac{b}{2}}$  một lần.



# Bảng phân công nhiệm vụ và đánh giá mức độ hoàn thành

Họ và tên	MSSV	Nhiệm vụ phân công	Mức độ hoàn thành
Nguyễn Thị Minh Phương	19522065	<ul style="list-style-type: none"><li>- Tìm tài liệu, tìm hiểu nội dung kiến thức.</li><li>- Lên dàn ý bài thuyết trình, làm slide.</li><li>- Soạn bộ câu hỏi Kahoot.</li><li>- Đại diện thuyết trình trong buổi thảo luận.</li></ul>	100%
Phan Nhật Minh	19520166	<ul style="list-style-type: none"><li>- Tìm tài liệu, tìm hiểu nội dung kiến thức.</li><li>- Tìm ví dụ minh họa, bài tập trên lớp, bài tập về nhà.</li><li>- Viết biên bản buổi thảo luận.</li><li>- Giải/sửa bài tập về nhà.</li></ul>	100%