

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**Khoa: Công nghệ thông tin**



**BÁO CÁO ĐỒ ÁN MÔN HỌC  
CƠ SỞ TRÍ TUỆ NHÂN TẠO**

Lab02: Logic

**Giảng viên hướng dẫn:**

Nguyễn Duy Khánh

Nguyễn Ngọc Băng Tâm

Thành phố Hồ Chí Minh, ngày 04 tháng 12 năm 2022

MỤC LỤC

- I. Mô tả & đánh giá thuật toán .....1
- II. Các test cases .....2
  - 1. Test case 1 .....2
  - 2. Test case 2 .....2
  - 3. Test case 3 .....3
  - 4. Test case 4 .....3
  - 5. Test case 5 .....4
- III. Tài liệu tham khảo .....5

## I. Mô tả & đánh giá thuật toán

Ta xây dựng một lớp **algorithms** để giải quyết yêu cầu bài toán, bao gồm những thuộc tính sau khi được khởi tạo:

- Một list **alpha[]** để chứa câu *alpha*
- Một list **KB[]** để chứa cơ sở tri thức *KB* từ đề bài
- Một list **new\_clauses[]** để chứa các mệnh đề mới được sinh ra từ hợp giải
- Một biến boolean **is\_solved** (giá trị mặc định False) để xác định *KB* entails *alpha* hay không.

Các hàm chính giải quyết bài toán bao gồm:

- Hàm kiểm tra tính đối ngẫu của các literals (**is\_inverse()**)
- Hàm kiểm tra một mệnh đề có luôn là chân trị True hay không, mệnh đề cần được loại bỏ vì không giúp ích cho việc hợp giải (**is\_always\_true()**)
- Hàm hợp giải hai mệnh đề tạo mệnh đề mới (**resolve()**)
- Hàm xử lý hợp giải các mệnh đề từ tập hội của cơ sở tri thức *KB* và phủ định của mệnh đề *alpha*, từ các mệnh đề mới xác định *KB* entails *alpha* (**pl\_resolution()**)

Và một số hàm phụ hỗ trợ: hàm lấy phủ định literal, hàm chuyển đổi mệnh đề từ file input sang dạng list các literals và ngược lại, hàm đọc-ghi file input, output.

Mô tả thuật toán:

- Hàm **pl\_resolution()** tạo một **clauses\_list** chứa *KB* và phủ định của mệnh đề *alpha*, và một vòng lặp với điều kiện kết thúc là khi không tìm được mệnh đề mới từ mệnh đề hợp giải hoặc tìm được lời giải (*alpha* và phủ định *alpha* cùng nằm trong **clauses\_list**). Hàm trả về giá trị boolean True hoặc False.

- Ta dùng hai vòng lặp lồng nhau để tiến hành hợp giải từng bộ 2 mệnh đề trong **clauses\_list**. Nếu hợp giải được mệnh đề rỗng “{}” tức là đã tìm được lời giải (**is\_solved** bật sang True). Kiểm tra các mệnh đề vừa được hợp giải, nếu chúng không luôn mang chân trị True và không nằm trong tập **clauses\_list** lẫn **new\_clauses** thì ta thêm chúng vào tập **new\_clauses** của lớp thuật toán đồng thời thêm vào tập hội **clauses\_list** ban đầu.

- Để tối ưu vòng lặp, hàm có sử dụng chỉ mục tạm đánh dấu **idx**:

- Với vòng lặp ban đầu, vòng **for j** sẽ bắt đầu từ vị trí **j = i + 1**.
- Sau khi chạy hết vòng lặp ban đầu và có các mệnh đề mới trong **clauses\_list**, ta cập nhật **idx** bằng vị trí của mệnh đề mới đầu tiên.
- Từ vòng lặp thứ 2 trở đi, vòng **for j** sẽ bắt đầu từ vị trí **j = idx** khi **i < idx**, như vậy các mệnh đề cũ hợp giải với nhau sẽ không cần hợp giải lại lần nữa mà sẽ chỉ hợp giải với các mệnh đề mới sinh ra. Khi **i = idx** tức là đã đến vị trí chỉ gồm những mệnh đề mới phía sau thì vòng **for j** chạy tương tự như vòng lặp ban đầu.

Đánh giá thuật toán:

**Ưu điểm:** dễ cài đặt, dễ hiểu, hợp giải tạo ra đầy đủ các mệnh đề.

**Nhược điểm:** độ phức tạp vẫn còn lớn, chưa thực sự tối ưu. Nếu được cần cài đặt giải thuật chọn lựa thông minh các cặp mệnh đề có khả năng hợp giải để tiết kiệm thời gian.

## II. Các test cases

### 1. Test case 1

input_1.txt	output_1.txt	Giải thích
D	5	
4	-B OR C	A with -A OR -B OR C
A	-A OR C	B with -A OR -B OR C
B	-C OR D	B with -B OR -C OR D
-A OR -B OR C	-A OR -B OR D	-A OR -B OR C with -B OR -C OR D
-B OR -C OR D	-B OR -C	-B OR -C OR D with -D
	6	
	C	A with -A OR C
	-B OR D	A with -A OR -B OR D
	-A OR D	B with -A OR -B OR D
	-C	B with -B OR -C
	-A OR -B	-A OR -B OR C with -B OR -C
	-B	-B OR C with -B OR -C
	3	
	D	A with -A OR D
	-A	B with -A OR -B
	{}	B with -B
	YES	KB entails alpha

### 2. Test case 2

input_2.txt	output_2.txt	Giải thích
-P OR -T	5	
4	-P OR R	-P OR Q with -Q OR R
-P OR Q	Q	-P OR Q with P
-Q OR R	-Q OR S	-Q OR R with -R OR S
-R OR S	-R OR -T	-R OR S with -S OR -T
-S OR -T	-S	-S OR -T with T
	7	
	-P OR S	-P OR Q with -Q OR S
	R	-Q OR R with Q
	-Q OR -T	-Q OR R with -R OR -T
	-R	-R OR S with -S
	-P OR -T	-P OR R with -R OR -T
	S	Q with -Q OR S
	-Q	-Q OR S with -S
	3	
	-P	-P OR Q with -Q
	-T	-S OR -T with S
	{}	Q with -Q
	YES	KB entails alpha

### 3. Test case 3

input_3.txt	output_3.txt	Giải thích
-A OR -B OR D 2 -A OR -B OR C -B OR -C OR D	5 -A OR -B OR D -B OR C -A OR C -C OR D -B OR -C 6 -A OR -B -B OR D C -A OR D -C -B 3 D -A { }	-A OR -B OR C with -B OR -C OR D -A OR -B OR C with A -A OR -B OR C with B -B OR -C OR D with B -B OR -C OR D with -D  -A OR -B OR C with -B OR -C -B OR -C OR D with -B OR C A with -A OR C B with -A OR -B OR D B with -B OR -C -B OR C with -B OR -C  A with -A OR D B with -A OR -B B with -B KB entails alpha

### 4. Test case 4

input_4.txt	output_4.txt	Giải thích
-A OR -D 4 -A OR B OR D -D OR E -D OR F -A OR -E OR -B	8 -A OR B OR E -A OR B OR F -A OR D OR -E B OR D -A OR -B OR -D E F -B OR -E 10 B OR E -B OR -D -A OR -E OR F B OR F -A OR -B D OR -E -A OR -D OR E -A OR -D OR F -A OR D -B 3 -A OR E	-A OR B OR D with -D OR E -A OR B OR D with -D OR F -A OR B OR D with -A OR -B OR -E -A OR B OR D with A -D OR E with -A OR -B OR -E -D OR E with D -D OR F with D -A OR -B OR -E with A  -D OR E with B OR D -D OR E with -B OR -E -D OR F with -A OR D OR -E -D OR F with B OR D -A OR -B OR -E with E A with -A OR D OR -E -A OR B OR E with -A OR -B OR -D -A OR B OR F with -A OR -B OR -D -A OR D OR -E with E E with -B OR -E  -D OR E with -A OR D

	-E OR F -A OR F 0 NO	-D OR F with D OR -E -D OR F with -A OR D Không phát sinh mệnh đề mới KB không entails alpha
--	-------------------------------	---

## 5. Test case 5

input_5.txt	output_5.txt	Giải thích
-G 7 -A OR B -A OR C OR E -B OR -C OR D -E OR F -D OR G -F OR G A	7 -A OR -C OR D B -A OR -B OR D OR E -A OR C OR F C OR E -B OR -C OR G -E OR G 14 -A OR D OR E -A OR -C OR G -A OR -B OR E OR G -A OR C OR G -C OR D -A OR -B OR D OR F -B OR D OR E C OR F -A OR D OR F -C OR G -A OR -B OR D OR G -A OR -B OR F OR G -B OR E OR G C OR G 14 -A OR E OR G -A OR D OR G -A OR F OR G -B OR D OR F -B OR D OR G -B OR F OR G -A OR -B OR G D OR E D OR F E OR G -B OR G -A OR G D OR G F OR G 0	-A OR B with -B OR -C OR D -A OR B with A -A OR C OR E with -B OR -C OR D -A OR C OR E with -E OR F -A OR C OR E with A -B OR -C OR D with -D OR G -E OR F with -F OR G  -A OR B with -A OR -B OR D OR E -A OR B with -B OR -C OR G -A OR C OR E with -B OR -C OR G -A OR C OR E with -E OR G -B OR -C OR D with B -B OR -C OR D with -A OR C OR F -B OR -C OR D with C OR E -E OR F with C OR E -A OR -C OR D with -A OR C OR F B with -B OR -C OR G -A OR -B OR D OR E with -E OR G -A OR C OR F with -B OR -C OR G C OR E with -B OR -C OR G C OR E with -E OR G  -A OR B with -A OR -B OR E OR G -A OR B with -A OR -B OR D OR G -A OR B with -A OR -B OR F OR G -B OR -C OR D with C OR F -B OR -C OR D with C OR G -E OR F with -B OR E OR G -D OR G with -A OR -B OR D OR G A with -A OR D OR E A with -A OR D OR F B with -B OR E OR G -B OR -C OR G with C OR G -A OR -C OR G with -A OR C OR G -C OR D with C OR G C OR F with -C OR G Không phát sinh mệnh đề mới

	NO	KB không entails alpha
--	----	------------------------

### III. Tài liệu tham khảo

1. [Artificial Intelligence: A Modern Approach \(4<sup>th</sup> edition\)](#)
2. Idea from github: [pl-resolution](#)