Họ và tên: Nguyễn Việt Dũng

MSSV: 20160696

**BÁO CÁO BÀI TẬP TUẦN LÝ THUYẾT TÍNH TOÁN**

**PROJECT 6: EQ\_DFA**

1. **Mô tả chương trình:**
2. **Tính năng chương trình:**

* Chương trình cho phép đọc file .jff biểu diễn Otomat đa định A, B (tên file do người dùng nhập)
* Cho biết liệu ngôn ngữ sinh bởi DFA A và B có trùng nhau hay không.

1. **Cấu trúc chương trình.**

Chương trình được code bằng Java theo mô hình hướng đối tượng, và được kế thừa từ Project 2 đã làm. Cụ thể chương trình được thiết kế như sau:

* Otomat lưu trữ các thuộc tính của một Otomat: tên Otomat (tên file), danh sách các State, trạng thái bắt đầu, danh sách trạng thái kết thúc, alphabet và ***cài đặt phương thức EDFA (cho phép kiểm tra L(A) = rỗng hay không)***
* State lưu trữ các thuộc tính của một trạng thái trong Otomat: id, tên trạng thái, tọa độ, có phải trạng thái bắt đầu hay kết thúc không?, danh sách hàm chuyển trạng thái với State là trạng thái hiện tại.
* Transition lưu trữ các thuộc tính của một hàm chuyển trạng thái: trạng thái hiện tại, trạng thái (hoặc tập trạng thái) tiếp theo, kí tự đầu vào.
* Main chứa hàm main để chạy chương trình
* OtomatTranducer cài đặt các phương thức để đọc, xuất file .jff biểu diễn Otomat, ***phương thức Giao và Hợp 2 DFA* và** ***phương thức minusOtomat (với đầu vào là 1 otomat và đầu ra là đối của otomat đó – lấy phần bù).***
* Phương thức Giao và Hợp 2 DFA có cải tiến so với Project 2 là đã loại bỏ các trạng thái không thể đến được bằng BFS

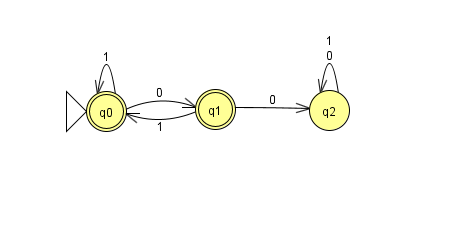
1. **Cài đặt và sử dụng chương trình.**

Chương trình sử dụng các thư viện có sẵn của java nên chỉ cần compile và chạy hàm main là sử dụng được.

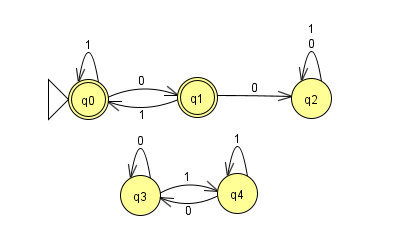
Các file .jff được đọc và xuất ở ...\EQDFA\

**III. Test Case đã thực hiện**

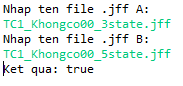
**1. Test Case cho câu trả lời là true:**

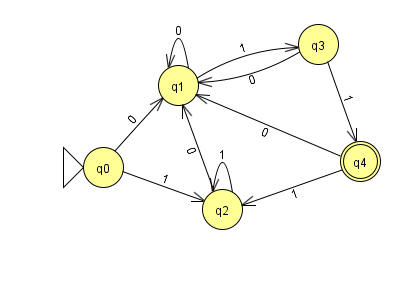
**1.1. DFA và DFA giống nó nhưng có các trạng thái không liên thông với thành phần đoán nhận**

**Hình 1: DFA sinh ngôn ngữ không chứa xâu 00 – TC1\_Khongco00\_3state.jff**

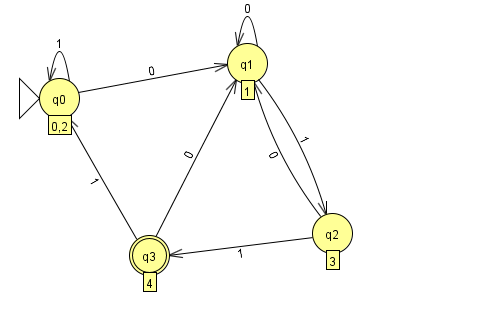


**Hình 2: DFA sinh ngôn ngữ không chứa 00 (có thêm 2 trạng thái nhưng không tham gia vào việc đoán nhận) - TC1\_Khongco00\_5state.jff**

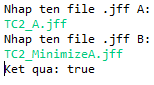


**1.2 DFA và DFA tối thiểu của nó (Minimize)** 

**Hình 3: DFA bất kì (có thể tối thiểu hóa) – TC1\_A.jff**



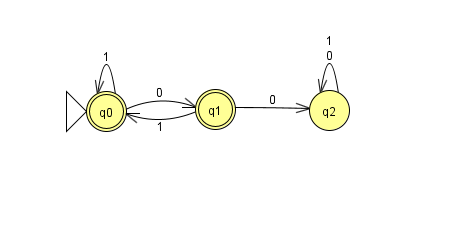
**Hình 4: DFA là tối thiểu hóa của DFA trên (được sinh bằng JFLAP) – TC1\_MinimizeA.jff**



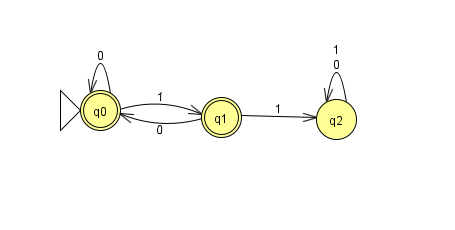
**2. Test Case cho câu trả lời là false:**

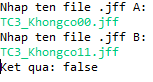
**2.1. 2 DFA bất kì**

**Hình 5: DFA sinh ngôn ngữ không có 00 – TC3\_Khongco00.jff**



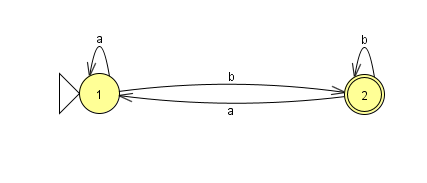
**Hình 6: DFA sinh ngôn ngữ không có 11 – TC3\_Khongco11.jff**





**2.2 2 DFA bất kì**

**Hình 7: DFA bất kì TC4\_A.jff**



**Hình 8: DFA bất kì TC4\_B.jff**

