

Lưu ý: Sinh viên chú thích đầy đủ Họ tên, Mã số SV, Nhóm vào bài làm và nộp lại trên elciti.

Câu 1 (6 điểm)



Có 7 phiến đá và 6 chú ếch với 2 màu khác nhau được bố trí như hình bên dưới. Làm thế nào để 3 chú ếch vàng ở bờ bên đây có thể sang bờ bên kia và ngược lại 3 chú ếch nâu có thể sang bờ còn lại. Biết rằng ếch có thể di chuyển như sau:

- Nhảy 1 bước: nếu phía trước là phiến đá trống
- Nhảy 2 bước: một chú ếch có thể **nhảy qua đầu ếch khác màu** nếu phía trước còn phiến đá trống

Bài toán được mô tả trong không gian trạng thái với các đặc điểm sau:

1. Trạng thái bài toán: vị trí các số 1, 2 trong danh sách xác định các trạng thái khác nhau của bài toán
2. Các thao tác (operator) để tạo ra trạng thái mới: nhảy sang trái 1 ô, nhảy sang trái 2 ô, nhảy sang phải 1 ô, nhảy sang phải 2 ô của các ô 1, hoặc 2
3. Trạng thái đầu:

1	1	1		2	2	2
---	---	---	--	---	---	---
4. Trạng thái cuối:

2	2	2		1	1	1
---	---	---	--	---	---	---
5. Việc tìm giải pháp cho bài toán là tìm đường đi từ trạng thái bắt đầu đến trạng thái đích.

Để giải quyết bài toán trên, cần khai báo các hằng và thư viện sau:

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  #define VANG 1 //Ech vang
5  #define NAU 2 //Ech nau
6  #define EMPTY 0 //Phiên đá trống
7  #define MAXLENGTH 100 //Su dung cai dat Stack
```

Mỗi trạng thái của bài toán cần lưu trữ vị trí của các chú ếch trong mảng 1 chiều **da[7]** và vị trí phiến đá trống được lưu trong biến **empty_rock** sẽ được khai báo theo cấu trúc State:

```
13 //Khai bao cau truc trang thai
14 typedef struct{
15     int da[7];
16     int empty_rock; //Luu chi so cua phiên đá trống
17 } State;
```

Bài toán đã cho có 4 phép toán: ếch vàng nhảy 1 bước (nhảy sang phải 1 ô), ếch vàng nhảy 2 bước (nhảy sang phải 2 ô), ếch nâu nhảy 1 bước (nhảy sang trái 1 ô), ếch nâu nhảy 2 bước (nhảy sang trái 2 ô).

Trường hợp ếch vàng nhảy 1 bước, cần kiểm tra vị trí bên cạnh trái phiến đá trống có phải là ếch vàng hay không?

```

72 //Ech vang nhảy 1 bước đến phiến đá trống
73 int echVangNhay1Buoc(State cur_state, State *result){
74     if(cur_state.empty_rock > 0 && cur_state.da[cur_state.empty_rock-1] == VANG){
75         *result = cur_state;
76         result->empty_rock = cur_state.empty_rock - 1;
77         result->da[result->empty_rock] = EMPTY;
78         result->da[cur_state.empty_rock] = VANG;
79         return 1;
80     }
81     return 0;
82 }

```

Tương tự, ta có trường hợp ếch vàng nhảy 2 bước:

```

84 //Ech vang nhảy 2 bước (qua đầu ech nâu) đến phiến đá trống
85 int echVangNhay2Buoc(State cur_state, State *result){
86     if(cur_state.empty_rock > 1 && cur_state.da[cur_state.empty_rock-2] == VANG
87        && cur_state.da[cur_state.empty_rock-1] == NAU)
88     {
89         *result = cur_state;
90         result->empty_rock = cur_state.empty_rock - 2;
91         result->da[result->empty_rock] = EMPTY;
92         result->da[cur_state.empty_rock] = VANG;
93         return 1;
94     }
95     return 0;
96 }

```

Hàm call_operator sẽ lần lượt gọi các phép toán theo thứ tự ếch vàng nhảy 1 bước, ếch vàng nhảy 2 bước, ếch nâu nhảy 1 bước, ếch nâu nhảy 2 bước:

```

120 //Goi cac phép toán trên trạng thái
121 int callOperator(State cur_state, State *result, int option){
122     switch(option){
123         case 1: return echVangNhay1Buoc(cur_state, result);
124         case 2: return echVangNhay2Buoc(cur_state, result);
125         case 3: return echNauNhay1Buoc(cur_state, result);
126         case 4: return echNauNhay2Buoc(cur_state, result);
127         default: printf("Error calls operators");
128         return 0;
129     }
130 }

```

Cấu trúc một node được cho như sau:

```

200 typedef struct Node{
201     State state;
202     struct Node* parent;
203     int no_function; //hành động sinh ra trạng thái hiện tại
204 } Node;

```

Chuỗi các hành động lần lượt là:

```

11 const char* action[] = {"Trạng thái đầu", "Ech vang nhảy 1 bước đến phiến đá trống",
12                          "Ech vang nhảy 2 bước đến phiến đá trống",
13                          "Ech nâu nhảy 1 bước đến phiến đá trống",
14                          "Ech nâu nhảy 2 bước đến phiến đá trống"};

```

Câu hỏi: Dựa vào bài thực hành số 1, hãy viết các hàm cần thiết để thực hiện các công việc sau:

- Cài đặt hàm 2 hành động còn lại cho ếch nâu và in các trạng thái con có thể có cho trạng thái đầu của bài toán (trạng thái đầu [1,1,1,0,2,2,2]) (3 điểm)

- b. Sử dụng **giải thuật DFS**, in tổng số trạng thái đã duyệt và chuỗi các hành động cần thực hiện để tìm ra trạng thái cuối [2,2,2,0,1,1,1] (**3 điểm**)

Lưu ý: in ra màn hình kết quả riêng cho mỗi câu 1a và 1b.