###### **PHƯƠNG PHÁP CHIA ĐỂ TRỊ ( Divide And Conque Strategy)**

Thực hiện: Nguyễn Trung Hiếu.

Mã SV: 09000003.

Tài liệu thao khảo:

* [Bài giảng và bài tập trên lớp](http://www.mediafire.com/?ovpxtb2j2chlm)
* [Thiết kế và đánh giá thuật toán – Trần Tuấn Minh](http://www.mediafire.com/?t2dtflm1e9n)

# Lý thuyết về thuật toán

## Ý tưởng

Cho bài toán A, chia dữ liệu bài toán A thành các miền đủ nhỏ tạo nên các bài toán con, giải lần lượt các bài toán con trên các miền đã chia, việc giải các bài toán con này được thực hiện theo cùng một cách. Lời giải của bài toán lớn A là sự tổng hợp được xây dựng từ lời giải của các bài toán con.

## Mô hình và lược đồ

#### Mô hình:

Xét bài toán P trên miền dữ liệu R.

* Gọi D&C(R) là thuật giải P trên miền dữ liệu R.
* Nếu R có thể phân rã thành n miền con (R= R1∪R2∪…∪Rn )
* Với R0 là miền đủ nhỏ để D&C(R0) có lời giải.

#### Lược đồ:

|  |
| --- |
| **Void** D&C(R)  {  **if**(R=R0)  Giải D&C(R0);  {  **else**  Chia miền R thành R1,R2,…,Rn  **for** (i = 1; i <=m; i++)  D&C(Ri)  Tổng hợp để nhận lời giải.  }  } |

* Chia để trị sử dụng kĩ thuật đệ qui, thông thường là đệ qui nhiều nhánh.
* Tính đúng của thuật toán D&C có thể được chứng minh như đối với giải thuật đệ quy sử dụng quy nạp.

## Một số bài toán được sử dụng phương pháp chia để trị

* Bài toán tìm kiếm: thuật toán tìm kiếm nhị phân.
* Bài toán sắp xếp: thuật toán Quicksort, Mergesort.
* Bài toán MinMax.
* Bài toán nhân số lớn, nhân ma trận kích thước lớn.
* Tính chuỗi Fourier.
* Bài toán phân tích ngữ nghĩa (xử lí ngôn ngữ).

# Một số ví dụ về thuật toán chi để trị

### Bài toán tìm kiếm nhị phân

## Phát biểu bài toán

Cho mảng A gồm n phần tử đã được sắp xếp, và phần tử khóa x. Tìm trong mảng A xem có x hay không. Nếu có thì trả về giá trị true, không thì false.

## Phân tích bài toán

#### Ý tưởng:

Giả xử mảng A đã xắp xếp có giá trị tăng.

Khi đó với phần tử Ai bất kì, ta đều có:

Nếu x > Ai thì x nằm trong khoảng [Ai+1…..An]

Nếu x < Ai thì x nằm trong khoảng [A1…..Ai-1]

Nếu x = Ai thì kết thúc thuật toán.

#### Thuật toán:

|  |
| --- |
| **void** binarysearch(A[],key,left,right) {  **int** mid;  **if**(left==right){  **if**(key==A[left])  **println**(true);  }  **else**{  mid=(left+right)/2;  **if**(key==A[mid])  **println**(true);  **else**{  **if**(key<A[mid])  binarysearch(A,key,left,mid-1);  **else**  binarysearch(A,key,mid+1,right);  }  }  } |

#### Độ phức tạp:

|  |  |  |
| --- | --- | --- |
| Trường hợp | Số lần so sánh | Khi |
| Tốt nhất | 1 | Key=mid ở lần đâu tiên |
| Xấu nhất | Log2n | Key nằm ở đầu, hoặc cuối mảng, hoặc không có trong mảng |
| Trung bình | Log2(n/2) | Xác xuất key xuất hiện tại các vị trí trong A là như nhau |

* Độ phức tạp thuật toán là O(log2(n/2))

##### Ví dụ minh họa

Cho dãy A đã được sắp xếp:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | 9 | 12 | 34 | 56 | 76 | 84 | 89 |

Tìm xem phần tử 56 có ở trong mảng A hay không?

Đầu tiên khởi tạo giá trị left=0; right=8. Vậy mid=4.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | 9 | 12 | 34 | 56 | 76 | 84 | 89 |

mid[4]=34 <56. Vậy left=5; right=8. Vậy mid=6.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | 9 | 12 | 34 | 56 | 76 | 84 | 89 |

mid[6]=76>56/. Vậy left=5; right=5. Vậy mid=5.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | 9 | 12 | 34 | 56 | 76 | 84 | 89 |

mid[5]=56. Vậy dừng lại do đã tìm thấy phần tử cần tìm kiếm.

