

**DANH MỤC ĐỀ TÀI**  
**ĐỒ ÁN GIẢI THUẬT VÀ LẬP TRÌNH**  
Năm học 2016-2017

**A. HƯỚNG DẪN THỰC HIỆN**

1. Phân tích bài toán.
2. Lựa chọn cấu trúc dữ liệu.
3. Lựa chọn phương pháp.
4. Triển khai lập trình.
5. Đánh giá và minh họa kết quả.
6. Viết báo cáo theo quy định chung.

**B. DANH SÁCH ĐỀ TÀI**

**Đề tài 1:** Tìm hiểu các thuật toán tìm đường đi và xây dựng chương trình tìm đường đi ngắn nhất trên đồ thị có hướng và có trọng số.

Mô tả:

Dữ liệu vào: Tập chứa ma trận biểu diễn đồ thị  $G$  với  $n$  đỉnh.

Phương pháp gợi ý: Dijkstra, Bellman-Ford.

Dữ liệu ra: Tập chứa khoảng cách ngắn nhất và đường đi (phụ thuộc vào phương pháp lựa chọn).

**Đề tài 2:** Tìm hiểu phương pháp tìm luồng cực đại trên mạng và xây dựng ứng dụng.

Mô tả:

Dữ liệu vào: Tập chứa ma trận biểu diễn mạng  $G$  với đỉnh phát  $s$  và đỉnh thu  $t$ .

Phương pháp gợi ý: Ford-Fulkerson.

Dữ liệu ra: Tập chứa luồng cực đại và giá trị của luồng cực đại.

**Đề tài 3:** Tìm hiểu các phương pháp giải bài toán quy hoạch tuyến tính và xây dựng ứng dụng.

Mô tả:

Dữ liệu vào: Tập chứa ma trận biểu diễn bài toán tối ưu tuyến tính.

Phương pháp gợi ý: Simple Method, Two phase Simple Method, M-method.

Dữ liệu ra: Tập chứa tập giá trị các biến đạt giá trị tối ưu, giá trị tối ưu của hàm mục tiêu.

**Đề tài 4:** Tìm hiểu các kỹ thuật phân rã và xây dựng chương trình dự đoán kết quả đánh giá của người dùng.

Mô tả:

Dữ liệu vào:  $n$ ,  $m$  và ma trận đánh giá  $X$  (ma trận  $X$  có những vị trí chưa được đánh giá)

Phương pháp gợi ý: Singular Value Decomposition, matrix factorization.

Dữ liệu ra: Tập chứa ma trận  $Y$  sao cho  $Y \cong X$ .

**Đề tài 5:** Tìm hiểu lọc cộng tác và xây dựng chương trình dự đoán kết quả đánh giá của người dùng.

Mô tả:

Dữ liệu vào:  $n$ ,  $m$  và ma trận đánh giá  $X$  (ma trận  $X$  có những vị trí chưa được đánh giá)

Phương pháp gợi ý: Item-item Collaborative Filtering, User-user Collaborative Filtering.

Dữ liệu ra: Tập chứa ma trận  $Y$  sao cho  $Y \cong X$ .

**Đề tài 6:** Tìm hiểu các thuật toán tìm cây khung nhỏ nhất trên đồ thị vô hướng và có trọng số.

Mô tả:

Dữ liệu vào: Tập chứa ma trận trọng số biểu diễn đồ thị  $G$  gồm  $n$  đỉnh.

Phương pháp gợi ý: Kruskal, Prim.

Dữ liệu ra: Tập chứa ma trận biểu diễn cây khung nhỏ nhất cần tìm.

**Đề tài 7:** Tìm hiểu phương pháp quay lui và xây dựng ứng dụng.

Mô tả: Tìm hiểu phương pháp và ứng dụng vào các bài toán:  $n$ -hậu hòa bình, mã đi tuần, liệt kê các cấu hình tổ hợp.

Dữ liệu vào: Tập chứa dữ liệu nhập của bài toán tương ứng.

Phương pháp gợi ý: Phương pháp Backtracking.

Dữ liệu ra: Tập chứa dữ liệu xuất của bài toán tương ứng.

**Đề tài 8:** Tìm hiểu các phương pháp giải bài toán vận tải và xây dựng ứng dụng.

Mô tả: Tìm hiểu phương pháp và ứng dụng vào các bài toán:  $n$ -hậu hòa bình, mã đi tuần, liệt kê các cấu hình tổ hợp.

