Báo Cáo Bài Tập Lớn

môn

Trí Tuệ Nhân Tạo

Nhóm x

Cấn Duy Cát  
Nguyễn Minh Trang  
Nguyễn Mạnh Duy  
Kiều Minh Đức  
Nguyễn Mạnh Cường

01-04-2016

Mục lục

Table of Contents ii

Revision History iii

1. Introduction 1

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Product Scope 1

1.5 References 2

2. Architectural Mechanisms 3

2.1 Analysis Mechanisms 3

2.1.1 Persistence mechanisms 3

2.1.2 Communication mechanisms 3

2.1.3 Security mechanisms 3

2.1.4 The other mechanisms 3

2.2 Analysis-to-Design-to-Implementation Mechanisms Map 4

2.3 Implementation Mechanisms 5

2.3.1 Security 5

2.3.2 Persistency – OODBMS – Object Store 8

2.3.3 Distribution – RMI 16

3. Logical View 18

3.1 Architectural Analysis 18

3.1.1 Upper-Level Layers 18

3.1.2 Upper-Level Layer Dependencies 18

3.2 Architectural Design 18

3.2.1 Incorporating Object Store 18

3.2.2 Architectural Layers and Their Dependencies 19

3.2.3 Packages and Their Dependencies 20

4. Process View 22

4.1 Processes 22

4.2 Design Element to Process Mapping 22

5. Deployment View 23

5.1 Nodes and Connections 23

5.2 Process-to-Node Map 23

Lịch sử thay đổi

|  |  |  |  |
| --- | --- | --- | --- |
| Người thay đổi | Ngày thay đổi | Lý do thay đổi | Phiên bản |
| Cat Can | 13-Mar-2016 | Create document template | 1.0 |
| Cat Can | 13-Mar-2016 | Finish Analysis Mechanisms | 1.1 |
| Cat Can | 21-Mar-2016 | Add some Analysis Mechanisms detail | 1.2 |
| Cat Can | 1-Apr-2016 | Update Analysis to Design to Implementation Mechanisms Map | 1.3 |
| Dung Ngo | 2-Apr-2016 | Add Persistency – OODBMS – Object Store | 1.4 |
| Trang Nguyen | 3-Apr-2016 | Finish Distribution – RMI | 1.5 |
| Luong Nguyen | 4-Apr-2016 | Add Security Implement Mechanism | 1.6 |
| Cat Can | 4-Apr-2016 | Add Logical View | 1.7 |
| Cat Can | 4-Apr-2016 | Paging, figure numbering | 1.8 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Giới thiệu

## Mục đích

## Nội dung tài liệu

* Phần 1: Không gian trạng thái.
* Phần 2: Phát biểu bài toán.
* Phần 3: Hàm mục tiêu.
* Phần 4: So sánh hàm đánh giá.

## Tài liệu tham khảo

[1] <http://www.cs.columbia.edu/~kathy/cs4701/Assignments/hw2-kt.html>

# Không gian trạng thái

* State :

Trong bảng ô vuông 3 hàng 3 cột,mỗi ô chứa 1 chữ có trong 2 tập input của bài toán là :Bảng 3 hàng – 3 cột , mỗi ô chứa 1 chữ có trong 2 tập input của bài toán đã cho là :

[S,O,P,E,A,R,W,R,Y]

[A,P,E,I,L,K,L,Y,E]

Do đó có 3² trạng thái có thể có trong bài toán náy ,các chữ cái có thể nằm bất kỳ vị trí nào trong 9 ô .Vì vậy có thể mô tả trạng thái của bài toán bằng một ma trận A3\*3= (aij) , aij∈{1..9}, {i,j=[0,1,2]}.Với các trường hợp mở rộng bảng n hàng – n cột thì số lượng trạng thái có thể là n² .

* Initial State :

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

Ta sẽ đẩy tất các các input vào trong ma trận ( như hình trên )

* Actions:

Với mỗi trạng thái s, ta sẽ thêm liên tiếp các chữ cái vào các ô vuông ( như trong hình ) . Ví dụ : Action(s) = {Add(a)}.

* Transition Model :

Với mỗi Action trên thì Transition Model sẽ trả về tập các chữ cái để tạo nên các từ có nghĩa ( cũng có thể không có nghĩa ) .Ví dụ với Action(s) ở trên và Action(s1)={Add(y)}, Action(s2) = {Add(e)}. Ta sẽ có Transition Model trả về Result ( s,s1,s2) = “aye” ( đây là một từ có nghĩa ).

* Path Cost :

Mỗi bước thêm chữ cái sẽ có giá trị là 1.

* Goal Test :

Khi chọn 1 chữ cái, nếu nó là chữ cái cuối cùng của bất kỳ từ nào có trong bảng thì từ đó phải có nghĩa .

Khi chọn song 9 chữ cái ( đầy bảng ) thì các hàng ngang , dọc , chéo phải tạo thành các từ có nghĩa .

# Phát biểu bài toán

Cường dịch

Set up the problem in such a way that you can consider one path through the matrix (e.g. 1st row, followed by 2nd row, followed by 3rd row). At each move, the search algorithm will choose the next letter in the path. The successor function must ensure when it chooses a letter that, if it is the last letter in any word on the board, the resulting word is valid.

# Hàm mục tiêu

Cường. The successor function will return all letters that can go into the next empty space on the path.

# So sánh hàm đánh giá

## Phân tích hàm đánh giá

### Hàm đánh giá thông thường

Hàm đánh giá sẽ trả về tần số của cặp chữ cái được hình thành bởi những chữ cái đã được lựa chọn với chữ cái kế tiếp. Như vậy, chữ cái kế tiếp được lựa chọn sao cho tần suất của cặp chữ cái đó là lớn nhất. Đối với chữ cái đầu tiên, ta lựa chọn chữ cái có tần suất ban đầu lớn nhất (tần suất này cũng có trong danh sách cặp từ).

Hàm đánh

### Hàm đánh giá nâng cao

## So sánh

ảnh chụp cái này t cho sau