Trong trường hợp mid[5]≠56 thì thuật toán cũng sẽ dừng lại do left=right và kết luận rằng trong mảng A không có phần tử 56.

### Bài toán MinMax:

## Phát biểu bài toán

Cho mảng A chưa sắp xếp.Tìm Min và Max.

## Phân tích bài toán

#### Ý tưởng

Tại mỗi bước chia đôi đoạn cần tìm, rồi tìm Min Max của từng đoạn. Sau đó tổng hợp lại kết quả.

Nếu đoạn chia chỉ có 1 phần tử thì Min = Max = phần tử đó.

#### Thuật toán

Với bước đầu khởi tạo, ta có left=0, right=A.length-1

|  |
| --- |
| **void** MaxMin(A[],left,right) {  **int** max1, min1, mid, max2, min2;  **if**(right-left<=1){  **if**(A[left]<A[right]){  max = A[right];  min = A[left];  }  **else** {  max = A[left];  min = A[right];  }  }  **else**{  mid=(left+right)/2;  MaxMin(A,left,mid);  max1=max;  min1=min;  MaxMin(A,mid+1,right);  max2=max;  min2=min;  **if** (max2 < max1) max = max1;  **if** (min2 > min1) min = min1;  }    } |

#### Độ phức tạp:

Gọi T(n) là số phép toán cần thực hiênk. Khi đó ta có:

Với , thì:

Vậy

##### Ví dụ minh họa:

Cho dãy A gồm các phần tử:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 9 | 6 | 84 | 21 | 46 | 7 | 16 | 3 | 31 |

Bước đầu khởi tạo, ta có left =0; right=9;

Do right-left>1=> mid=4;

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 9 | 6 | 84 | 21 | 46 | 7 | 16 | 3 | 31 |
| MaxMin(A,left,mid)  MaxMin(A,0,4) | | | | | MaxMin(A,mid+1,right)  MaxMin(A,5,9) | | | | |
| 1 | 9 | 6 | 84 | 21 | 46 | 7 | 16 | 3 | 31 |
| MaxMin(A,0,2) | | | MaxMin(A,3,4) | | MaxMin(A,5,7) | | | MaxMin(A,8,9) | |
| 1 | 9 | 6 | 84 | 21 | 46 | 7 | 16 | 3 | 31 |
| MaxMin(A,0,1) | |  |  |  | MaxMin(A,5,6) | |  |  |  |
| 1 | 9 | 6 | 84 | 21 | 46 | 7 | 16 | 3 | 31 |
| Bài toán có sử dụng đệ qui nên đảo ngược để tìm kết quả | | | | | | | | | |
| 1 | 9 | 6 | 84 | 21 | 46 | 7 | 16 | 3 | 31 |
| 1 | 9 | 6 | 84 | 21 | 46 | 7 | 16 | 3 | 31 |
| 1 | 9 | 6 | 84 | 21 | 46 | 7 | 16 | 3 | 31 |
| ***1*** | 9 | 9 | ***84*** | 21 | 46 | 7 | 7 | 3 | 31 |

Vậy Max=84; Min=1;

Chú thích:

|  |  |
| --- | --- |
| Màu | Ý nghĩa |
|  | Chỉ giá trị mid trong các lần |
|  | Chỉ các giá trị MaxMin đã tìm ra, kết thúc |
|  | Chỉ các giá trị Min trong các lần so sánh |
|  | Chỉ các giá trị Max trong các lần so sánh |

### Bài toán sắp xếp: MergeSort

## Phát biểu bài toán

Cho mảng A chưa sắp xếp. Sắp xếp tăng dần các phần tử trong mảng A bằng phương pháp trộn hai đường trực tiếp.

## Phân tích bài toán

#### Ý tưởng

Bài toán được lặp đi lặp lại nhiều lần. Trong mỗi bước lặp đều thực hiện 2 giai đoạn.

Giai đoạn 1: Phân bố

Phân bố p phần tử trong mảng A vào 2 mảng đệm A1,A2 bằng phương pháp luân phiên.

Giai đoạn 2: Trộn có sắp xếp

Trộn p phần tử ở dãy A1 với p phần tử ở dãy A2. Kết quả trộn được đưa vào A, trong khi chưa hết dãy A1,A2.

Thuật toán sẽ thực hiện đến khi p>F.

Sau mỗi bước lặp thì p=p\*2.

VD:

Khi ở bước lặp đầu, p=1:

A[k]A1 khi k=0,2,4,…,2n

A[k]A2 khi k=1,3,5,…,2n+1

Khi ở bước lặp thứ 2, p=2:

A[k]A1 khi k=0,1,3,4,7,8…,4n,4n+1

A[k]A2 khi k=2,3,5,6,8,9…,4n+2,4n+3

Khi ở bước lặp thứ 3, p=4:

A[k]A1 khi k=0,1,2,3,8,9,10,11,…,8n,8n+1,8n+2,8n+3

A[k]A2 khi k=4,5,6,7,12,13,14,15,…,8n+4,8n+5,8n+6,8n+7

Tương tự ở các bước lặp sau.