Dữ liệu vào: Tập chứa ma trận cước phí và các giá trị của các trạm phát và thu.

Phương pháp gợi ý: Thuật toán thế vị, thuật toán phân phối.

Dữ liệu ra: Tập chứa ma trận phân phối tối ưu.

**Đề tài 9:** Tìm hiểu phương pháp quy hoạch động và xây dựng ứng dụng.

Mô tả: Tìm hiểu phương pháp và ứng dụng vào các bài toán: Bài toán chiếc túi, bài toán tổ hợp, thuật toán Floyd tìm đường đi ngắn nhất.

Dữ liệu vào: Tập chứa dữ liệu nhập của bài toán tương ứng.

Phương pháp gợi ý: Phương pháp Dynamic Programming.

Dữ liệu ra: Tập chứa dữ liệu xuất của bài toán tương ứng.

**Đề tài 10:** Tìm hiểu các thuật toán sắp xếp và đánh giá để so sánh.

Mô tả: Tìm hiểu phương pháp sắp xếp: chọn, chèn, nổi bọt, nhanh, ...

Dữ liệu vào: Mảng gồm  $n$  số.

Phương pháp gợi ý: Các loại thuật toán sắp xếp, đánh giá bằng ký pháp  $O$  lớn.

Dữ liệu ra: Mảng số sau khi sắp xếp.

**Đề tài 11:** Kiểm tra đồ thị liên thông

Mô tả: Viết chương trình kiểm tra một đồ thị vô hướng có liên thông không.

Dữ liệu vào: Tập dữ liệu mô tả đồ thị.

Phương pháp gợi ý: sử dụng thuật toán duyệt đồ thị.

Dữ liệu ra: Trạng thái đồ thị: Liên thông/không liên thông.

**Đề tài 12:** Cài đặt cây nhị phân tìm kiếm

Mô tả: Cho tệp văn bản. Hãy xây dựng cây nhị phân tìm kiếm, có khóa là một từ trong tệp văn bản. Cấu trúc dữ liệu biểu diễn một nút của cây nhị phân có ít nhất hai trường: từ (khóa) và số lần xuất hiện của từ trong tệp. Đọc tệp văn bản để tạo cây nhị phân tìm kiếm, nếu có từ lặp lại thì chỉ tăng số lần xuất hiện của từ. In các từ và số lần xuất hiện mỗi từ.

**Đề tài 13:** Ngựa đi tuần

Mô tả: Viết chương trình thực hiện di chuyển con ngựa trên bàn cờ có  $8 \times 8$  ô. Giả sử đặt con ngựa trên một ô bất kỳ, di chuyển con ngựa trên bàn cờ sao cho đi qua được

nhiều ô nhất, với điều kiện mỗi ô chỉ được đi qua một lần. Yêu cầu di chuyển ba cách sau:

- Di chuyển con ngựa một cách ngẫu nhiên.
- Di chuyển con ngựa đến ô có các khả năng đi tiếp theo ít nhất.
- Di chuyển con ngựa, nếu hết đường đi thì quay lui (sử dụng giải thuật quay lui).

**Đề tài 14:** Định giá biểu thức số học

Mô tả: Viết chương trình định giá biểu thức số học sử dụng cấu trúc dữ liệu ngăn xếp.

Yêu cầu:

a. Chuyển biểu thức trung tố thành dạng hậu tố hoặc tiền tố (không chứa dấu ngoặc).

b. Định giá biểu thức trong hai trường hợp:

1) Biểu thức dạng tiền tố.

2) Biểu thức dạng hậu tố.

**Đề tài 15:** Duyệt đồ thị

Mô tả: Cho tệp văn bản chứa ma trận kề, biểu diễn đồ thị có hướng G. Viết chương trình đọc tệp văn bản và tạo đồ thị. Chương trình cho phép duyệt đồ thị theo chiều rộng và duyệt đồ thị theo chiều sâu. Yêu cầu sử dụng cấu trúc dữ liệu:

a. Ma trận kề để biểu diễn đồ thị.

b. Danh sách kề (danh sách liên kết) để biểu diễn đồ thị.

**Đề tài 16:** Xếp lịch thi

Mô tả: Giả sử mỗi sinh viên phải thi một số môn trong n môn thi. Hãy lập lịch thi sao cho không có sinh viên nào có 2 môn thi cùng 1 thời gian và số đợt thi là ít nhất.

