TRƯỜNG ĐẠI HỌC ĐIỆN LỰC

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO CHUYÊN ĐỀ HỌC PHẦN**

**NHẬP MÔN TRÍ TUỆ NHÂN TẠO**

**ĐỀ TÀI:**

**THUẬT TOÁN DI TRUYỀN CHO BÀI TOÁN SẮP XẾP THỜI KHÓA BIỂU**

**Sinh viên thực hiện : HOÀNG THỊ PHƯƠNG**

**ĐINH THỊ THẢO**

**Giảng viên hướng dẫn: PHẠM THỊ KIM DUNG**

**Khoa : CÔNG NGHỆ THÔNG TIN**

**Chuyên ngành : CÔNG NGHỆ PHẦN MỀM**

**Lớp : D14CNPM3**

**Khóa : 2019 – 2024**

***Hà Nội, ngày 27 tháng 12 năm 2021***

**PHIẾU CHẤM ĐIỂM**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Họ và tên sinh viên** | **Nội dung thực hiện** | **Điểm** | **Chữ ký** |
| 1 | Hoàng Thị Phương  19810310172 |  |  |  |
| 2 | Đinh Thị Thảo  19810310256 |  |  |  |

|  |  |  |
| --- | --- | --- |
| **Họ và tên giảng viên** | **Chữ ký** | **Ghi chú** |
| Giảng viên chấm 1: |  |  |
| Giảng viên chấm 2: |  |  |

# **MỤC LỤC**

[MỤC LỤC 3](#_Toc91365124)

[LỜI MỞ ĐẦU 4](#_Toc91365125)

[CHƯƠNG 1: CƠ SỞ LÝ THUYẾT 5](#_Toc91365126)

[1.1 Giới thiệu về trí tuệ nhân tạo 5](#_Toc91365127)

[1.1.1 Trí tuệ nhân tạo là gì? 5](#_Toc91365128)

[1.1.2 Lĩnh vực của AI 5](#_Toc91365129)

[1.2 Giới thiệu giải thuật di truyền 6](#_Toc91365130)

[2.2 Phân tích thiết kế hệ thống 9](#_Toc91365131)

[2.2.1 Nhiễm sắc thể của bài toán Thời khóa biểu 9](#_Toc91365132)

[2.2.2 Thuật toán sắp xếp thời khóa biểu 10](#_Toc91365133)

[2.2.3 Khởi tạo quần thể 11](#_Toc91365134)

[2.2.4 Thuật toán đánh giá độ thích nghi 12](#_Toc91365135)

[2.2.5 Thuật toán Đột biến, Biến dị, Lai ghép Nhiễm sắc thể 13](#_Toc91365136)

[2.2.6 Thuật toán khử vi phạm số buổi học của một môn học 14](#_Toc91365137)

[2.2.7 Thuật toán khử vi phạm trùng lịch giáo viên 15](#_Toc91365138)

[2.2.8 Thuật toán cân bằng số lượng tiết dạy của giáo viên 17](#_Toc91365139)

[2.2.9 Các bước lập lịch 18](#_Toc91365140)

[2.3 Mô hình dữ liệu quan hệ 19](#_Toc91365141)

[CHƯƠNG 3: XÂY DỰNG CHƯƠNG TRÌNH 20](#_Toc91365142)

[3.1 Trang đăng nhập 20](#_Toc91365143)

[3.2 Trang chính 20](#_Toc91365144)

[3.3 Các trang chức năng trong phần mềm 21](#_Toc91365145)

[3.3.1 Trang giáo viên 21](#_Toc91365146)

[3.3.1 Trang danh sách lớp 21](#_Toc91365147)

[3.3.2 Trang danh sách môn học 22](#_Toc91365148)

[3.3.3 Trang danh sách phòng học 22](#_Toc91365149)

[3.3.4 Trang phân lịch 23](#_Toc91365150)

[3.3.4 Trang thời khóa biểu 23](#_Toc91365151)

[3.3.5 Trang lịch dạy giáo viên 24](#_Toc91365152)

[KẾT LUẬN 25](#_Toc91365153)

**LỜI MỞ ĐẦU**

Trong ngành khoa học máy tính, tìm kiếm lời giải tối ưu cho các bài toán là vấn đề được các nhà khoa học máy tính đặc biệt rất quan tâm. Mục đích chính của các thuật toán tìm kiếm lời giải là tìm ra lời giải tối ưu nhất cho bài toán trong thời gian nhỏ nhất. Các thuật toán nhờ tìm kiếm không có thông tin / vét cạn ( tìm kiếm trên danh sách, trên cây hoặc đồ thị ) sử dụng phương pháp đơn giản nhất và trực quan nhất hoặc các thuật toán tìm kiếm có thông tin sử dụng heurictics để áp dụng các tri thức về cấu trúc của không gian tìm kiếm nhằm giảm thời gian cần thiết cho việc tìm kiếm được sử dụng nhiều nhưng chỉ với không gian tìm kiếm nhỏ và không hiệu quả khi tìm kiếm trong không gian tìm kiếm lớn.

Tuy nhiên, trong thực tiễn có rất nhiều bài toán tối ưu với không gian tìm kiếm rất lớn cần phải giải quyết. Vì vậy, việc đòi hỏi thuật giải chất lượng cao và sử dụng kỹ thuật trí tuệ nhân tạo đặc biệt rất cần thiết khi giải quyết các bài toán có không gian tìm kiếm lớn. Thuật giải di truyền (genetic algorithm) là một trong những kỹ thuật tìm kiếm lời giải tối ưu đã đáp ứng được yêu cầu của nhiều bài toán và ứng dụng.

Thuật giải di truyền đã được phát minh ra để bắt chước quá trình phát triển tự nhiên trong điều kiện quy định sẵn của môi trường. Các đặc điểm của quá trình này đã thu hút sự chú ý của John Holand (ở đại học Michigan) ngay từ những năm 1970. Holand tin rằng sự gắn kết thích hợp trong thuật giải máy tính có thể tạo ra một kỹ thuật giúp giải quyết các vấn đề khó khăn giống như trong tự nhiên đã diễn ra thông qua quá trình tiến hóa.

Trên thế giới hiện nay, Thuật Giải Di Truyền kết hợp với Công nghệ thông tin được ứng dụng để giải quyết những vấn đề phức tạp trong hệ thống điện một cách rất hiệu quả. Vì vậy chúng em chọn đề tài “ Thuật toán di truyền cho bài toán sắp xếp thời khóa biểu” làm báo cáo kết thúc học phần môn học.

# **CHƯƠNG 1: CƠ SỞ LÝ THUYẾT**

## **1.1 Giới thiệu về trí tuệ nhân tạo**

### **1.1.1 Trí tuệ nhân tạo là gì?**

Trí tuệ nhân tạo là trí thông minh của máy do con người tạo ra. Ngay từ khi chiếc máy tính điện tử đầu tiên ra đời, các nhà khoa học máy tính đã hướng đến phát hiển hệ thống máy tính (gồm cả phần cứng và phần mềm) sao cho nó có khả năng thông minh như loài người. Mặc dù cho đến nay, theo quan niệm của người viết, ước mơ này vẫn còn xa mới thành hiện thực, tuy vậy những thành tựu đạt được cũng không hề nhỏ: chúng ta đã làm được các hệ thống (phần mềm chơi cờ vua chạy trên siêu máy tinh GeneBlue) có thể thắng được vua cờ thế giới; chúng ta đã làm được các phần mềm có thể chứng minh được các bài toán hình học; v.v. Hay nói cách khác, trong một số lĩnh vực, máy tính có thể thực hiện tốt hơn hoặc tương đương con người (tất nhiên không phải tất cả các lĩnh vực). Đó chính là các hệ thống thông minh.

Như vậy, trí tuệ nhân tạo ở đây là nói đến khả năng của máy khi thực hiện các công việc mà con người thường phải xử lý; và khi dáng vẻ ứng xử hoặc kết quả thực hiện của máy là tốt hơn hoặc tương đương với con người thì ta gọi đó là máy thông minh hay máy đó có trí thông minh. Hay nói cách khác, đánh giá sự thông minh của máy không phải dựa trên nguyên lý nó thực hiện nhiệm vụ đó có giống cách con người thực hiện hay không mà dựa trên kết quả hoặc dáng vẻ ứng xử bên ngoài của nó có giống với kết quả hoặc dáng vẻ ứng xử của con người hay không.

