

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**  
**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**  
**KHOA: CÔNG NGHỆ THÔNG TIN**



## **BÁO CÁO LAB 2**

**ĐỀ TÀI: LOGIC**

**Môn học: CƠ SỞ TRÍ TUỆ NHÂN TẠO**

*Sinh viên thực hiện: Trần Đình Nhật Trí*  
*Mã số sinh viên: 21120576*

*Giảng viên lý thuyết: GS. TS. Lê Hoài Bắc*  
*Giảng viên thực hành: Nguyễn Ngọc Đức*

**Thành phố Hồ Chí Minh, ngày 27 tháng 11 năm 2023**

# MỤC LỤC

I.	KỊCH BẢN KIỂM THỬ: .....	3
1.	Kịch bản 1: .....	3
2.	Kịch bản 2: .....	3
3.	Kịch bản 3: .....	4
4.	Kịch bản 4: .....	5
5.	Kịch bản 5: .....	5
II.	ƯU ĐIỂM, NHƯỢC ĐIỂM CỦA GIẢI THUẬT HỢP GIẢI VÀ GIẢI PHÁP KHẮC PHỤC VẤN ĐỀ: .....	7
1.	Ưu điểm: .....	7
2.	Nhược điểm: .....	7
3.	Giải pháp khắc phục vấn đề: .....	7
III.	TIÊU CHÍ ĐÁNH GIÁ .....	9
IV.	TÀI LIỆU THAM KHẢO .....	9

## I. KỊCH BẢN KIỂM THỬ:

### 1. Kịch bản 1:

Input	Output	Ghi chú
<b>P OR R OR Q</b>	7	Tách (P OR R OR Q) thành 3 mệnh đề sau đó negate thành (-P), (R) và (Q)
<b>4</b>	P OR Q	(P OR -R) hợp giải với (Q OR R)
<b>P OR -R</b>	Q OR -R	(P OR -R) hợp giải với (-P OR Q OR -R)
<b>R OR Q</b>	P OR R	(Q OR R) hợp giải với (P OR -Q)
<b>-Q OR P</b>	-P OR Q	(Q OR R) hợp giải với (-P OR Q OR -R)
<b>-P OR -R OR Q</b>	Q	(Q OR R) hợp giải với (-R)
	R	(Q OR R) hợp giải với (-Q)
	-P OR -R	(-P OR Q OR -R) hợp giải với (-Q)
	3	
	P	(P OR -R) hợp giải với (P OR R)
	-Q OR -R	(P OR -Q) hợp giải với (-P OR -R)
	{}	(-R) hợp giải (R)
	YES	

### 2. Kịch bản 2:

Input	Output	Ghi chú
<b>-A</b>	5	
<b>5</b>	-A OR -B	(-A OR -B OR C) hợp giải với (-C)
<b>-A OR -B OR C</b>	-B OR C	(-A OR -B OR C) hợp giải với (negative -A)
<b>-A OR C</b>	-A OR B	(-A OR C) hợp giải với (B OR -C)
<b>B OR -C</b>	-A	(-A OR C) hợp giải với (-C)
<b>-C</b>	C	(-A OR C) hợp giải với (negative -A)
<b>-B</b>	3	
	-A OR -C	(B OR -C) hợp giải với (-A OR -B)

B	(B OR -C) hợp giải với (C)
{}	(-C) hợp giải với (C)
YES	

### 3. Kịch bản 3:

Input	Output	Ghi chú
-S OR U	3	Tách (-S OR U) thành 2 mệnh đề sau đó negate thành (S), và (-U)
6	S OR T	(S OR U) hợp giải với (T OR -U)
S OR U	Q OR -S OR -U	(T OR -U) hợp giải với (Q OR -S OR -T OR -U)
-U OR T	Q OR -T OR -U	(Q OR -S OR -T OR -U) hợp giải với (S)
T OR S OR -U	5	
-T OR -S OR -U OR Q	Q OR S OR -T	(S OR U) hợp giải với (Q OR -T OR -U)
-Q OR S	Q OR -U	(T OR -U) hợp giải với (Q OR -T OR -U)
-Q OR T OR S	Q OR T OR -U	(S OR T OR -U) hợp giải với (Q OR -S OR -U)
	Q OR S OR -U	(S OR T OR -U) hợp giải với (Q OR -T OR -U)
	S OR -T OR -U	(-Q OR S) hợp giải với (Q OR -T OR -U)
	6	
	Q OR S	(S OR U) hợp giải với (Q OR -U)
	Q OR S OR T	(S OR U) hợp giải với (Q OR T OR -U)
	S OR -T	(S OR U) hợp giải với (S OR -T OR -U)
	S OR -U	(T OR -U) hợp giải với (S OR -T OR -U)
	S OR T OR -U	(-Q OR S) hợp giải với (Q OR T OR -U)
	-Q OR S OR -U	(-Q OR S OR T) hợp giải với (S OR -T OR -U)
	0	
NO		KB KHÔNG entail $\alpha$ vì không phát sinh được mệnh đề mới và không tìm thấy mệnh đề rỗng.