Đầu vào: Các tập tin văn bản chứa thông tin về: Số môn thi, Tên của các môn thi. Số sinh viên, Các môn học của từng sinh viên

Đầu ra: Tập tin văn bản chứa Lịch thi với số đợt thi ít nhất và không có sinh viên nào thi nhiều hơn 1 môn ở mỗi đợt thi.

Gợi ý: Ứng dụng lý thuyết đồ thị và bài toán tô màu

**Đề tài 17:** Quản lý gia phả

Mô tả đề tài : *Cài đặt cây gia phả và lưu vào tập tin, để hỗ trợ cho quá trình tìm kiếm thông tin.*

**Yêu cầu của đề tài :**

- Nắm vững cơ sở lý thuyết về cấu trúc dữ liệu, các giải thuật duyệt cây. Vận dụng vào cài đặt cây gia phả.

- Giao diện thân thiện người dùng.

*Các chức năng chính của chương trình :*

- **Nhập cây gia phả:** của một dòng họ và lưu vào tập tin. Mỗi dòng họ là một tập tin. Chương trình cũng có thể dựng lại cây gia phả của dòng họ từ tập tin đã lưu. Mỗi người trong dòng họ cần lưu các thông tin như: họ tên, ngày sinh, giới tính, nghề nghiệp, ngày mất, họ tên chồng/vợ, số con (nếu là người nữ), v.v...

- **Hiện thị cây gia phả:** của dòng họ lên màn hình để kiểm tra.

Ví dụ: Có thể hiện thị cây gia phả của một dòng họ Nguyễn như sau:

N. Văn Phước

N. Thị Lộc (nút lá, vì là nữ)

N. Văn Thọ

N. Thị Hạnh

N. Văn Phúc

N. Văn Phú

N. Văn Quý  
N. Thị Cát (nút lá, vì là nữ)  
N. Văn Tường

...  
- ***Tìm mối quan hệ***: Cho phép nhập vào họ tên của hai người nào đó, chương trình sẽ tìm xem họ có mối quan hệ thân thuộc gì với nhau không. Nếu có, thì cho biết mối quan hệ đó là gì (cha, mẹ, vợ, chồng, con trai, con gái, ông, bà nội, ông, bà hay cô, bác, v.v...)

Ví dụ: Với cây gia phả minh họa như trên, thì khi nhập vào N. Văn Quý và N. Thị Hạnh thì chương trình sẽ cho biết Quý là cháu trai của Hạnh, gọi Hạnh là cô Hai.

- ***Tìm người theo mối quan hệ***: Cho phép nhập vào họ tên X của một người nào đó, và mối quan hệ cần tìm. Chương trình sẽ hiển thị tên của người thân của X với mối quan hệ đã cho.

Ví dụ: Với cây gia phả minh họa như trên, thì khi nhập vào tên N. Văn Quý và mối quan hệ 'bác', thì chương trình sẽ hiển thị tên của các bác của Quý: N. Văn Phúc.

**MÔI TRƯỜNG CÀI ĐẶT** : C, C ++

**Đề tài 18**: Giải sudoku 9x9

Yêu cầu đặt ra là viết chương trình giải sudoku 9x9 (sử dụng thuật toán quay lui)

1. Chương trình đọc dữ liệu từ file INPUT.TXT

2. Sau khi đọc file thì tiến hành xử lý

3. Ghi lời giải cho ô chữ ra file OUTPUT.TXT

Các bạn giúp mình với nhé, cảm ơn các bạn rất nhiều...!!!!

**File INPUT.TXT:**

```
8 0 6 0 4 9 2 0 5
0 9 0 0 0 0 3 0 4
4 0 0 2 3 0 0 0 0
1 0 0 4 0 3 0 5 9
6 0 5 0 0 0 4 0 8
2 4 0 8 0 5 0 0 7
0 0 0 0 5 8 0 0 2
9 0 8 7 2 0 5 0 1
```

**File OUTPUT.TXT**

```
8 3 6 1 4 9 2 7 5
7 9 2 5 8 6 3 1 4
4 5 1 2 3 7 8 9 6
1 8 3 4 7 2 6 5 9
6 7 5 3 9 1 4 2 8
2 4 9 8 6 5 1 3 7
3 1 4 9 5 8 7 6 2
5 2 7 6 1 4 9 8 3
9 6 8 7 2 3 5 4 1
```

## Đề tài 19: Quản lý thanh toán điện thoại

Mô tả đề tài: Công ty điện thoại muốn có một chương trình thanh toán tiền điện thoại cho các khách hàng của mình. Hiện tại, công ty có lưu trữ 2 tệp tin, một tệp về khách hàng, một tệp về các cuộc điện thoại đã gọi.