Trong môn học này, chúng ta sẽ tìm hiểu các phương pháp để làm cho máy tính biết cách giải bài toán, biết cách lập luận, biết cách học, v.v.

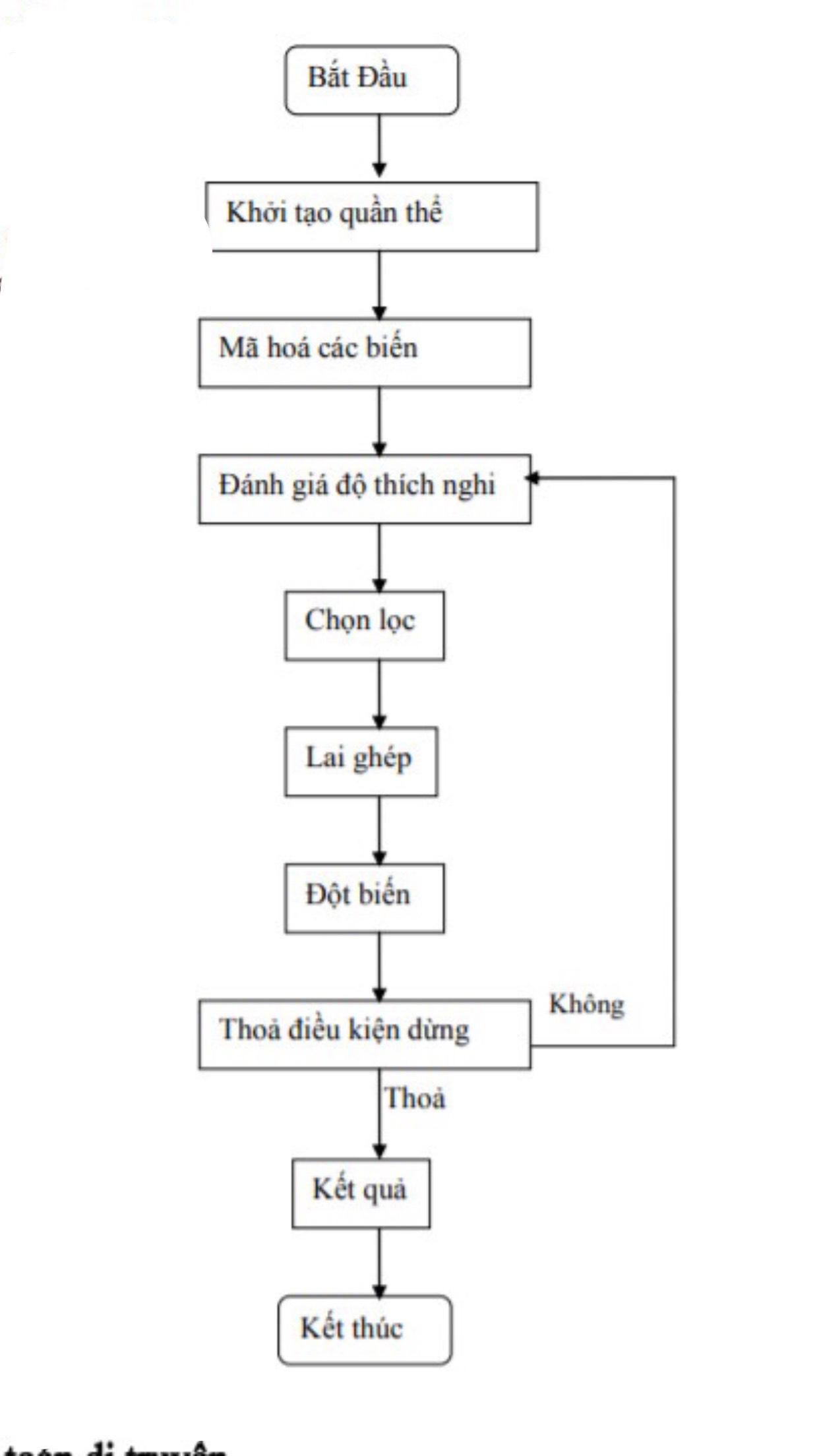
**1.1.2 Lĩnh vực của AI**

* Lập luận, suy diễn tự động: Khái niệm lập luận (reasoning), và suy diễn (reference) được sử dụng rất phổ biến trong lĩnh vực AI. Lập luận là suy diễn logic, dùng để chỉ một tiến trình rút ra kết luận (tri thức mới) từ những giả thiết đã cho (được biểu diễn dưới dạng cơ sở tri thức). Như vậy, để thực hiện lập luận người ta cần có các phương pháp lưu trữ cơ sở tri thức và các thủ tục lập luận trên cơ sở tri thức đó.
* Biểu diễn tri thức: Muốn máy tính có thể lưu trữ và xử lý tri thức thì cần có các phương pháp biểu diễn tri thức. Các phương pháp biểu diễn tri thức ở đây bao gồm các ngôn ngữ biểu diễn và các kỹ thuật xử lý tri thức. Một ngôn ngữ biểu diễn tri thức được đánh giá là “tốt” nếu nó có tính biểu đạt cao và các tính hiệu quả của thuật toán lập luận trên ngôn ngữ đó. Tính biểu đạt của ngôn ngữ thể hiện khả năng biểu diễn một phạm vi rộng lớn các thông tin trong một miền ứng dụng. Tính hiệu quả của các thuật toán lập luận thể hiện chi phí về thời gian và không gian dành cho việc lập luận. Tuy nhiên, hai yếu tố này dường như đối nghịch nhau, tức là nếu ngôn ngữ có tínhbiểu đạt cao thì thuật toán lập luận trên đó sẽ có độ phức tạp lớn (tính hiệu quả thấp)và ngược lại (ngôn ngữ đơn giản, có tính biểu đạt thấp thì thuật toán lập luận trên đó sẽ có hiệu quả cao). Do đó, một thách thức lớn trong lĩnh vực AI là xây dựng các ngôn ngữ biểu diễn tri thức mà có thể cân bằng hai yếu tố này, tức là ngôn ngữ có tínhbiểu đạt đủ tốt (tùy theo từng ứng dụng) và có thể lập luận hiệu quả.
* Lập kế hoạch: khả năng suy ra các mục đích cần đạt được đối với các nhiệm vụ đưa ra, và xác định dãy các hành động cần thực hiện để đạt được mục đích đó.
* Học máy: là một lĩnh vực nghiên cứu của AI đang được phát triển mạnh mẽ và có nhiều ứng dụng trong các lĩnh vực khác nhau như khai phá dữ liệu, khám phá tri thức,…
* Xử lý ngôn ngữ tự nhiên: là một nhánh của AI, tập trung vào các ứng dụng trên ngôn ngữ của con người. Các ứng dụng trong nhận dạng tiếng nói, nhận dạng chữ viết, dịch tự động, tìm kiếm thông tin,…
* Hệ chuyên gia: cung cấp các hệ thống có khả năng suy luận để đưa ra những kết luận. Các hệ chuyên gia có khả năng xử lý lượng thông tin lớn và cung cấp các kết luận dựa trên những thông tin đó. Có rất nhiều hệ chuyên gia nổi tiếng như các hệ chuyên gia y học MYCIN, đoán nhận cấu trúc phân tử từ công thức hóa học DENDRAL, …
* Robotics…

## **1.2 Giới thiệu giải thuật di truyền**

Genetic algorithms (thuật giải di truyền) là một giải thuật mô phỏng theo quá trình chọn lọc tự nhiên, là kỹ thuật chung giúp giải quyết vấn đề bài toán bằng cách mô phỏng sự tiến hóa của con ngƣời hay của sinh vật nói chung (dựa trên thuyết tiến hóa muôn loài của Darwin) trong điều kiện qui định sẵn của môi trƣờng. Lấy ý tƣởng từ quá trình tiến hoá tự nhiên, xuất phát từ một lớp các lời giải tiềm năng ban đầu, GA tiến hành tìm kiếm trên không gian lời giải bằng cách xây dựng lớp lời giải mới tốt hơn (tối ƣu hơn) lời giải cũ. Quá trình xây dựng lớp lời giải mới đƣợc tiến hành dựa trên việc chọn lọc, lai ghép, đột biến từ lớp lời giải ban đầu. Quần thể lời giải trải qua quá trình tiến hoá: ở mỗi thế hệ lại tái sinh các lời giải tƣơng đối tốt, trong khi các lời giải “xấu” thì chết đi.