##### Ví dụ minh họa

Cho dãy A gồm các phần tử:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 6 | 3 | 42 | 59 | 7 | 9 | 11 | 21 | 12 | 17 | 32 | 45 | 26 | 52 | 84 | 64 | 29 |

Sắp xếp dãy trên theo phương pháp MergeSort

## Bước lặp đầu tiên:

Giai đoạn 1. Phân bố

Phân bố p phần tử mảng A vào 2 mảng A1, A2. p=1.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A1 | 1 | 3 | 59 | 9 | 21 | 17 | 45 | 52 | 64 |
| A2 | 6 | 42 | 7 | 11 | 12 | 32 | 26 | 84 | 29 |

Giai đoạn 2. Trộn có sắp xếp

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | 1 | 6 | 3 | 42 | 7 | 59 | 9 | 11 | 12 | 21 | 17 | 32 | 26 | 45 | 52 | 84 | 29 | 64 |

## Bước lặp thứ 2:

Giai đoạn 1. Phân bố

Phân bố p phần tử mảng A vào 2 mảng A1, A2. p=p\*2=2.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A1 | |  |  | | --- | --- | | 1 | 6 | | |  |  | | --- | --- | | 7 | 59 | | |  |  | | --- | --- | | 12 | 21 | | |  |  | | --- | --- | | 26 | 45 | | |  |  | | --- | --- | | 29 | 64 | |
| A2 | |  |  | | --- | --- | | 3 | 42 | | |  |  | | --- | --- | | 9 | 11 | | |  |  | | --- | --- | | 17 | 32 | | |  |  | | --- | --- | | 52 | 84 | |  |

Giai đoạn 2. Trộn có sắp xếp

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | 1 | 3 | 6 | 42 | 7 | 9 | 11 | 59 | 12 | 17 | 21 | 32 | 26 | 45 | 52 | 84 | 29 | 64 |

## Bước lặp thứ 3:

Giai đoạn 1. Phân bố

Phân bố p phần tử mảng A vào 2 mảng A1, A2. p=p\*2=4.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A1 | |  |  |  |  | | --- | --- | --- | --- | | 1 | 3 | 6 | 42 | | |  |  |  |  | | --- | --- | --- | --- | | 12 | 17 | 21 | 32 | | |  |  | | --- | --- | | 29 | 64 | |
| A2 | |  |  |  |  | | --- | --- | --- | --- | | 7 | 9 | 11 | 59 | | |  |  |  |  | | --- | --- | --- | --- | | 26 | 45 | 52 | 84 | |  |

Giai đoạn 2. Trộn có sắp xếp

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | 1 | 3 | 6 | 7 | 9 | 11 | 42 | 59 | 12 | 17 | 21 | 26 | 32 | 45 | 52 | 84 | 29 | 64 |

## Bước lặp thứ 4:

Giai đoạn 1. Phân bố

Phân bố p phần tử mảng A vào 2 mảng A1, A2. p=p\*2=8.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A1 | |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | 1 | 3 | 6 | 7 | 9 | 11 | 42 | 59 | | |  |  | | --- | --- | | 29 | 64 | |
| A2 | |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | 12 | 17 | 21 | 26 | 32 | 45 | 52 | 84 | |  |

Giai đoạn 2. Trộn có sắp xếp

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | 1 | 3 | 6 | 7 | 9 | 11 | 12 | 17 | 21 | 26 | 32 | 42 | 45 | 52 | 59 | 84 | 29 | 64 |

## Bước lặp thứ 5:

Giai đoạn 1. Phân bố

Phân bố p phần tử mảng A vào 2 mảng A1, A2. p=p\*2=16.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A1 | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 1 | 3 | 6 | 7 | 9 | 11 | 12 | 17 | 21 | 26 | 32 | 42 | 45 | 52 | 59 | 84 | |
| A2 | |  |  | | --- | --- | | 29 | 64 | |

Giai đoạn 2. Trộn có sắp xếp

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | 1 | 3 | 6 | 7 | 9 | 11 | 12 | 17 | 21 | 26 | 29 | 32 | 42 | 45 | 52 | 59 | 64 | 84 |

## Bước lặp thứ 6:

Giai đoạn 1. Phân bố

Phân bố p phần tử mảng A vào 2 mảng A1, A2. p=p\*2=32.

Do p=32>18 số phần tử của mảng A nên kết quả giai đoạn 2 ở bước lặp thứ 5 là kết quả của bài toán. Dãy đã được sắp xếp.