#### 4. Kịch bản 4:

Input.txt	Output.txt	Ghi chú
<b>A OR -B OR -C</b>	10	Tách (A OR -B OR -C) thành 3 mệnh đề sau đó negate thành (-A), (B) và (C)
<b>7</b>	A OR -D	(A OR -C) hợp giải với (C OR -D)
<b>A OR -C</b>	-C	(A OR -C) hợp giải với (-A)
<b>B OR -D</b>	A	(A OR -C) hợp giải với (C)
<b>-C OR D</b>	B OR -C	(B OR -D) hợp giải với (-C OR D)
<b>A OR -C OR D</b>	A OR B OR -C	(B OR -D) hợp giải với (A OR -C OR D)
<b>C OR -D</b>	-C OR -D	(B OR -D) hợp giải với (-B OR -C)
<b>A OR -B</b>	D	(-C OR D) hợp giải với (C)
<b>-B OR -C</b>	A OR D	(A OR -C OR D) hợp giải với (C)
	-B OR -D	(C OR -D) hợp giải với (-B OR -C)
	-B	(A OR -B) hợp giải với (-A)
	6	
	A OR B	(B OR -D) hợp giải với (A OR D)
	-D	(B OR -D) hợp giải với (-B OR -D)
	A OR -B OR -C	(A OR -C OR D) hợp giải với (-B OR -D)
	A OR B OR -D	(C OR -D) hợp giải với (A OR B OR -C)
	A OR C	(C OR -D) hợp giải với (A OR D)
	{}	(-A) hợp giải với (A)
	<b>YES</b>	

#### 5. Kịch bản 5:

Input	Output	Ghi chú
<b>P OR -R OR S OR T OR Z</b>	6	Tách (P OR -R OR S OR T OR Z) thành 5 mệnh đề sau đó negate thành (-P), (R), (-S), (-T) và (-Z)
<b>4</b>	S OR -T OR -Z	(-R OR -T OR -Z) hợp giải với (R OR S OR -T)
<b>-R OR -T OR -Z</b>	-Q OR -R OR -S OR -Z	(-R OR -T OR -Z) hợp giải với (-Q OR -S OR T)
<b>R OR S OR -T</b>	-T OR -Z	(-R OR -T OR -Z) hợp giải với (R)
<b>P OR -Q OR -S</b>	P OR -Q OR R OR -T	(R OR S OR -T) hợp giải với (P OR -Q OR -S)

<b>-Q OR -S OR T</b>	R OR -T	(R OR S OR -T) hợp giải với (-S)
<b>P OR -R OR S OR T OR Z</b>	-Q OR -S	(P OR -Q OR -S) hợp giải với (-P)
<b>4</b>	9	
<b>-R OR -T OR -Z</b>	P OR -Q OR -T OR -Z	(-R OR -T OR -Z) hợp giải với (P OR -Q OR R OR -T)
<b>R OR S OR -T</b>	-Q OR R OR -T	(R OR S OR -T) hợp giải với (-Q OR -S)
	-Q OR -S OR -Z	(-Q OR -S OR T) hợp giải với (-T OR -Z)
	P OR -Q OR R OR -S	(-Q OR -S OR T) hợp giải với (P OR -Q OR R OR -T)
	-Q OR R OR -S	(-Q OR -S OR T) hợp giải với (R OR -T)
	-Q OR -R OR -T OR -Z	(S OR -T OR -Z) hợp giải với (-Q OR -R OR -S OR -Z)
	-Q OR -T OR -Z	(S OR -T OR -Z) hợp giải với (-Q OR -S)
	P OR -Q OR -S OR -T OR -Z	(-Q OR -R OR -S OR -Z) hợp giải với (P OR -Q OR R OR -T)
	-Q OR -S OR -T OR -Z	(-Q OR -R OR -S OR -Z) hợp giải với (R OR -T)
	4	
	-Q OR R OR -T OR -Z	(R OR S OR -T) hợp giải với (-Q OR -S OR -Z)
	-Q OR S OR -T OR -Z	(R OR S OR -T) hợp giải với (-Q OR -R OR -T OR -Z)
	P OR -Q OR R OR -T OR -Z	(R OR S OR -T) hợp giải với (P OR -Q OR -S OR -T OR -Z)
	P OR -Q OR -S OR -Z	(-Q OR -S OR T) hợp giải với (P OR -Q OR -T OR -Z)
	2	
	-Q OR R OR -S OR -Z	(-Q OR -S OR T) hợp giải với (-Q OR R OR -T OR -Z)
	P OR -Q OR R OR -S OR -Z	(-Q OR -S OR T) hợp giải với (P OR -Q OR R OR -T OR -Z)
	0	
	<b>NO</b>	KB KHÔNG entail $\alpha$ vì không phát sinh được mệnh đề mới và không tìm thấy mệnh đề rỗng.

## **II. ƯU ĐIỂM, NHƯỢC ĐIỂM CỦA GIẢI THUẬT HỢP GIẢI VÀ GIẢI PHÁP KHẮC PHỤC VẤN ĐỀ:**

### ***1. Ưu điểm:***

- Độ hoàn thiện cao: Nếu một tập hợp các mệnh đề không thỏa mãn, thì kết thúc quá trình hợp giải của các mệnh đề đó chứa mệnh đề trống, đảm bảo tính hoàn thiện của kết quả.
- Độ chính xác cao: Kết quả trả về luôn là hệ quả được suy ra từ cơ sở tri thức ban đầu, đảm bảo tính đúng đắn của kết quả.
- Là phương pháp đơn giản và dễ hiểu, phù hợp với các vấn đề logic mệnh đề cơ bản.
- Được sử dụng rộng rãi trong lĩnh vực trí tuệ nhân tạo và hình thành cơ sở cho nhiều phương pháp giải quyết vấn đề khác.
- Có thể được kết hợp với các phương pháp khác để tăng tốc độ giải quyết vấn đề.

### ***2. Nhược điểm:***

- Mỗi cặp mệnh đề hợp giải nhau có thể tạo ra nhiều kết quả và các kết quả này không đóng góp vào quá trình chứng minh đúng. Điều này có thể dẫn đến việc phải xử lý một lượng lớn các kết quả không cần thiết, gây lãng phí thời gian và tài nguyên tính toán.
- Hệ thống đòi hỏi vết cạn sẽ duyệt qua hết các trường hợp hợp giải, điều này có thể gây mất nhiều thời gian và không khả thi đối với các vấn đề có số lượng biến và mệnh đề lớn.
- Giải thuật hợp giải có thể trở nên rất chậm và tốn nhiều tài nguyên tính toán khi áp dụng cho các vấn đề lớn với số lượng biến và mệnh đề lớn.
- Có thể không thể giải quyết được các vấn đề với các mệnh đề phức tạp hoặc có tính khó xác định cao.

### ***3. Giải pháp khắc phục vấn đề:***

- Phương pháp Davis-Putnam:  
Thuật toán giải quyết SAT (Satisfiability) ban đầu của Davis và Putnam, hiện được gọi là thủ tục Davis-Putnam, có thể nhanh hơn nhiều so với việc chỉ "thử tất cả". Thủ tục này dựa trên việc chúng ta thường có thể xác định giá trị

đúng/sai của một mệnh đề hoặc công thức mà không cần biết tất cả các giá trị của các biến của nó, vì một đối số đúng đơn lẻ sẽ làm cho mệnh đề đó là đúng và một mệnh đề sai đơn lẻ sẽ làm cho công thức là sai.

---

**Algorithm 1** Davis-Putnam Procedure

---

```

1: procedure DAVIS-PUTNAM( $\phi$ )
2:    $S \leftarrow \{(\phi, [])\}$   $\triangleright$  Initially, S contains only a pairing of  $\phi$  with the empty truth assignment.
3:   while  $S$  is not empty do
4:      $(\phi', t) \leftarrow$  an arbitrary element of  $S$ 
5:      $x \leftarrow$  an arbitrary literal in  $\phi'$ 
6:      $S \leftarrow (S - \{(\phi', t)\}) \cup \text{CHASE}(\phi, x, t) \cup \text{CHASE}(\phi, \bar{x}, t)$ 
7:   end while
8: end procedure

9: function CHASE( $\phi, x, t$ )
10:  Set  $x$  to true in  $t$ 
11:  Delete all clauses from  $\phi$  that contain the literal  $x$ 
12:  Delete the literal  $\bar{x}$  from all clauses in  $\phi$ 
13:  if  $\phi$  is empty then abort from DAVIS-PUTNAM with  $t$   $\triangleright t$  is a satisfying assignment
14:  elseif  $\phi$  contains an empty clause then return  $\{\}$   $\triangleright t$  contains bad decisions
15:  elseif  $\phi$  contains a unit clause ( $y$ ) then return CHASE( $\phi, y, t$ )  $\triangleright$  Continue the chase
16:  else return  $\{(\phi, t)\}$ 
17: end function