Trong GA, một tập các biến của bài toán đƣa ra đƣợc mã hóa sang một chuỗi (hay một cấu trúc mã hóa khác) tƣơng tự nhƣ một nhiễm sắc thể trong tự nhiên. Mỗi chuỗi bao gồm một giải pháp có thể của bài toán. Giải thuật di truyền sử dụng các toán tử đƣợc sinh ra bởi sự chọc lọc tự nhiên một quần thể các chuỗi nhị phân (hoặc các cấu trúc khác), mã hóa khoảng tham số trên mỗi thế hệ, khảo sát các phạm vi khác nhau của không gian tham số, và định hƣớng tìm kiếm đối với khoảng mà là xác suất cao để tìm kiếm sự thực hiện tốt hơn. Thuật toán di truyền gồm có bốn quy luật cơ bản là lai ghép, đột biến, sinh sản và chọn lọc tự nhiên



Hình 1.1 Sơ đồ cấu trúc thuật toán di truyền

**CHƯƠNG 2: ỨNG DỤNG GIẢI THUẬT DI TRUYỀN VÀO BÀI TOÁN XẾP LỊCH THỜI KHOÁ BIỂU**

**2.1 Khảo sát thực tế**

**2.1.1 Mô tả đề tài**

Đề tài sẽ thực hiện nghiên cứu các tài liệu thu thập được tử trường THCS, khảo sát thực tế trường học để hiểu được các nội dung trong việc quản lý dữ liệu, phần chia thời khóa biểu. Phân tích khả năng có tiết kiệm được thời gian và cống sức trước và sau khi có đề tài và đề tài có khả năng áp dụng vào thực tế hay không.

Đầu tiên, dữ liệu thô sẽ được cấu trúc hóa để lưu trữ trong hệ quản trị CSDL SQL Server, dữ liệu được thiết kế có những thành phần ngoài khảo sát nhưng sẽ giúp việc quản lý và lập trình dễ dàng hơn.

Sau đó, sẽ thiết kế giao diện để quản lý những danh mục dữ liệu đã thiết kế trước đó. Dữ liệu lưu trữ thời khóa biểu sẽ được thiết kế riêng theo thuật toán của.

Thuật toán sẽ thiết kế sau đó thử nghệm và chỉnh sửa, cuối cùng sẽ có một thuật toán được đưa vào sử dụng. Thuật toán sẽ lấy dữ liệu từ danh mục dữ liệu trước đó, thực thi theo các bước để tạo thành một thời khóa biểu học và dạy ban đầu. Người sử dụng có được thời khóa biểu sẽ có thể sửa lại thủ công theo ý muôn của mình mà không phụ thuộc vào thuật toán. Lúc này khả năng chồng chéo về lịch dạy và học có thể xảy ra.

Để giải quyết vấn đề trùng lịch học hoặc dạy, nhóm sẽ thiết kế một thuật toán khác chỉ nhằm một mục đích là cảnh báo những chồng chéo đang xảy ra cho người sử dụng. Sau khi người sử dụng.

#### **2.1.2 Bài toán đặt ra**

Cho phép người sử dụng cài đặt các yêu cầu về thời khóa biểu sẽ được phân chia. Giao diện thân thiện dễ sử dụng.

Cho phép người sử dụng có thể thêm lớp học, thêm phòng, thêm giáo viên và sử đổi bổ sung một cách nhanh chóng và tiện lợi.

Hệ thống sẽ tự động thiết lập thời khóa biểu cho học sinh và thời khóa biểu cho giáo viên một cách nhanh chóng hiệu quả.

## **2.2 Phân tích thiết kế hệ thống**

### **2.2.1 Nhiễm sắc thể của bài toán Thời khóa biểu**

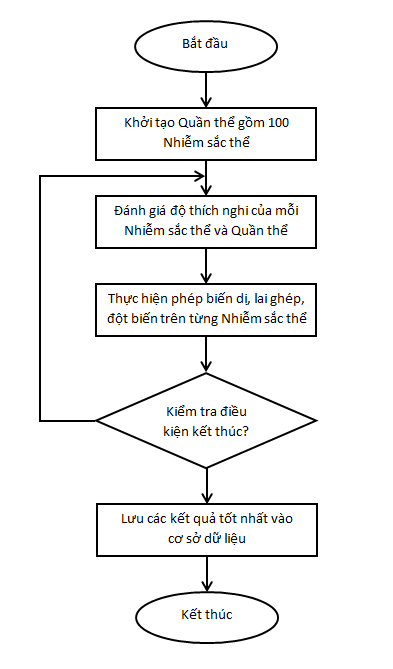
Tùy vào từng bài toán mà người ta sử dụng cấu trúc của nhiễm sắc thể khác nhau, mỗi nhiễm sắc thể được tạo ra với mục đích phù hợp với bài toán, dễ dàng chuyển về dạng thực tế sau khi tìm được lời giải đủ tốt. Nhiễm sắc thể trong bài toán này sử dụng cấu trúc mảng 3 chiều.

* Chiều thứ nhất biểu diễn các Thứ trong tuần, từ thứ 2 đến thứ 7
* Chiều thứ hai biểu diễn các Tiết từ tiết 1 đến 10
* Chiều thứ ba biểu diễn lớp và phòng
* Mỗi phần tử có cấu trúc như sau:
* Lớp
* Thứ (2 - 7)
* Tiết (1 - 10)
* Môn Học
* Giáo Viên

Ngoài ra, nhiễm sắc thể còn mang theo những danh sách dữ liệu như giáo viên, lớp học, môn học, phòng học và các thông tin đính kèm để phục vụ cho việc tiến hóa như:

* Tổng số tiết đã được phân môn học
* Tiết đã được phân giáo viên
* Tổng số tiết đã và chưa được phân giáo viên
* Tổng số tiết không vi phạm điều kiện
* Tổng số tiết vi phạm ít nhất 1 điều kiện
* Và các cấu trúc phục vụ cho việc lập trình

### **2.2.2 Thuật toán sắp xếp thời khóa biểu**

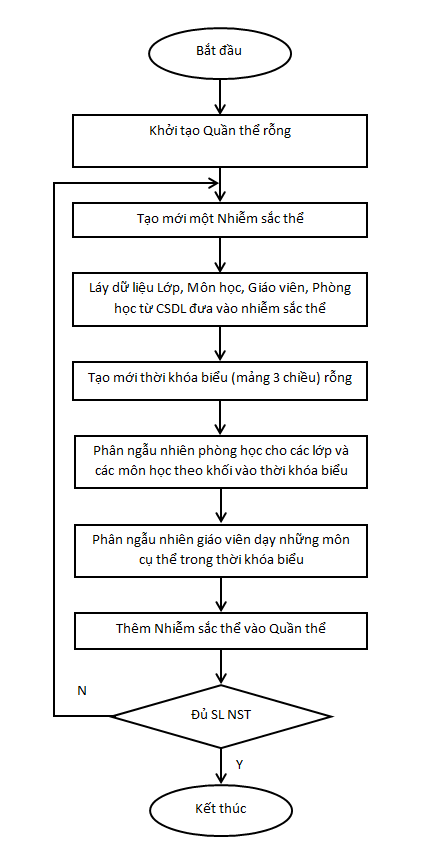


Hình 2.1 Thuật toán phân thời khóa biểu

Thuật toán sắp xếp thời khóa biểu trên mô phỏng tổng thể cách sắp xếp thời khóa biểu dựa vào giải thuật di truyền, những thuật toán con trong thuật toán trên quyết định sự thành công của thuật toán cải tiến mới, dưới đây là những mô hình của các thuật toán con.

### **2.2.3 Khởi tạo quần thể**

Quần thể được tạo thành bằng cách thêm các nhiễm sắc thể sau khi đã khởi tạo các giá trị ban đầu và đã được phân lịch ngẫu nhiên các phòng học, môn học và các giáo viên dạy môn học đó.