1. Tệp khách hàng mang tên **customer.dat**. Mỗi dòng trong tệp này gồm có:

- Tên của khách hàng: tên là một chuỗi ký tự, độ dài không vượt quá 25 và kết thúc bởi dấu “;”.
- Số điện thoại của khách hàng: số điện thoại nằm trong khoảng 80000000 và 89999999. Chẳng hạn dòng đầu tiên của tệp là:  
Trần Thị Thanh; 8271954

2. Tệp lưu trữ các cuộc điện thoại mang tên **phone.dat**. Mỗi dòng có những thông tin về một cuộc điện thoại như sau:

- Số điện thoại: trong khoảng 80000000 đến 89999999.
- Số phút đã gọi: trong khoảng 1 đến 120.
- Thời điểm bắt đầu gọi: từ 0h00 đến 23h59 (đồng hồ 24 giờ/ngày).
- Ngày gọi: theo định dạng dd/mm/yyyy.
- Miền: nội hạt, lân cận, xa và rất xa, được viết tắt là: NH, LC, X, RX.

Chẳng hạn, dòng đầu tiên của tệp phone.dat có thể như sau:

8567345 4 8.23 14/06/2006 NH

Đương nhiên là có thể có nhiều dòng điện thoại.

Hai tệp trên được liên hệ với nhau qua số điện thoại. Sau khi đọc một cuộc điện thoại trong tệp **phone.dat**, cuộc gọi sẽ được xử lý cho khách hàng có số điện thoại trùng với số điện thoại của cuộc điện thoại đó. Tệp các cuộc gọi ban đầu chưa được sắp xếp nhưng tệp kết quả phải được sắp xếp theo sự tăng dần của số điện thoại. Có thể giả sử rằng tất cả các dữ liệu trên tệp đều hợp lệ và đương nhiên có thể có những khách hàng không có cuộc nào.

3. Tệp mang tên **result.dat** cần có tiêu đề thích hợp và nêu rõ những thông tin sau:

- Với mỗi khách hàng:
  - + Tổng số tiền mà họ phải trả công ty.
  - + Số cuộc gọi của từng miền.
- Thông tin tổng thể:
  - + Toàn bộ số tiền thu được từ khách hàng.
  - + toàn bộ các cuộc gọi của khách hàng trong từng miền.

### Xử lý:

- Việc tính tiền cho mỗi cuộc gọi như sau:

Tiền = Giá cơ bản \* Số phút \* Hệ số miền.

Giá cơ bản là 400đ.

Hệ số miền đối với nội hạt là 1, với lân cận là 2, với xa là 3, với rất xa là 4.

Đối với các cuộc gọi bắt đầu từ 23h00 đến 5h00 các ngày trong tuần và ngày Thứ Bảy, Chủ nhật thì được giảm giá 30%

## **Đề tài 20: Học từ vựng tiếng anh thông qua trò chơi**

Mô tả đề tài: Xây dựng trò chơi đồ chữ đơn giản để giúp người chơi củng cố từ vựng trong Tiếng Anh.

Yêu cầu:

Về lý thuyết: - Nắm vững cơ sở lý thuyết về cấu trúc dữ liệu, đặc biệt là cấu trúc bảng băm và các kỹ thuật lưu trữ ngoài. Vận dụng các lý thuyết này vào cài đặt tự điển.

Về lập trình:

*Các chức năng chính của chương trình :*

- **Thêm từ vựng:** cho phép thêm từ vựng mới vào trong tự điển và lưu tự điển vào tập tin. Với mỗi từ trong tự điển cần lưu trữ các thông tin về loại từ (danh từ, động từ,...), phiên âm quốc tế và các nghĩa có thể có của từ.

- **Hiện thị & cập nhật tự điển:** Hiện thị danh sách các từ có trong tự điển để kiểm tra. Sau khi hiện thị, người dùng có thể thực hiện các thao tác sau:

+ Hiện thị thông tin chi tiết của một từ vựng.

+ Sửa thông tin chi tiết của một từ vựng.

+ Xóa một từ vựng nào đó.