```

---

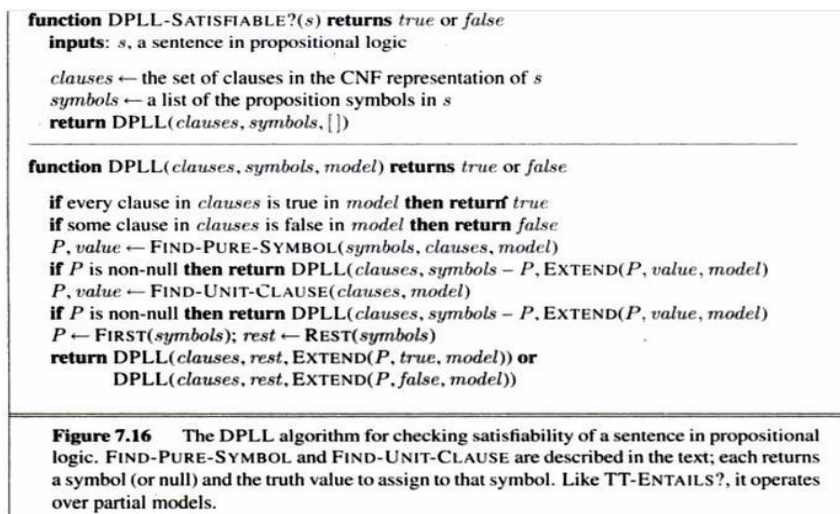
Hình 1. Mã giả thuật giải Davis-Putnam

- Phương pháp DPLL:

Thuật toán này là phiên bản được mô tả bởi Davis, Logemann, Hilary Putnam (1960) và Loveland (1962), vì vậy nó được gọi là DPLL theo tên viết tắt của cả bốn tác giả. DPLL lấy đầu vào là một câu ở dạng thông thường liên kết—một tập hợp các mệnh đề. Giống như BACKTRACKINGSEARCH và TT-ENTAILS, nó thực chất là một phép liệt kê đệ quy, theo chiều sâu đầu tiên của các mô hình có thể. Nó thể hiện ba cải tiến so với sơ đồ đơn giản của TT-ENTAILS:

- Chấm dứt sớm: Thuật toán phát hiện xem câu phải đúng hay sai, ngay cả với mô hình đã hoàn thành một phần. Một mệnh đề là đúng nếu bất kỳ literal nào cũng đúng, ngay cả khi các literal khác chưa có chân trị. Do đó, toàn bộ suy luận có thể được đánh giá là đúng ngay cả trước khi mô hình hoàn thành.
- Pure symbol heuristic: ký hiệu thuần túy (pure symbol) là một ký hiệu luôn xuất hiện với cùng “một dấu” trong tất cả mệnh đề.
- Mệnh đề đơn vị heuristic: Mệnh đề đơn vị được định nghĩa trước đó là một mệnh đề chỉ có một đối số. Trong ngữ cảnh của DPLL, nó cũng có nghĩa là các mệnh đề trong đó tất cả các đối số ngoại trừ một trong số chúng đã được gán FALSE bởi mô hình.





Hình 2. Mã giả thuật giải DPLL

### III. TIÊU CHÍ ĐÁNH GIÁ

Tiêu chí	Điểm tự đánh giá
Read input data and store in suitable data structure (0.5)	0.5
Implementation of resolution method (1.0)	1.0
Inference process and results (2.5)	2.5
Testcases, report, evaluations (1.0)	0.5

### IV. TÀI LIỆU THAM KHẢO

[CS270 Combinatorial Algorithms & Data Structures Spring 2006 - Lecture 1:](#)

[1.17.06 - Lecturer: Satish - Scribe: Jason Wolfe](#)

Artificial Intelligence A Modern Approach (4th Edition)

[DPLL algorithm - Wikipedia](#)

[dangcpr report](#)

[EdricJ logic source](#)

----- HẾT -----