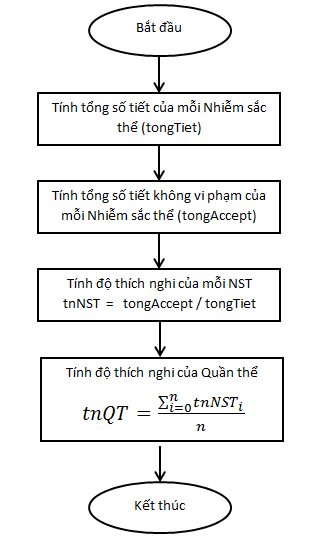


Hình 2.2 Thuật toán tạo quần thể

### **2.2.4 Thuật toán đánh giá độ thích nghi**

Độ thích nghi của quần thể và cá thể được đánh giá sau khi đã có một thời khóa biểu cụ thể ở phần khởi tạo. Độ thích nghi được tính toán dựa vào số lượng **tiết học** không vi phạm các điều kiện của thời khóa biểu.

|  |  |  |  |
| --- | --- | --- | --- |
| **Số tiết học của mỗi khối** | | | |
| **Khối** | **Tiết chính** | **Tiết phụ** | **Tổng số tiết** |
| **6** | 25 | 4 | 29 |
| **7** | 25 | 4 | 29 |
| **8** | 25 | 4 | 29 |
| **9** | 26 | 4 | 30 |
| **Tổng:** | | | 117 |



Hình 2.3 Thuật toán tính độ thích nghi của Quần thể và NST

### **2.2.5 Thuật toán Đột biến, Biến dị, Lai ghép Nhiễm sắc thể**

Các nhiễm sắc thể sẽ tiến hóa sau mỗi lần biến dị hay đột biến hoặc một cặp nhiễm sắc thể bố mẹ kết hợp tạo thành một nhiễm sắc thể con, việc lập trình đột biến, biến dị hay lai ghép nhiễm sắc thể được gọi thành lập trình di truyền, việc lập trình nhằm tới một kết quả là tạo thành một thế hệ mới của quần thể có độ thích nghi cao hơn.

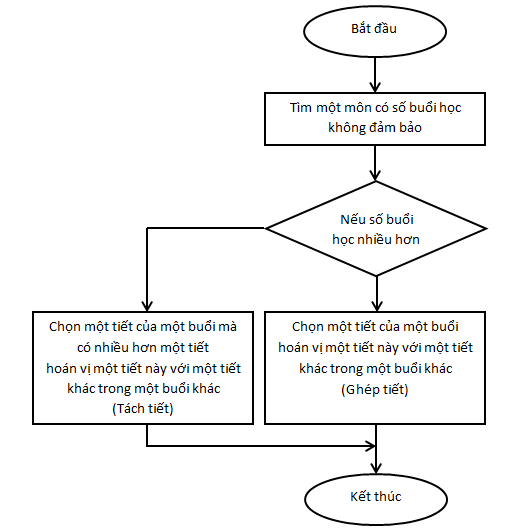
Thuật toán Đột biến, Biến dị: Thuật toán này được lập trình thành nhiều thuật toán con dựa vào các ràng buộc của bài toán thực tế, trong bài toán sắp xếp thời khóa biểu này, các thuật toán sẽ chọn những nhiễm sắc thể vi phạm một trong số các ràng buộc, sau đó thực hiện khử vi phạm để tạo thành nhiễm sắc thể mới có độ thích nghi cao hơn. Trong một số trường hợp nhiễm sắc thể mới có độ thích nghi thấp hơn sau khi đột biến.

Các ràng buộc trong bài toán:

* Ràng buộc về phòng học:
  + Sức chứa của phòng phải lớn hơn hoặc bằng sỉ số học sinh trong lớp
* Ràng buộc về lớp học:
  + Phải học đủ số tiết đã quy định trong mỗi khối
  + Không quá một giáo viên dạy trong một tiết
  + 2 lớp không trung phòng trong 1 tiết học
  + Các tiết học phải liên tục
* Ràng buộc môn học
  + Các môn cố định: Chào cờ, Sinh hoạt, Nghỉ phải đảm bảo đúng vị trí
  + Môn thể dục phải học trái buổi
  + Các môn kép nếu học cùng một buổi thì phải học liên tục
  + Học đủ số lượng tiết và buổi trong một tuần
* Ràng buộc giáo viên
  + Không dạy quá số lượng tiết quy định
  + Chỉ dạy 1 lớp tại một thời điểm
* Ràng buộc mềm khác
  + Đàm bảo thời gian di chuyển của giáo viên
  + Tránh các tiết mà một giáo viên thường đi họp hoặc làm các công việc chuyên môn khác
  + Hạn chế các tiết trống giữa các tiết cho giáo viên

### **2.2.6 Thuật toán khử vi phạm số buổi học của một môn học**

Dựa vào các ràng buộc trên, các thuật toán tiến hóa được xây dựng để khử các vi phạm nếu xảy ra trong thời khóa biểu. Một môn học được quy định phải học đúng số tiết và số buổi, nếu số buổi học không đảm bảo, thuật toán dưới sẽ khử việc này.



Hình 2.4 Thuật toán khử vi phạm số buổi NST

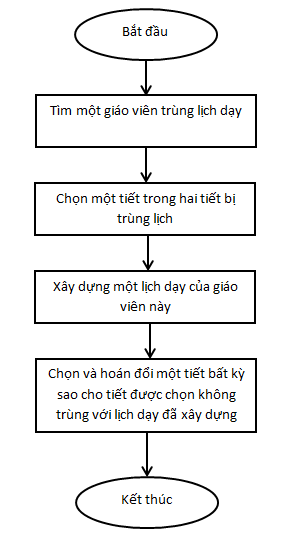
VD: Thời khóa biểu của một lớp, thứ 3 có 3 tiết văn. Tiến hành tách một tiết văn ra khỏi thứ 3 bằng cách hoán vị với tiết Anh văn ngày thứ 6.



### **2.2.7 Thuật toán khử vi phạm trùng lịch giáo viên**

Mỗi giáo viên khi phân dạy một môn cho vài lớp học sẽ có khả năng trùng một vài buổi học nào đó, việc khử vi phạm trùng lịch dạy có nhiều cách để thực hiện, sau đây là một cách để khử vi phạm này:

* Tìm giáo viên trùng lịch dạy
* Chọn 1 trong 2 tiết bị trùng, xây dựng lịch dạy của giáo viên này
* Chọn một tiết khác của môn khác không trùng lịch dạy trên
* Hoán đổi 2 tiết đã chọn ở bước trên



Hình 2.5 Thuật toán khử vi phạm trùng lịch giáo viên



Hình 2.6: Thời khóa biểu trùng lịch dạy môn Toán

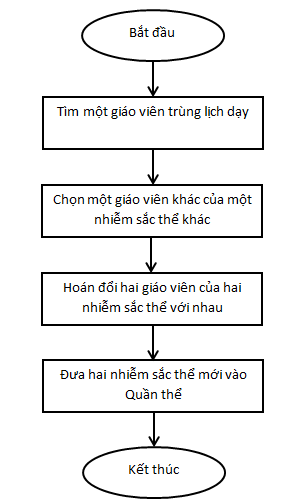
Ở ví dụ trên, một giáo viên dạy môn toán cho cả hai lớp 6A1 và 6A2, và ở tiết 2 ngày thứ 2 đã xảy ra trùng lịch dạy. Thực hiện như thuật toán trên ta có kết quả ở hình 2-15.



Hình 2.7: Thời khóa biểu sau khi hoán đổi hai môn Toán <-> Ngoại ngữ

Một thuật toán khác là sử dụng phép lai tạo nhiễm sắc thể, trong trường hợp này, nhiễm sắc thể được chọn để lai tạo dựa vào đoạn gene (giáo viên dạy môn cùng môn này). Sau khi chọn đoạn gene phù hợp, thuật toán sẽ hoán vị hai đoạn gene (giáo viên) của hai nhiễm sắc thể này để tạo thành hai nhiễm sắc thể mới (Chi, 2012).

Thuật toán này không đảm bảo sẽ tạo thành một nhiễm sắc thể tốt vì không đảm bảo việc lịch dạy của giáo viên có bị trùng hay không. Thuật toán được cải tiến bằng cách chọn những giáo viên có lịch dạy không trùng để thay thế, nhưng việc so sánh thời khóa biểu để tìm kiếm giáo viên phù hợp tốn nhiều chi phí xử lý nhưng kết quả có thể không cao, do vậy thuật toán này ít được áp dụng vào lập trình bài toán.



Hình 2.8 Thuật toán khử vi phạm trùng lịch dạy của giáo viên

### **2.2.8 Thuật toán cân bằng số lượng tiết dạy của giáo viên**

Số lượng tiết dạy tối đa của một giáo viên được quy định cụ thể cho từng giáo viên một, việc quy định này tùy thuộc vào tình hình thực tế của mỗi trường và của tường giáo viên. Tuy nhiên trong lúc phân lịch ngẫu nhiên sẽ khó tránh khỏi việc một giáo viên được phân công dạy quá ít hoặc quá nhiều. Thuật toán sau thực hiện biến dị trên một nhiễm sắc thể nhằm điều chỉnh lại số lượng tiết dạy của tất cả giáo viên về trạng thái cân bằng.

* Trước tiên, Tính tổng số tiết mà giáo viên đã dạy (tongDaDay)
* Tinh tổng số tiết giáo viên chưa dạy (tongChuaDay = tongDuocDay - tongDaDay)
* Sắp xếp giáo viên theo chiều giảm dần của tongChuaDay
* Tính trung bình cộng của tổng số tiết chưa dạy = tongTB

Chuyển tiết dạy của những giáo viên có tongChuaDay nhỏ hơn tongTB đến những

giáo viên có tongChuaDay lớn hơn tongTB

### **2.2.9 Các bước lập lịch**

Tạo mới Nhiễm sắc thể (NST)

Đọc dữ liệu và đưa vào NST

Phân ngẫu nhiên lịch môn học vào lớp

Phân ngẫu nhiên giáo viên dạy

Tiếp tục tiến hóa?

Khử vi phạm của toàn bộ tiết học

Kiểm tra sự vi phạm của toàn bộ tiết học trong tất cả các lớp học

Tính độ thích nghi của NST

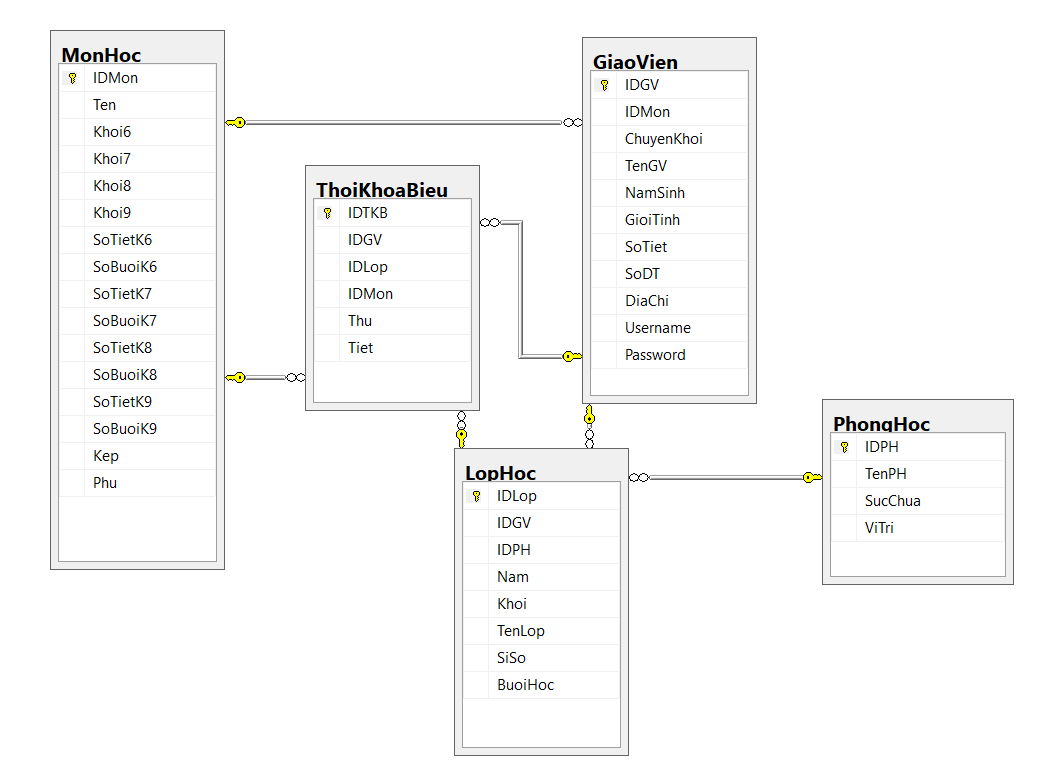
Hiển thị kết quả thời khóa biểu

Y

N

Hình 2.9 Thuật toán lập lịch

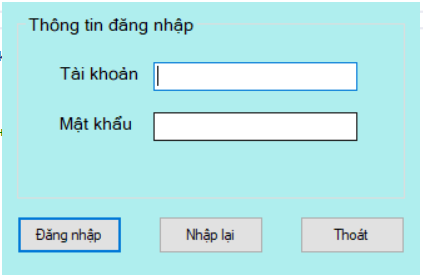
### **2.3 Mô hình dữ liệu quan hệ**



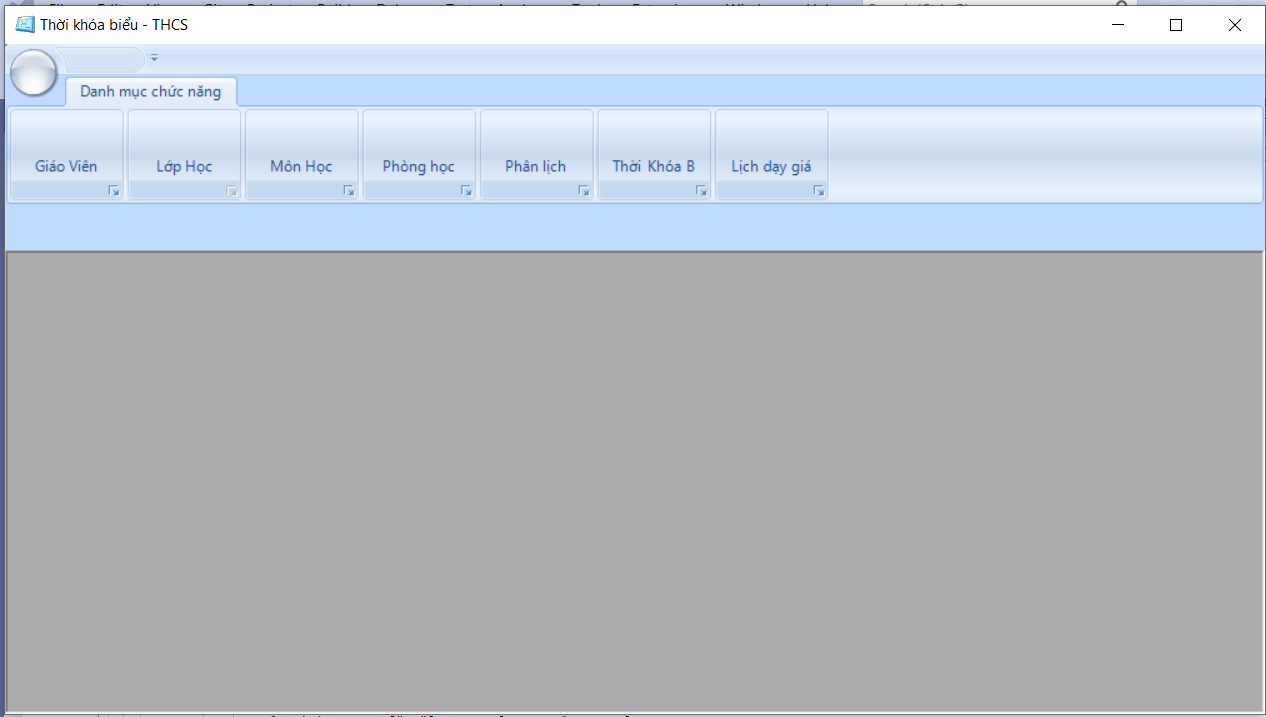
Hình 2.10 Mô hình dữ liệu quan hệ

# **CHƯƠNG 3: XÂY DỰNG CHƯƠNG TRÌNH**

## **3.1 Trang đăng nhập**

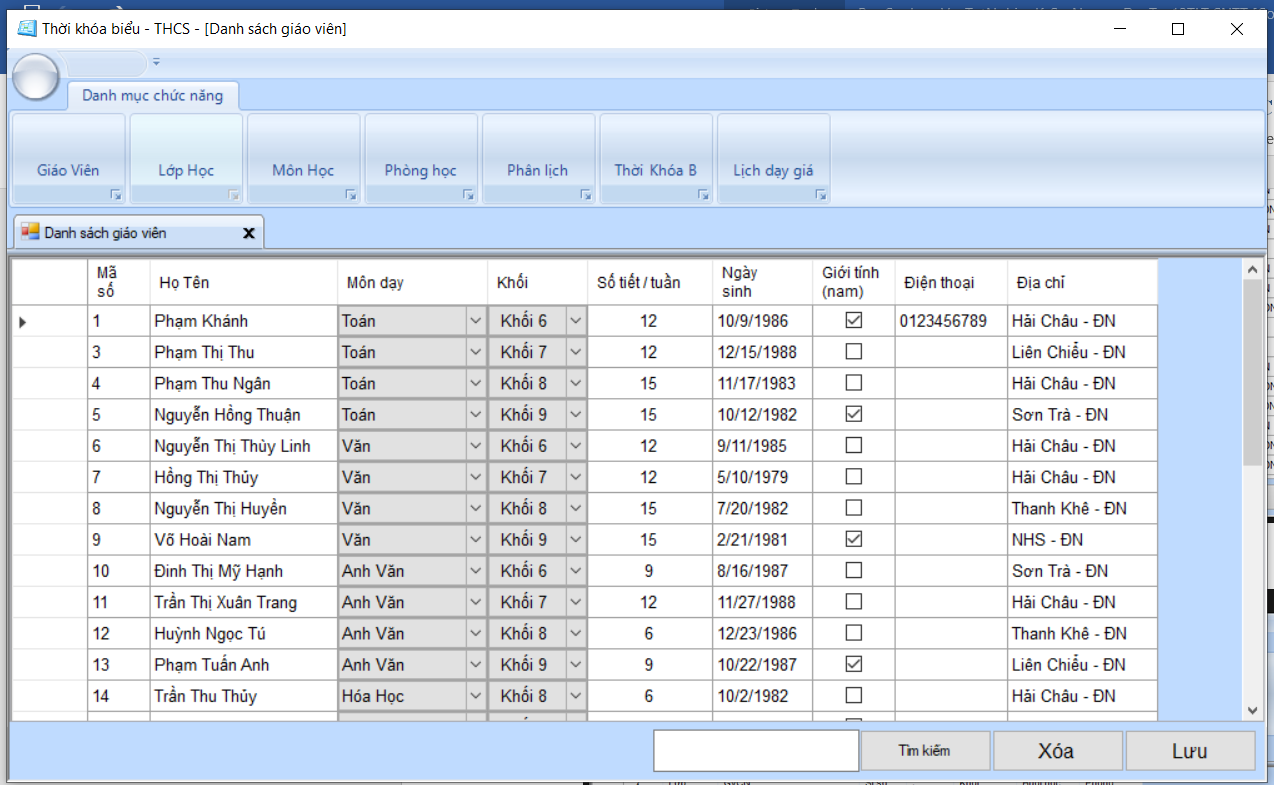
****

## **3.2 Trang chính**

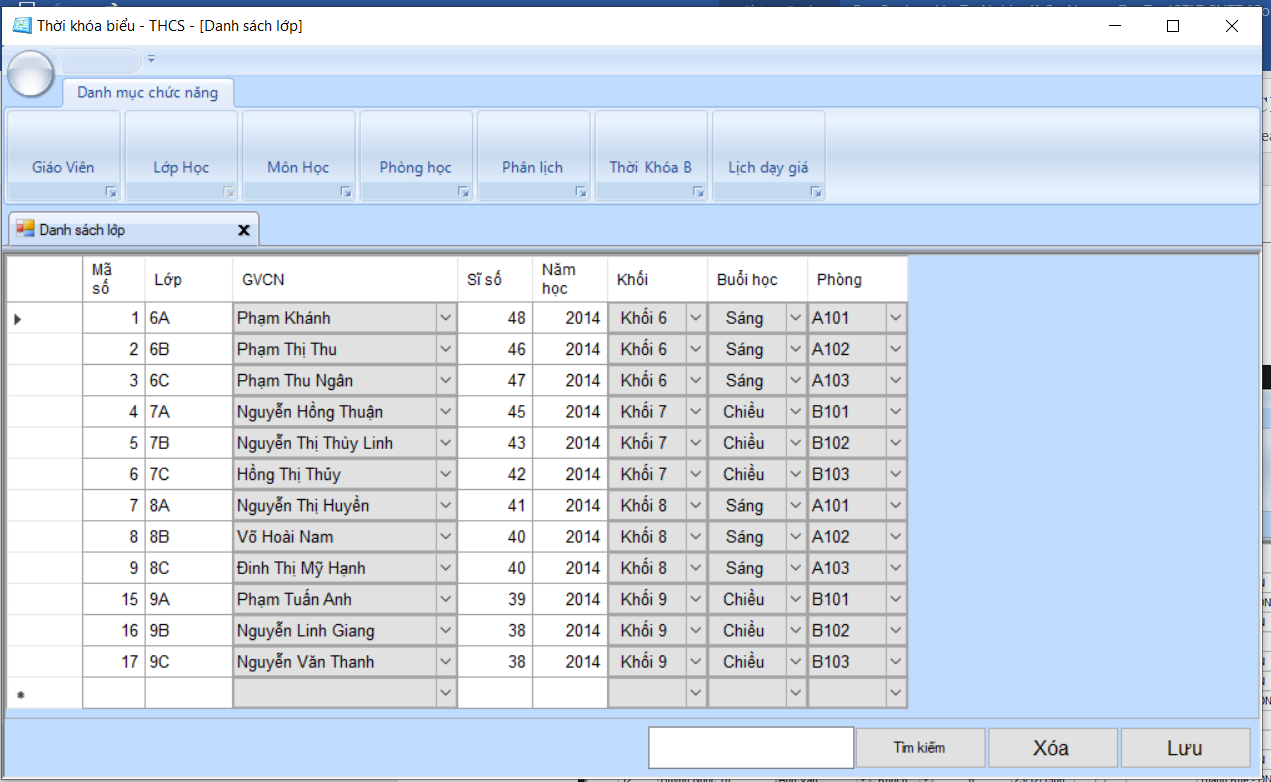
****

## **3.3 Các trang chức năng trong phần mềm**

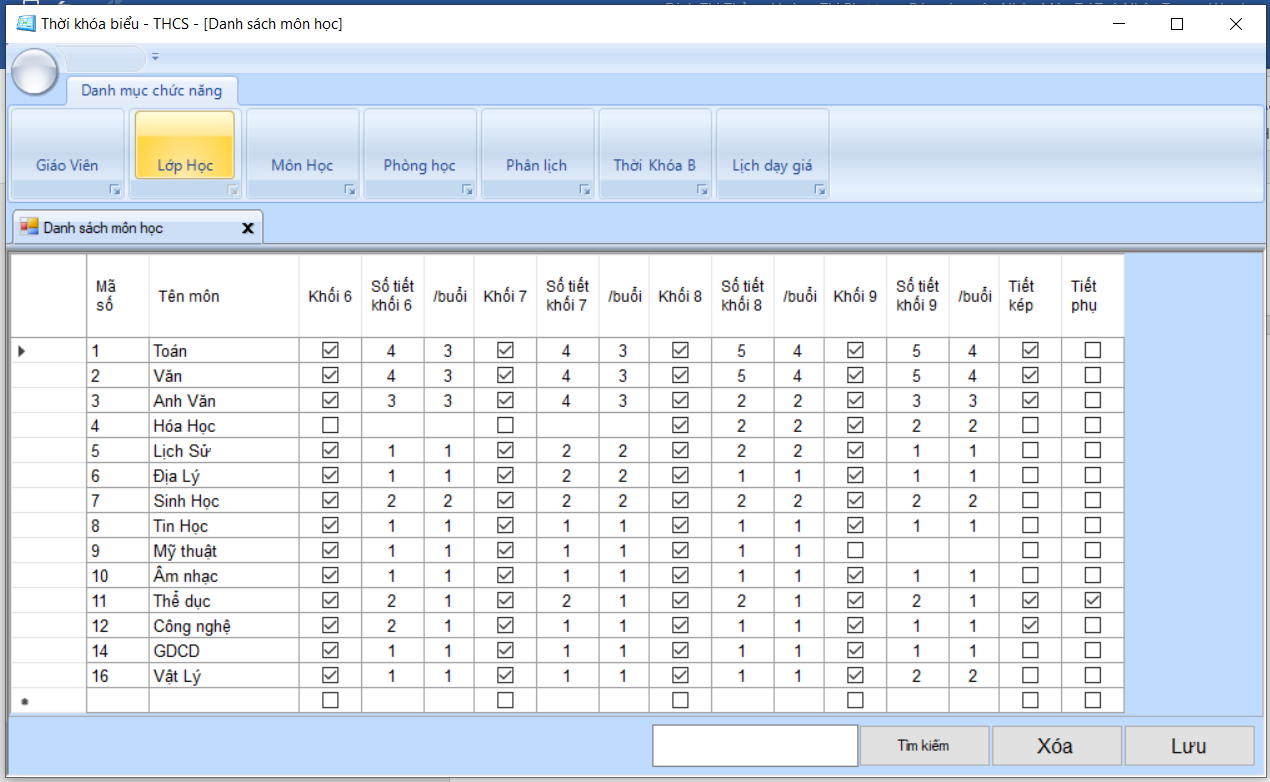
### **3.3.1 Trang giáo viên**

****

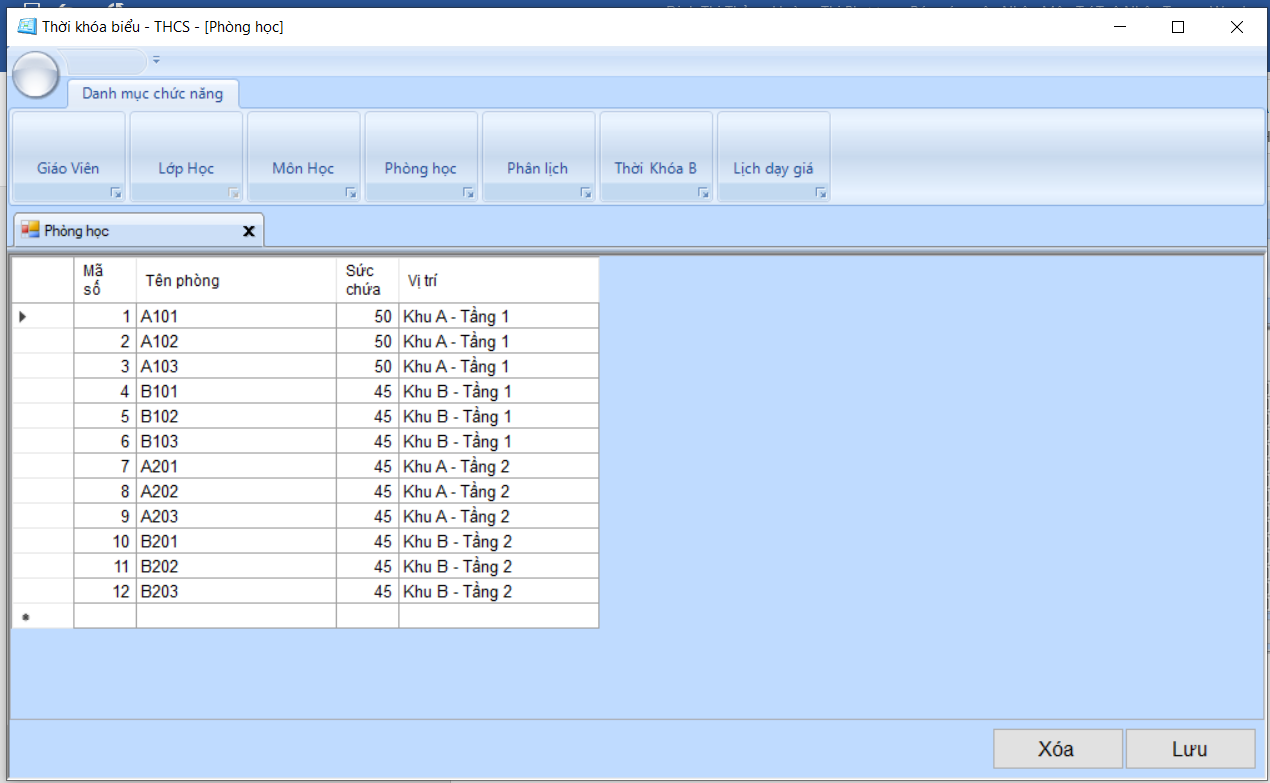
### **3.3.1 Trang danh sách lớp**

****

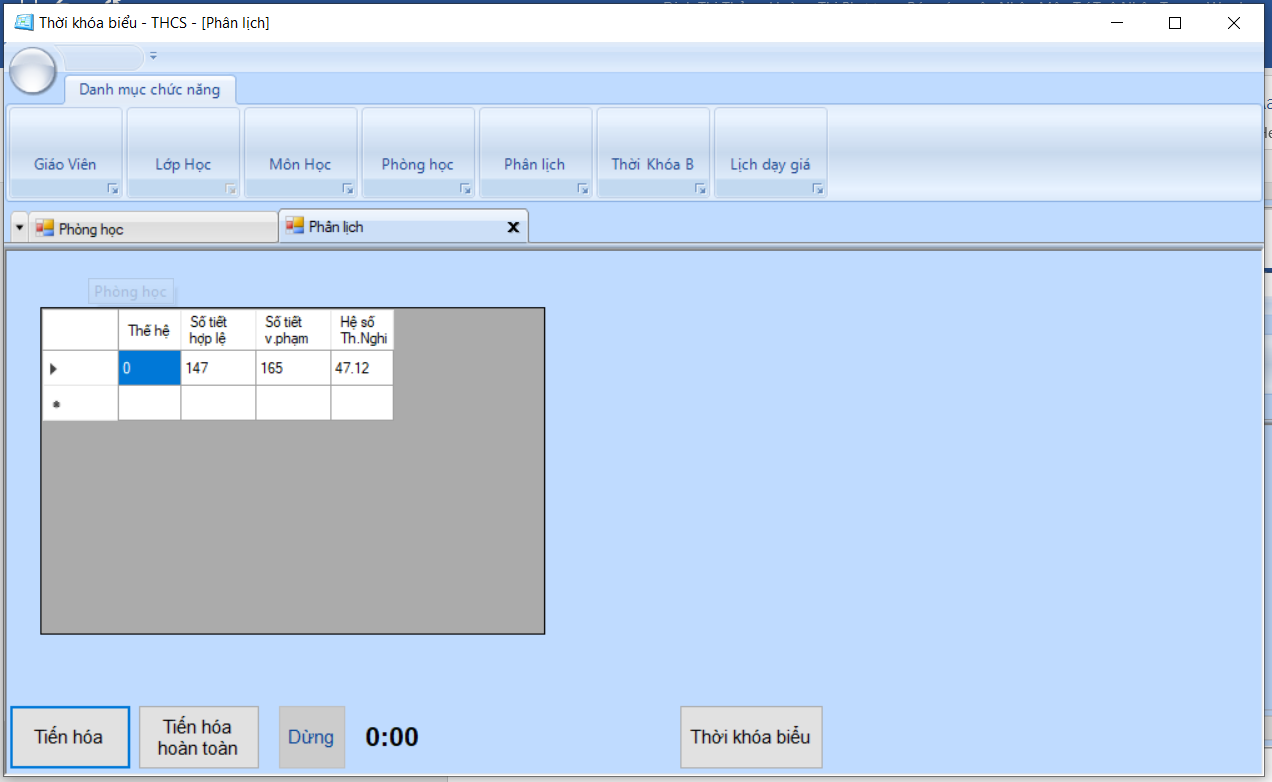
### **3.3.2 Trang danh sách môn học**

****

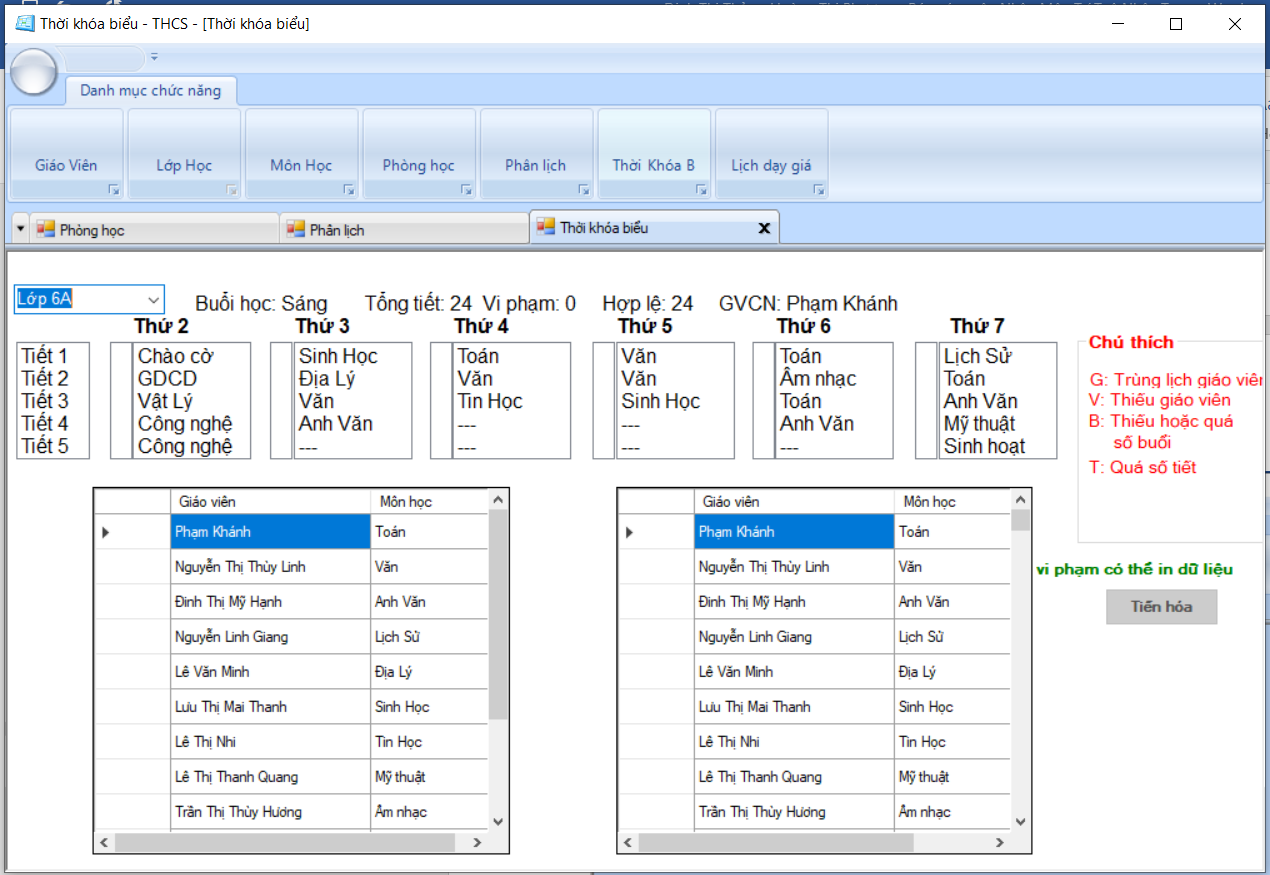
### **3.3.3 Trang danh sách phòng học**

****

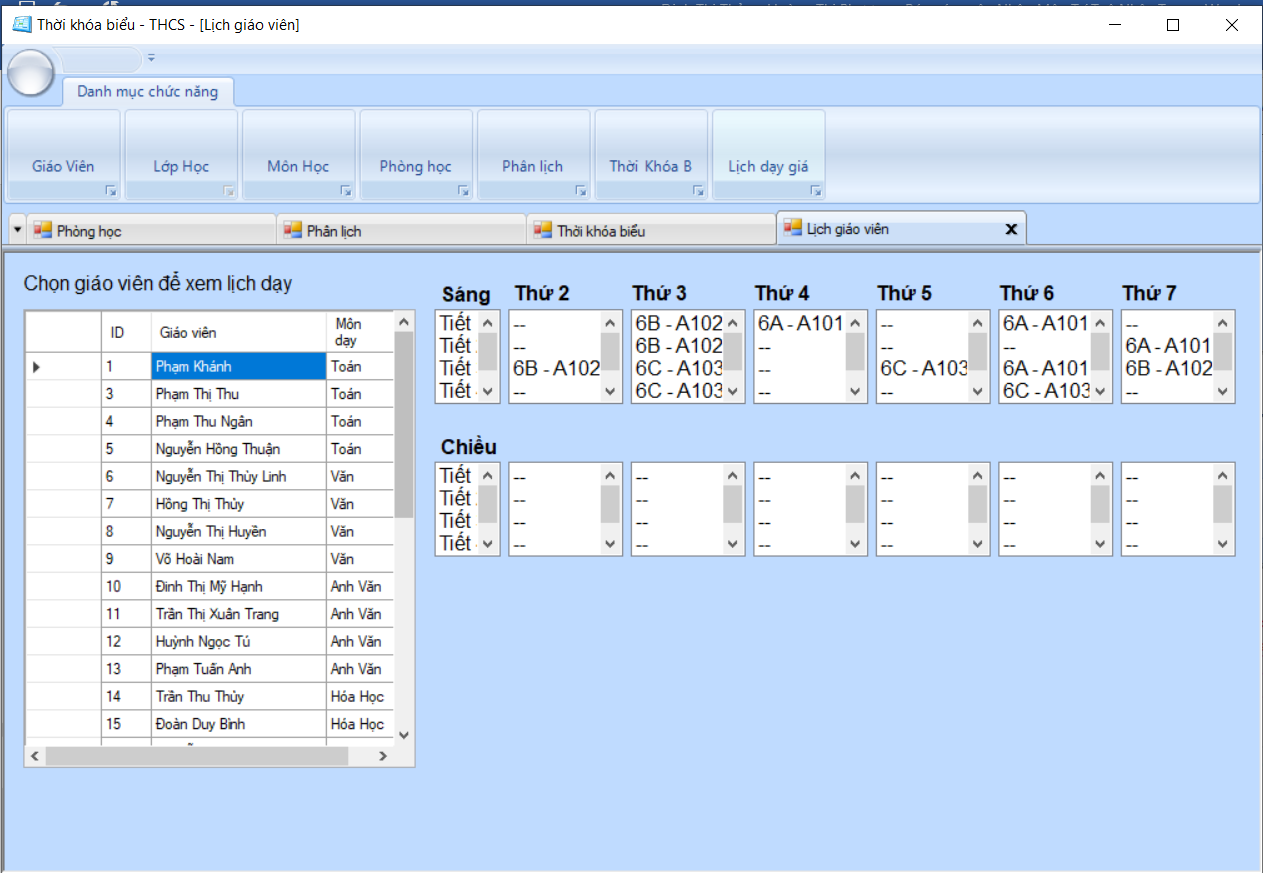
### **3.3.4 Trang phân lịch**

****

### **3.3.4 Trang thời khóa biểu**

****

### **3.3.5 Trang lịch dạy giáo viên**

****

# **KẾT LUẬN**

Phần mềm đang chạy thực nghiệm, bước đầu đạt được những kết quả như:

* + - Truy xuất dữ liệu nhanh chóng phù hợp với quy mô quản lý của chương trình với hệ thống CSDL phù hợp.
    - Hệ thống dữ liệu đảm bảo việc nhập dữ liệu, quản lý, tra cứu, khai thác dữ liệu được nhanh chóng, thuận tiện.
    - Phần mềm giúp người quản lý việc cập nhập thông tin, dữ liệu về: môn học, lớp, giáo viên, phòng.
    - Phần mềm về cơ bản là sắp xếp phân chia được lịch học và lịch dạy của giáo viên.
    - Giảm thiểu thời gian và công sức quản lý, tăng độ chính xác, phục vụ đạt hiệu quả cao.

Giao diện thân thiện, dễ sử dụng.