- **Chơi trò chơi đồ chữ:** Chương trình chọn ngẫu nhiên một từ vựng nào đó, xóa một cách ngẫu nhiên một ký tự trong từ và yêu cầu người chơi nhập vào ký tự bị thiếu. Nếu người chơi nhập vào một ký tự làm thành một từ hợp lệ (nghĩa là có trong tự điển) thì chương trình hiện thị thông tin về từ đó, người chơi sẽ được tính điểm. Còn nếu người chơi nhập vào ký tự không làm thành từ hợp lệ, thì chương trình sẽ hiện thị thông báo lỗi: ‘Không có từ này trong tiếng Anh’

Ví dụ: Chương trình hiện thị từ c\_t:

- Nếu người chơi nhập vào chữ u thì sẽ tạo thành từ cut, khi đó chương trình sẽ hiện thị thông tin về từ cut và tính điểm cho người chơi.
- Hoặc nếu người chơi nhập vào chữ a thì sẽ tạo thành từ cat, khi đó chương trình sẽ hiện thị thông tin về từ cat và tính điểm cho người chơi.
- Hoặc nếu người chơi nhập vào chữ i thì sẽ tạo thành từ cit, thì chương trình sẽ hiện thị thông báo lỗi: ‘Không có từ này trong tiếng Anh’.

Môi trường cài đặt : C, C ++

## **Hướng dẫn chung cho các đề tài từ 21-25:**

Tạo file văn bản có lưu trữ khoảng 10-20 mẫu tin (record). Viết chương trình quản lý đọc dữ liệu từ file và thực hiện các chức năng:

- In danh sách các thông tin của tất cả các đối tượng (có bổ sung cột số thứ tự)
- Bổ sung thêm 1 đối tượng vào danh sách.
- Tìm kiếm mẫu tin theo các tiêu chí khác nhau và hiện thị mẫu tin đó
- Sắp xếp theo thứ tự tăng dần/ giảm dần
- Chèn mẫu tin sao cho mảng vẫn được sắp theo thứ tự
- Xóa 1 mẫu tin

## **Đề tài 21: Quản lý sinh viên**

Thông tin sinh viên cần quản lý gồm: mã sinh viên, họ, tên, ngày sinh (ngày, tháng, năm), giới tính, lớp (bắt đầu là khóa học, ví dụ: 12T3, 13T2), điểm toán, điểm lý, điểm tin, *điểm trung bình, học bổng*. Bổ sung các chức năng sau:

- Tính điểm trung bình (ĐTB) và học bổng cho các sinh viên theo qui ước:

$$HB = \begin{cases} 200 & \text{nếu } ĐTB \geq 8 \\ 100 & \text{nếu } 7 \leq ĐTB < 8 \\ 0 & \text{nếu } ĐTB < 7 \end{cases}$$

- In ra bảng thông kê theo lớp gồm các cột: lớp, số sinh viên, tổng học bổng
- Xóa các sinh viên nữ khóa 08

### Đề tài 22: Quản lý vật tư

Thông tin vật tư cần quản lý gồm: mã vật tư, tên vật tư, loại vật tư, đơn vị tính, ngày nhập (ngày, tháng, năm), nhà sản xuất, số lượng, đơn giá, *thành tiền*. Bổ sung các chức năng sau:

- Xóa các vật tư có số lượng lớn hơn 100
- Tính thành tiền (TT) = số lượng \* đơn giá và được giảm theo qui ước:

$$\text{giảm} = \begin{cases} 25\% & \text{nếu số lượng} > 200 \\ 10\% & \text{nếu } 100 < \text{số lượng} \leq 200 \\ 0 & \text{nếu số lượng} \leq 100 \end{cases}$$

- In ra bảng thông kê theo loại vật tư gồm các cột: loại vật tư, số lượng, tổng tiền

### Đề tài 23: Quản lý nhân viên

Thông tin nhân viên cần quản lý gồm: mã nhân viên, họ, tên, ngày sinh (ngày, tháng, năm), giới tính, đơn vị, chức vụ, hệ số lương, *lương*, *phụ cấp*, *thực lĩnh*. Bổ sung các chức năng sau:

- Xóa các nhân viên có độ tuổi > 50
- Tính lương, phụ cấp (PC), thực lĩnh (TL) cho các nhân viên theo qui ước:  
Lương = hệ số lương \* lương cơ bản (LCB = 850)

$$PC = \begin{cases} 40\% \text{ lương} & \text{nếu chức vụ là GD hoặc PGD} \\ 25\% \text{ lương} & \text{nếu chức vụ là TP hoặc PP} \\ 0 & \text{cho các đối tượng còn lại} \end{cases}$$

$$TL = \text{lương}$$

- In ra bảng thông kê theo đơn vị gồm các cột: đơn vị, số nhân viên nam, số nhân viên nữ, tổng thực lĩnh

### Đề tài 24: Quản lý hàng nhập trong 1 tháng

Thông tin phiếu nhập kho hàng gồm: mã hàng, tên hàng, đơn vị tính, ngày nhập, số lượng, đơn giá, *thành tiền*. Bổ sung các chức năng sau:

- Tính thành tiền (TT) = số lượng \* đơn giá và được giảm theo qui ước:

$$\text{giảm} = \begin{cases} 15\% & \text{nếu số lượng} > 100 \\ 10\% & \text{nếu } 50 < \text{số lượng} \leq 100 \\ 0 & \text{nếu số lượng} \leq 50 \end{cases}$$

- In ra bảng thống kê theo từng ngày trong tháng gồm các cột: ngày nhập, tổng số tiền
- Xóa các mặt hàng có đơn vị tính là “hộp” và có mã bắt đầu là “mh1”

**Đề tài 25:** Quản lý thông tin khách ở khách sạn trong 1 tháng

Thông tin cần quản lý gồm: mã hóa đơn, tên khách, số chứng minh, giới tính, số phòng (bắt đầu là A, hoặc B, .... Ví dụ: A203, B119, ...), *loại phòng*, *đơn giá*, ngày đến, ngày đi, *tiền phòng*. Bổ sung các chức năng sau:

- Tính loại phòng, đơn giá (DG), tiền phòng theo qui ước:

$$DG = \begin{cases} 400 & \text{nếu phòng loại A} \\ 300 & \text{nếu phòng loại B} \\ 250 & \text{cho các trường hợp còn lại} \end{cases}$$

Loại phòng là ký tự đầu tiên của số phòng

Tiền phòng = DG \* số ngày ở

- In ra bảng thống kê theo loại phòng gồm các cột: loại phòng, tổng tiền
- Xóa các hóa đơn có khách tên “Minh”

**Đề tài 26: Đề tài 1:** Xây dựng chương trình quản lý bài báo khoa học sử dụng danh sách liên kết

Thông tin bài báo khoa học bao gồm: Mã bài báo, Tên bài báo, Tên tạp chí, Loại tạp chí (SCI, SCIE, ISI, SCOPUS & Tạp chí khác), Số xuất bản, Tập xuất bản, Năm xuất bản, Nhà xuất bản, Tác giả chính (mỗi bài báo yêu cầu phải có ít nhất 1 tác giả chính, có thể có nhiều hơn 1 tác giả chính), Đồng tác giả (mỗi bài báo có thể có hoặc không có đồng tác giả, số lượng đồng tác giả có thể là nhiều người).

Chức năng hệ thống:

- Import dữ liệu từ file .txt;
- Hiện thị dữ liệu các bài báo;
- Thêm, sửa, xóa bài báo khoa học (lưu ý: có thể thêm cuối danh sách hoặc lựa chọn 1 vị trí bất kỳ để chèn vào)
- Tìm kiếm bài báo theo: Tên bài báo
- Filter bài báo theo: Năm xuất bản, Tên tác giả (có thể là Tác giả chính hoặc Đồng tác giả), Tên tạp chí & Loại tạp chí.
- Sắp xếp danh sách bài báo theo: Năm xuất bản và Tên bài báo.
- Export dữ liệu ra file .txt.

**Đề tài 27:** Danh bạ điện thoại

Danh bạ điện thoại cố định là một danh sách tuyến tính, mỗi phần tử của nó ứng với một đơn vị thuê bao gồm 03 (ba) trường:

- Tên đơn vị hoặc chủ đơn vị thuê bao.
- Địa chỉ
- Số điện thoại (không bao gồm mã vùng).

Một đơn vị Tổng đài muốn quản lý tất cả thuê bao của tất cả các tỉnh (thành phố) nên muốn tập hợp dữ liệu từ 01 tập tin có nhiều trường có cấu trúc như sau:

(xxxx) *Tên tỉnh 1*

*Tên đơn vị 1    Địa chỉ 1    Số điện thoại 1*



Tên đơn vị 2    Địa chỉ 2    Số điện thoại 2

....

(xxxx) Tên tỉnh 2

Tên đơn vị 1    Địa chỉ 1    Số điện thoại 1

Tên đơn vị 2    Địa chỉ 2    Số điện thoại 2

....

Chọn cấu trúc dữ liệu để lưu trữ Danh bạ trên và viết chương trình đọc dữ liệu từ tập tin lưu trữ trên máy tính. Yêu cầu thực hiện các công việc sau:

1. Tìm và thông báo nếu có 2 số điện thoại trùng nhau trong một tỉnh và yêu cầu người dùng xóa 1 số nếu trùng.
2. Liệt kê danh bạ điện thoại từng tỉnh khi có yêu cầu.
3. Xây dựng chức năng tìm kiếm và hiển thị thông tin theo Số điện thoại hoặc tên đơn vị.

Dữ liệu vào: Tập chứa danh sách danh bạ các tỉnh.

Phương pháp gợi ý: Sử dụng danh sách liên kết, ngăn xếp, hàng đợi.....

Dữ liệu ra: Hiển thị các thao tác theo menu chọn và lưu trữ sang tập tin nếu có yêu cầu.

### **Đề tài 28:** Sắp xếp lịch học tập

Một sinh viên CNTT nếu muốn đạt một cấp độ nào đó trong quá trình học thì phải hoàn thành một số môn học theo quy định được mô tả như dưới đây:

Số hiệu môn học	Tên môn học	Môn cần học trước
M1	Nhập môn tin học	Không
M2	Giải tích số	M4
M3	Cấu trúc dữ liệu và giải thuật	M1, M4
M4	Giải tích	Không
M5	Lập trình Assembly	M1

.....

.....

.....

Chọn cấu trúc dữ liệu để lưu trữ Bảng tiến độ học tập trên và viết chương trình đọc dữ liệu từ tập tin lưu trữ trên máy tính. Yêu cầu thực hiện các công việc sau:

1. Kiểm tra tính hợp lý của dữ liệu. Ví dụ

Số hiệu môn học	Tên môn học	Môn cần học trước
M1	Nhập môn tin học	M2
M2	Giải tích số	M1

Thì dữ liệu không phù hợp

2. Xây dựng lại cách lưu trữ dữ liệu theo dạng cây quan hệ.
3. Xuất ra tập tin các trình tự học tập của sinh viên thỏa mãn yêu cầu  $x$  (số môn tối thiểu  $\leq x \leq$  số môn tối đa).

Dữ liệu vào: Tập tin chứa bảng mô tả môn chương trình học.

Phương pháp gợi ý: Sử dụng danh sách liên kết, sắp xếp tôpô.....

Dữ liệu ra: Hiển thị các thao tác theo menu chọn và lưu trữ sang tập tin nếu có yêu cầu.

### **Đề tài 29:** Sơ đồ mạng

Trường Đại học Bách khoa bao gồm nhiều khu giảng đường khác nhau. Tổ quản trị mạng cần xây dựng một sơ đồ mạng để kết nối tất cả các khu với nhau. Các khu có thể kết nối được với 1 số khu khác hoặc không. Trong 1 khu các phòng có thể kết nối được với nhau hoặc không. Kèm theo là độ dài chi phí dây mạng cần kết nối

Ví dụ:

Ký hiệu Khu

Khu có thể kết nối

A	C (150), E (100), F (120)
B	C (90), F (50)
C	A (150), B (90), E (70), F (80)
Ký hiệu Phòng	Phòng có thể kết nối
A101	A102 (5), A103 (9), A201 (12)
A102	A202 (12), A135 (15)
C105	C205 (12)

.....

....

Chọn cấu trúc dữ liệu để lưu trữ sơ đồ kết nối của các Khu, Phòng.

1. Đọc dữ liệu từ tập tin lưu trữ các Khu, Phòng.
2. Đưa ra Sơ đồ kết nối các Khu với chi phí thấp nhất

Dữ liệu vào: Tập tin chứa bảng mô tả kết nối và chi phí giữa các khu, phòng.

Phương pháp gợi ý: Sử dụng cây khung nhỏ nhất

Dữ liệu ra: Tập tin chứa các lưu trữ mạng kết nối các Khu, Phòng với chi phí thấp nhất.

### **Đề tài 30:** Ong tìm mật

Một chú ong đi hút mật trong rừng, lượng mật thu nhận được ước lượng sