HUST

ĐẠI HỌC BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

ONE LOVE. ONE FUTURE.

ĐẠI HỌC
BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY
OF SCIENCE AND TECHNOLOGY

CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN

Sơ đồ thuật toán quy hoạch động

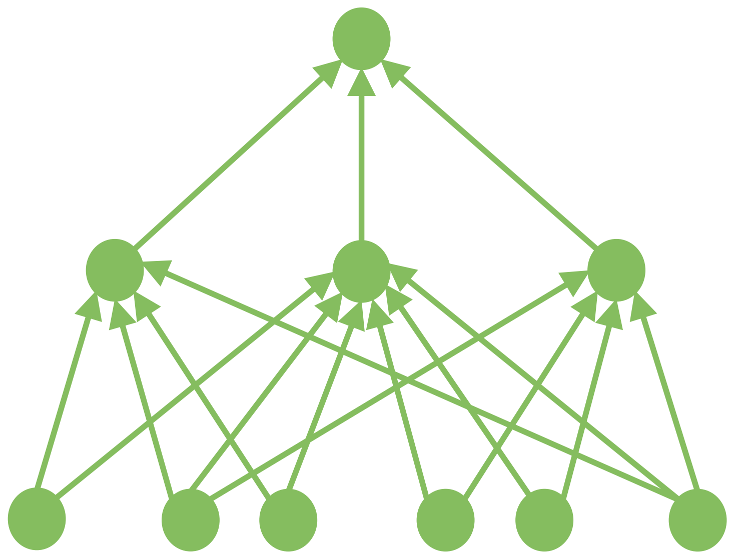
ONE LOVE. ONE FUTURE.

Giới thiệu

- Thuật toán quy hoạch động (Dynamic programming algorithm) được phát minh bởi Bellman trong thế chiến thứ 2. Tên đầu tiên của thuật toán này là multi-stage decision process (ra quyết định qua nhiều giai đoạn)

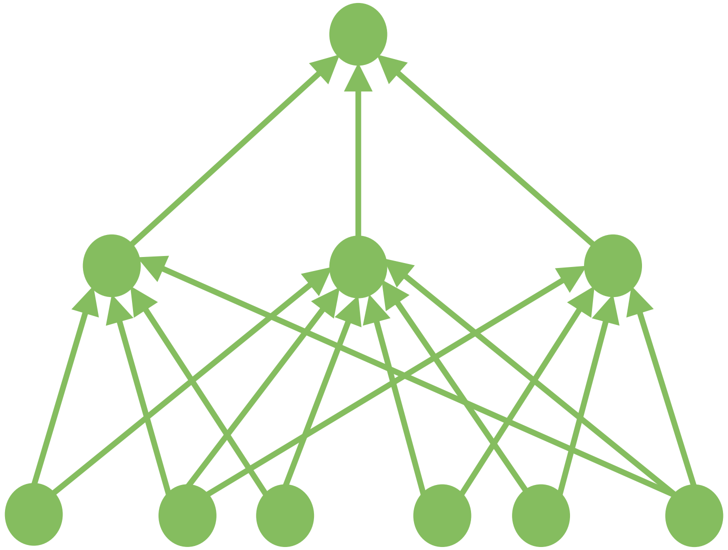
- Thuật toán quy hoạch động là một kỹ thuật mạnh để giải các bài toán tối ưu bằng cách chia chúng thành các bài toán nhỏ hơn và giải các bài toán con một lần duy nhất

- Thuật toán quy hoạch động có nhiều điểm giống với Thuật toán quay lui (backtracking) và Thuật toán chia để trị (divide conquer)



Sơ đồ thuật toán

- **CHIA** bài toán xuất phát thành các bài toán con không nhất thiết độc lập với nhau
- **GIẢI** các bài toán con từ nhỏ đến lớn, lời giải được lưu trữ lại vào bộ nhớ (để đảm bảo mỗi bài toán chỉ giải đúng 1 lần)
  - Bài toán con nhỏ nhất phải được giải một cách trực tiếp, đơn giản
- **KẾT HỢP** lời giải của bài toán lớn hơn từ lời giải đã có của các bài toán con nhỏ hơn (cần sử dụng cộng thức truy hồi)
  - Số lượng bài toán con cần được bị chặn bởi một hàm đa thức của kích thước dữ liệu đầu vào



Sơ đồ thuật toán

```
map<problem, value> Memory;

value DP(problem P) {
    if (is_base_case(P))
        return base_case_value(P);

    if (Memory.find(P) != Memory.end())
        return Memory[P];

    value result = some value;
    for (problem Q in subproblems(P))
        result = Combine(result, DP(Q));

    Memory[P] = result;
    return result;
}
```

Bộ nhớ, ánh xạ bài toán và lời giải

Bài toán con nhỏ nhất, tương ứng với bước cơ sở của thuật toán đệ quy

Luôn kiểm tra bộ nhớ xem bài toán con đã giải chưa, đã giải rồi không giải lại, nếu chưa giải thì thực hiện giải.

Giải bài toán con

Giải tất cả bài toán con và Kết hợp những lời giải tìm được để hình thành lời giải bài toán lớn hơn

Lưu kết quả bài toán con vào trong bộ nhớ

Sơ đồ thuật toán

```
map<problem, value> Memory;

value DP(problem P) {
    if (is_base_case(P))
        return base_case_value(P);

    if (Memory.find(P) != Memory.end())
        return Memory[P];

    value result = some value;
    for (problem Q in subproblems(P))
        result = Combine(result, DP(Q));

    Memory[P] = result;
    return result;
}
```

Sơ đồ thuật toán

```
map<problem, value> Memory;

value DP(problem P) {
    if (is_base_case(P))
        return base_case_value(P);

    if (Memory.find(P) != Memory.end())
        return Memory[P];

    value result = some value;
    for (problem Q in subproblems(P))
        result = Combine(result, DP(Q));

    Memory[P] = result;
    return result;
}
```

Bộ nhớ, ánh xạ bài toán và lời giải

Bài toán con nhỏ nhất, tương ứng với bước cơ sở của thuật toán đệ quy

Luôn kiểm tra bộ nhớ xem bài toán con đã giải chưa, đã giải rồi không giải lại, nếu chưa giải thì thực hiện giải.

Giải một bài toán con

Giải tất cả bài toán con và Kết hợp những lời giải tìm được để hình thành lời giải bài toán lớn hơn

Lưu kết quả bài toán con vào trong bộ nhớ

Đặc điểm

- So với thuật toán chia để trị, thuật toán quy hoạch động cũng có 3 bước là Chia, Giải bài toán con và Kết hợp. Tuy nhiên, trong chia để trị, các bài toán con là độc lập; còn trong quy hoạch động, các bài toán con gối nhau hay chồng chéo lên nhau.
- Khó khăn lớn nhất trong đề xuất thuật toán quy hoạch động là Công thức truy hồi (tên khác là Công thức quy hoạch động, Công thức đệ quy)
- Có 2 cách tiếp cận: Top-Down và Bottom-Up, trong đó Top-Down tự nhiên và dễ hiểu, dễ cài đặt
- Thiết kế bộ nhớ ảnh hưởng lớn tới tốc độ của thuật toán
- Bộ nhớ còn dung để truy vết, tìm ra tường minh lời giải tối ưu

Ví dụ: Đoạn con có tổng lớn nhất

- **Phát biểu**: Cho dãy n số nguyên $(a_1, a_2, \dots, a_n)$, hãy tìm đoạn con bao gồm các phần tử liên tiếp của dãy sao cho tổng các phần tử được chọn là cực đại.
- **Ví dụ**: Cho dãy 7 số nguyên sau

-16	**7**	**-3**	**0**	**-1**	**5**	**-4**

Đoạn con có tổng -12

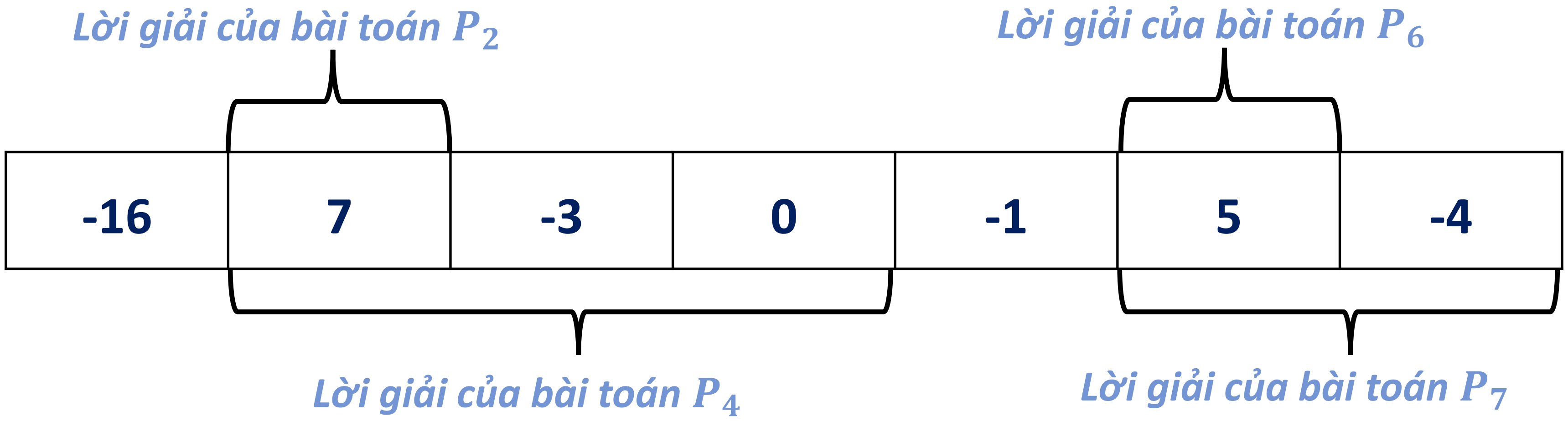
Đoạn con có tổng 0

  - *Lời giải tối ưu (dãy con có tổng cực đại bằng 8): 7, -3, 0, -1, 5*

- Thuật toán chia để trị có độ phức tạp $\mathrm{O}(\mathrm{n}\log(\mathrm{n}))$, chúng ta có thể làm tốt hơn với thuật toán quy hoạch động hay không?

Ví dụ: Đoạn con có tổng lớn nhất

- **Phát biểu**: Cho dãy n số nguyên $(a_1, a_2, \ldots, a_n)$, hãy tìm tổng của các phần tử của đoạn con bao gồm các phần tử liên tiếp của dãy sao cho tổng các phần tử được chọn là cực đại.
- **Xác định bài toán con:** P_i là bài toán tìm đoạn con bao gồm các phần tử liên tiếp có tổng lớn nhất mà phần tử cuối cùng là a_i, với mọi $i = 1, \ldots, n$.



Ví dụ: Đoạn con có tổng lớn nhất

- **Phát biểu**: Cho dãy n số nguyên $(a_1, a_2, \dots, a_n)$, hãy tìm tổng các phần tử của đoạn con bao gồm các phần tử liên tiếp của dãy sao cho tổng các phần tử được chọn là cực đại.

- **Xác định bài toán con**: P_i là bài toán tìm đoạn con bao gồm các phần tử liên tiếp có tổng lớn nhất mà phần tử cuối cùng là a_i, với mọi $i = 1, \dots, n$.

- **Công thức quy hoạch động** (công thức kết hợp lời giải các bài toán con để thu được lời giải bài toán cha): Gọi S_i là tổng các phần tử của lời giải của $P_i, \forall\, i = 1, \dots, n$.

Ta có: $S_1 = a_1,$

$$S_i = \begin{cases} s_{i-1} + a_i \text{ nếu } s_{i-1} > 0 \\ a_i \text{ nếu } s_{i-1} \leq 0 \end{cases}$$

- **Ví dụ minh họa:**

-16	7	-3	0	-1	5	-4

$$S_1 = -16, S_2 = a_2 = 7, S_3 = S_2 + a_3 = 4, S_4 = S_3 + 0 = 4,$$

$$S_5 = S_4 + (-1) = 3, S_6 = S_5 + 5 = 8, S_7 = S_6 + (-4) = 4$$

Ví dụ: Đoạn con có tổng lớn nhất

- **Công thức quy hoạch động** (công thức kết hợp lời giải các bài toán con để thu được lời giải bài toán cha): Gọi S_i là tổng các phần tử của lời giải của $P_i, \forall\, i = 1, \dots, n$. Ta có: $S_1 = a_1$,

$$S_i = \begin{cases} s_{i-1} + a_i \;\; nếu\; s_{i-1} > 0 \\ a_i \;\; nếu\; s_{i-1} \le 0 \end{cases}$$

- **Lời giải:** Tổng của các phần tử của đoạn con bao gồm các phần tử liên tiếp của dãy có tổng các phần tử được chọn lớn nhất là:

$$\boldsymbol{max(S_1, S_2, \dots, S_n)}$$

Ví dụ: Đoạn con có tổng lớn nhất

- **Lời giải:** Tổng của các phần tử của đoạn con bao gồm các phần tử liên tiếp của dãy có tổng các phần tử được chọn lớn nhất là:

$$\boldsymbol{max(S_1, S_2, \dots, S_n)}$$

Ví dụ minh họa:

-16	7	-3	0	-1	5	-4

$$\boldsymbol{S_1 = -16, S_2 = a_2 = 7, S_3 = S_2 + a_3 = 4, S_4 = S_3 + 0 = 4,}$$
$$\boldsymbol{S_5 = S_4 + (-1) = 3, S_6 = S_5 + 5 = 8, S_7 = S_6 + (-4) = 4}$$

Lời giải: $\boldsymbol{max(S_1, S_2, \dots, S_7) = 8}$

Ví dụ: Đoạn con có tổng lớn nhất

- **Công thức quy hoạch động** (công thức kết hợp lời giải các bài toán con để thu được lời giải bài toán cha): Gọi S_i là tổng các phần tử của lời giải của $P_i, \forall\, i = 1, \dots, n$. Ta có: $S_1 = a_1$,

$$S_i = \begin{cases} s_{i-1} + a_i \text{ nếu } s_{i-1} > 0 \\ a_i \text{ nếu } s_{i-1} \le 0 \end{cases}$$

- **Lời giải:** Tổng của các phần tử của đoạn con bao gồm các phần tử liên tiếp của dãy có tổng các phần tử được chọn lớn nhất là:

$$\boldsymbol{max(S_1, S_2, \dots, S_n)}$$

- **Độ phức tạp thuật toán: $\boldsymbol{O(n)}$**

Bài tập: Dãy con tăng chặt dài nhất

- **Đề bài**: Cho dãy số nguyên $A = (a_1, a_2, \dots, a_n)$ thỏa mãn điều kiện các phần tử đôi một khác nhau ($a_i \neq a_j, \forall i \neq j$). Một dãy con của A là một dãy thu được bằng cách xóa đi một số phần tử trong A. Một dãy con $B = (b_1, \dots, b_k)$ được gọi là tăng chặt khi $b_i < b_{i+1}, \forall i \in \{1, \dots, k-1\}$. Tìm độ dài của dãy con tăng chặt của A có độ dài lớn nhất.

- **Hình thức**: Làm tại nhà và nộp lại trên hệ thống chấm code

Ví dụ: Dãy con chung dài nhất

- **Phát biểu**: Cho 2 dãy ký tự $\mathrm{X} = (x_1, x_2, \dots, x_n)$ và $Y = (y_1, \dots, y_m\}$. Một dãy con của A là một dãy thu được bằng cách xóa đi một số phần tử trong A. Hãy tìm độ dài của dãy con chung dài nhất của 2 X và Y.

- **Ví dụ**:
  - $X = "abcb“$ và $Y = "bdcab“$
  - Dãy con chung dài nhất của 2 dãy là dãy $"bcb“$ với độ dài 3

- Nhận xét: Thuật toán vét cạn, so sánh tất cả các dãy con của X và Y sẽ có độ phức tạp $\mathrm{O}(2^{\mathrm{n}} \times 2^{\mathrm{m}} \times \max(\mathrm{m}, \mathrm{n}))$. Chúng ta có thể giải bài toán này nhanh hơn với một thuật toán quy hoạch động hay không?

Ví dụ: Dãy con chung dài nhất

- **Phát biểu:** Cho 2 dãy ký tự $X = (x_1, x_2, \dots, x_n)$ và $Y = (y_1, \dots, y_m\}$. Một dãy con của A là một dãy thu được bằng cách xóa đi một số phần tử trong A. Hãy tìm độ dài của dãy con chung dài nhất của 2 X và Y.

- **Xác định bài toán con:** Gọi $\mathrm{S(i,j)}$ là độ dài của dãy con chung dài nhất của 2 dãy, dãy con của X là $\mathrm{X_i = (x_1, \dots, x_i)}$ với $\mathrm{i} \in \{1, \dots, \mathrm{n}\}$ và dãy con của Y là $\mathrm{Y_j} = \left(\mathrm{y_1, \dots, y_j}\right)$ với $\mathrm{j} \in \{1, \dots, \mathrm{m}\}$.

- **Bài toán cơ sở (bài toán con nhỏ nhất):**

$$S(i,0) = 0, \forall i \in \{1, \dots, n\}$$
$$S(0,j) = 0, \forall j \in \{1, \dots, m\}$$

Ví dụ: Dãy con chung dài nhất

- **Xác định bài toán con**: Gọi $S(i,j)$ là độ dài của dãy con chung dài nhất của 2 dãy, dãy con của X là $X_i = (x_1, \dots, x_i)$ với $i \in \{1, \dots, n\}$ và dãy con của Y là $Y_j = (y_1, \dots, y_j)$ với $j \in \{1, \dots, m\}$.

- **Bài toán cơ sở** (bài toán con nhỏ nhất):

$$S(i,0) = 0, \forall i \in \{1, \dots, n\}$$

$$S(0,j) = 0, \forall j \in \{1, \dots, m\}$$

- **Công thức quy hoạch động:**

$$S(i,j) = max \begin{cases} S(i-1, j-1) \text{ nếu } \ x_i = y_j \\ S(i-1, j) \\ S(i, j-1) \end{cases}$$

Ví dụ: Dãy con chung dài nhất

- Công thức quy hoạch động

$$S(i,j) = max \begin{cases} S(i-1,j-1)\ nếu\ \ x_i = y_j \\ S(i-1,j) \\ S(i,j-1) \end{cases}$$

X							
	3	7	2	5	1	4	9

Y										
	4	3	2	3	6	1	5	4	9	7

	1	2	3	4	5	6	7	8	9	10
1	0	1	1	1	1	1	1	1	1	1
2	0	1	1	1	1	1	1	1	1	2
3	0	1	2	2	2	2	2	2	2	2
4	0	1	2	2	2	2	3	3	3	3
5	0	1	2	2	2	3	3	3	3	3
6	1	1	2	2	2	3	3	4	4	4
7	1	1	2	2	2	3	3	4	5	5

Ví dụ: Dãy con chung dài nhất

- **Xác định bài toán con**: Gọi $S(i, j)$ là độ dài của dãy con chung dài nhất của 2 dãy, dãy con của X là $X_i = (x_1, \dots, x_i)$ với $i \in \{1, \dots, n\}$ và dãy con của Y là $Y_j = (y_1, \dots, y_j)$ với $j \in \{1, \dots, m\}$.

- **Bài toán cơ sở** (bài toán con nhỏ nhất):

$$S(i, 0) = 0, \forall i \in \{1, \dots, n\}$$

$$S(0, j) = 0, \forall j \in \{1, \dots, m\}$$

- **Công thức quy hoạch động:**

$$S(i, j) = max \begin{cases} S(i-1, j-1) \text{ nếu } x_i = y_j \\ S(i-1, j) \\ S(i, j-1) \end{cases}$$

- **Độ phức thuật toán:** $O(n \times m)$

Bài tập: Dãy con tăng chặt dài nhất

- **Đề bài**: Cho dãy $A = (a_1, a_2, \ldots, an)$. Một dãy con của dãy A là một dãy thu được bằng cách loại bỏ một số phần tử khỏi A. Tìm độ dài của dãy con của A là một cấp số cộng với bước nhảy bằng 1 và có độ dài lớn nhất.

- **Hình thức**: Làm tại nhà và nộp lại trên hệ thống chấm code

HUST

hust.edu.vn fb.com/dhbkhn

THANK YOU !