**TRƯỜNG ĐẠI HỌC SÀI GÒN**

**KHOA CÔNG NGHỆ THÔNG TIN**

**🙣🙠🙢🙡**



Cơ Sở Trí Tuệ Nhân Tạo

Thành viên:

Bùi Anh Khoa – 3116410060

Trần Quang Danh – 3116410010

Nguyễn Duy Tân – 3117410226

**………, tháng………năm………**

Mục lục

[1 Bài 1 1](#_Toc58598481)

[1.1 Câu a 1](#_Toc58598482)

[1.2 Câu b 2](#_Toc58598483)

[1.2.1 Giải thích thuật toán 2](#_Toc58598484)

[1.2.2 Ví dụ 2](#_Toc58598485)

[1.2.2.1 Bước 1 xác định ma trận chi phí cơ hội 3](#_Toc58598486)

[1.2.2.2 Bước 2 kiểm tra điều kiện tối ưu 3](#_Toc58598487)

[1.2.2.3 Bước 3 thực hiện phân công tối ưu 4](#_Toc58598488)

[2 Bài 2 5](#_Toc58598489)

[2.1 Câu a 5](#_Toc58598490)

[2.2 Câu b 6](#_Toc58598491)

[3 Bài 3 6](#_Toc58598492)

# Bài 1

## Câu a

a. Giải thuật Heuristic

Ký hiệu:

M là số máy

N là số công việc

TJ là thời gian máy thực hiện công việc thứ j

- Chọn việc J chưa phân công có thời gian thực hiện cao nhất phân công cho máy có thời gian làm việc thấp nhất.

Lặp lại các bước sau cho tới khi hoàn thành tất cả công việc:

* Chọn việc J chưa phân công có thời gian cao nhất
* Chọn máy M có thời gian làm việc thấp nhất
* Bố trí việc J cho máy M

Ví dụ:

N=7, M=3

TJ= 3 4 1 5 2 3 6

Sắp xếp thời gian làm việc của các máy theo thứ tự giảm dần

TJ=6 5 4 3 3 2 1

Chọn công việc cao nhất cho máy thấp nhất

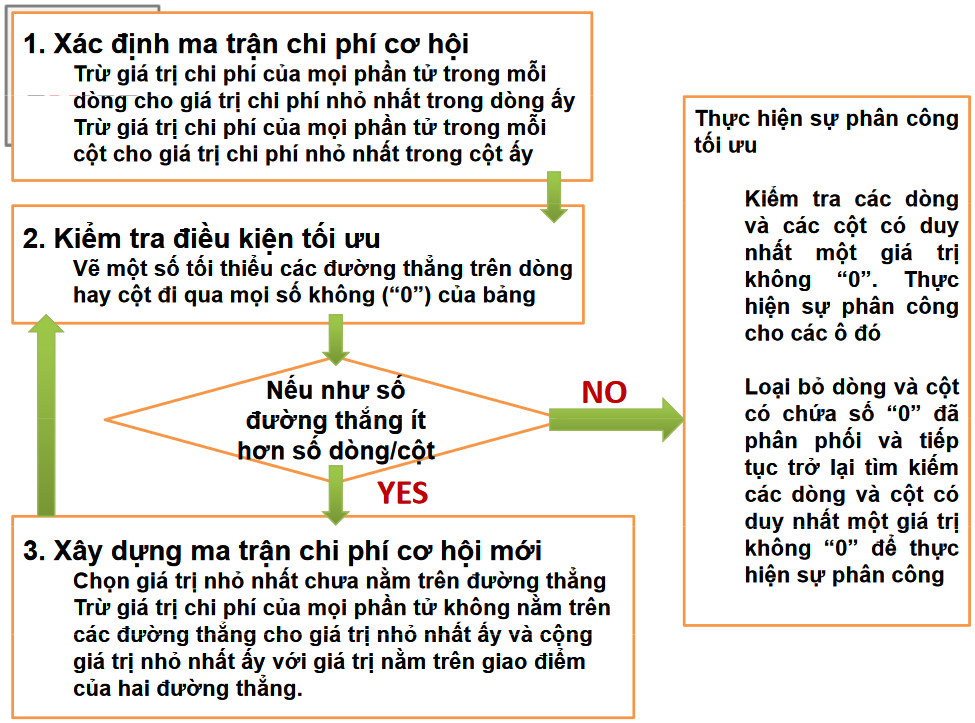
|  |  |  |  |
| --- | --- | --- | --- |
| M1 | 6 |  |  |
| M2 | 5 |  |  |
| M3 | 4 |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| M1 | 6 | 2 |  |
| M2 | 5 | 3 |  |
| M3 | 4 | 3 |  |

|  |  |  |  |
| --- | --- | --- | --- |
| M1 | 6 | 2 |  |
| M2 | 5 | 3 |  |
| M3 | 4 | 3 | 1 |

## Câu b

### Giải thích thuật toán

* Thuật toán Hungarian: dựa trên tính chất rút giảm ma trận. Khi trừ đi hay cộng thêm các giá trị thích hợp vào các phần tử ma trận chi phí ta sẽ có một ma trận chi phí cơ hội. Chi phí cơ hội là giá trị thiệt hại khi có sự phân công chưa phải là tối ưu nhất.
* Nếu ta có thể rút giảm ma trận đến khi có các phần tử có giá trị không “0” ở mỗi dòng và cột thì có thể đạt được sự phân công tối ưu vào các ô có giá trị không “0” đó.

### Ví dụ

* Có n = 8 công việc, m = 3 người. thời gian tij để người thứ j (j = 1, …, m) làm công việc thứ i (i = 1, …, n) là:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 | 6 | 6 | 4 | 7 | 8 | 8 | 12 | 8 |
| 2 | 6 | 5 | 7 | 8 | 9 | 7 | 8 | 5 |
| 3 | 6 | 6 | 7 | 9 | 10 | 6 | 5 | 7 |

#### Bước 1 xác định ma trận chi phí cơ hội

* Trừ giá trị của mọi phần tử trong mỗi dòng cho giá trị nhỏ nhất trong dòng ấy.
* Giá trị nhỏ nhất của từng dòng:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 | 6 | 6 | 4 | 7 | 8 | 8 | 12 | 8 |
| 2 | 6 | 5 | 7 | 8 | 9 | 7 | 8 | 5 |
| 3 | 6 | 6 | 7 | 9 | 10 | 6 | 5 | 7 |

* Chi phí cơ hội tính theo dòng:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 | 2 | 2 | 0 | 3 | 4 | 4 | 8 | 4 |
| 2 | 1 | 0 | 2 | 3 | 4 | 2 | 3 | 0 |
| 3 | 1 | 1 | 2 | 4 | 5 | 1 | 0 | 2 |

* Trừ giá trị của mọi phần tử trong mỗi cột cho giá trị nhỏ nhất trong cột ấy.
* Giá trị nhỏ nhất của từng cột:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 | 2 | 2 | 0 | 3 | 4 | 4 | 8 | 4 |
| 2 | 1 | 0 | 2 | 3 | 4 | 2 | 3 | 0 |
| 3 | 1 | 1 | 2 | 4 | 5 | 1 | 0 | 2 |

* Chi phí cơ hội tính theo cột:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 | 1 | 2 | 0 | 0 | 0 | 3 | 8 | 4 |
| 2 | 0 | 0 | 2 | 0 | 0 | 1 | 3 | 0 |
| 3 | 0 | 1 | 2 | 1 | 1 | 0 | 0 | 2 |

#### Bước 2 kiểm tra điều kiện tối ưu

* Vẽ một số tối thiểu các đường thẳng trên dòng hay cột đi qua mọi số không (“0”) của bảng.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 | 1 | 2 | 0 | 0 | 0 | 3 | 8 | 4 |
| 2 | 0 | 0 | 2 | 0 | 0 | 1 | 3 | 0 |
| 3 | 0 | 1 | 2 | 1 | 1 | 0 | 0 | 2 |

* Vì số đường thẳng = với số dòng nên thỏa điểu kiện tối ưu.

#### Bước 3 thực hiện phân công tối ưu

* Kiểm tra các dòng và các cột có duy nhất một giá trị không “0” Thực hiện sự phân công cho các ô đó. Loại bỏ dòng và cột có chứa số “0” đã phân phối và tiếp tục trở lại tìm kiếm các dòng và cột có duy nhất một giá trị không “0” để thực hiện sự phân công.
* Kiểm tra dòng, ta thấy không có dòng nào có 1 số không “0”.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 | 1 | 2 | 0 | 0 | 0 | 3 | 8 | 4 |
| 2 | 0 | 0 | 2 | 0 | 0 | 1 | 3 | 0 |
| 3 | 0 | 1 | 2 | 1 | 1 | 0 | 0 | 2 |

* Kiểm tra cột:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 | 1 | 2 | 0 | 0 | 0 | 3 | 8 | 4 |
| 2 | 0 | 0 | 2 | 0 | 0 | 1 | 3 | 0 |
| 3 | 0 | 1 | 2 | 1 | 1 | 0 | 0 | 2 |

* Vậy người 1 sẽ làm công việc 3.
* Người 2 sẽ làm công việc 2, 8.
* Người 3 sẽ làm công việc 7.
* Ta có mảng sau đây (công việc màu xanh là đã được phân công):

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 | 1 | 2 | 0 | 0 | 0 | 3 | 8 | 4 |
| 2 | 0 | 0 | 2 | 0 | 0 | 1 | 3 | 0 |
| 3 | 0 | 1 | 2 | 1 | 1 | 0 | 0 | 2 |

* Tiếp tục kiểm tra theo cột:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 | 1 | 2 | 0 | 0 | 0 | 3 | 8 | 4 |
| 2 | 0 | 0 | 2 | 0 | 0 | 1 | 3 | 0 |
| 3 | 0 | 1 | 2 | 1 | 1 | 0 | 0 | 2 |

* Vậy người 1 sẽ làm công việc 3.
* Người 2 sẽ làm công việc 2, 8.
* Người 3 sẽ làm công việc 6, 7.
* Ta có mảng sau đây (công việc màu xanh là đã được phân công):

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 | 1 | 2 | 0 | 0 | 0 | 3 | 8 | 4 |
| 2 | 0 | 0 | 2 | 0 | 0 | 1 | 3 | 0 |
| 3 | 0 | 1 | 2 | 1 | 1 | 0 | 0 | 2 |

* Tiếp tục kiểm tra theo dòng:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 | 1 | 2 | 0 | 0 | 0 | 3 | 8 | 4 |
| 2 | 0 | 0 | 2 | 0 | 0 | 1 | 3 | 0 |
| 3 | 0 | 1 | 2 | 1 | 1 | 0 | 0 | 2 |

* Vậy người 1 sẽ làm công việc 1, 3.
* Người 2 sẽ làm công việc 2, 8.
* Người 3 sẽ làm công việc 6, 7.

Ta có mảng sau đây (công việc màu xanh là đã được phân công):

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 | 1 | 2 | 0 | 0 | 0 | 3 | 8 | 4 |
| 2 | 0 | 0 | 2 | 0 | 0 | 1 | 3 | 0 |
| 3 | 0 | 1 | 2 | 1 | 1 | 0 | 0 | 2 |

# Bài 2

## Câu a

* a->f,a->(f->p),p^q->d,a,a^d->g
  + a->f: -a v f
  + a -> (f -> p) ⬄ a -> (-f v p) ⬄ -a v (-f v p )
  + p^q->d : (-p v –q) v d
  + a
  + a^d -> g : (-a v –d) v g
    - (-a v f) ^( -a v (-f v p )) ^( (-p v –q) v d) ^ a ^ ((-a v –d) v g) => g (\*)

Để chứng minh (\*) đúng ta tách (\*) thành 2 biểu thức cần chứng minh đúng

* + –a ^ (-a v (-f v p )) ^( (-p v –q) v d) ^ a ^( (-a v –d) v g) => g (1)
  + f ^ (-a v (-f v p)) ^ (-p v –q v d) ^ a ^ (-a v -d v g) => g (2)
    - (1) Đúng vì VT có –a ^ a
    - (2) –( f ^ -a v (-f v p)) ^ (-p v –q v d) ^ a ^ (-a v -d v g)) v g

⬄ -f v – ( -a v (-f v p)) v – (-p v –q v d) v – a v a ^ d ^ - g v g

(2) Đúng vì có - g v g

**Kết luận g đúng**

## Câu b

* Giải thuật Robinson:
  + B1: Dạng chuẩn: -a v f, -a v (-f v p ), (-p v –q) v d, a, (-a v –d) v g=>g
  + B2: CHuyển vế kết luận : -a v f, -a v (-f v p ), (-p v –q) v d, a, (-a v –d) v g , -g
  + B3: Tuyển cặp mệnh đề đối ngẫu
    - -a v –f v –p v d

# Bài 3

a. Trình bày phương pháp biểu diễn tri thức theo luật dẫn:

- Khái niệm luật dẫn: là một kiểu biểu diễn tri thức có cấu trúc. Ý tưởng cơ bản là tri thức có thể được cấu trúc bằng một cặp điều kiện – hành động:”Nếu điều kiện xảy ra thì hành động sẽ được thi hành”

Ví dụ: NẾU đèn giao thông là đỏ THÌ bạn không được đi thẳng, NẾU máy tính đã mở mà không khởi động được THÌ kiểm tra nguồn điện, …

Ngày nay, các luật đã trở nên phổ biến và được áp dụng rộng rãi trong nhiều hệ thống trí tuệ nhân tạo khác nhau. Luật dẫn có thể là một công cụ mô tả để giải quyết các vấn đề thực tế thay cho các kiểu phân tích vấn đề truyền thống. Trong trường hợp này, các luật được dùng như là những chỉ dẫn (tuy có thể không hoàn chỉnh) nhưng rất hữu ích để trợ giúp cho các quyết định trong quá trình tìm kiếm, từ đó làm giảm không gian tìm kiếm. Một ví dụ khác là luật sinh có thể được dùng để bắt chước hành vi của những chuyên gia. Theo cách này, luật dẫn không chỉ đơn thuần là một kiểu biểu diễn tri thức trong máy tính mà là một kiểu biễu diễn các hành vi của con người.

b. Các thành phần cơ bản trong mô hình tri thức luật dẫn

- Tập các sự kiền (Facts)

F = {f1,f2,…,fn}.

-Tập các quy tắc R(Rules) áp dụng trên các sự kiện dạng như sau:

f1^f2^…fi 🡪 q

Trong đó, các fi, q đều thuộc F.

c. Trình bày phương pháp suy diễn tiến trên hệ luật dẫn

Suy diễn tiến: là quá trình suy luận xuất pháp từ một số sự kiện ban đầu, xác định các sự kiện có thể được “sinh” ra từ sự kiện này

Các ký hiệu:

**Hypos**: tập các sự kiện giả thiết

**Goals**: tập các sự kiện mục tiêu

**facts**: tập các sự kiện đã biết

**rules**: tập các luật được áp dụng để sinh ra sự kiện mới

**R**: tập luật trong Cơ sở tri thức

**Backward\_Chaining:**

Bước 1: **facts** =**Hypos**;

**rules** = [];

Bước 2:

while **Goals** Ë **facts** do

Bước 2.1: Tìm luật r **R** có thể áp dụng được trên **facts** để sinh

ra sự kiện mới.

Bước 2.2: Nếu tìm được r thì:

+ Bổ sung r vào **rules**

+ Bổ sung sự kiện mới vào **facts**

Ngược lại: Kết thúc suy luận, không tìm được lời giải

Bước 3: Kết luận tìm được mục tiêu

**Giải:**

Sự kiện ban đầu:{H, K}

Xét r3: H -> A {A, H, K}

Xét r1: A -> E {A, E, H, K}

Xét r5: E **^** K -> B {A, B, E, H, K}

Xét r2: B -> D {A, B, D, E, H, K}

Xét r6: D **^** E **^** K -> C {A, B, C, D, E, H, K}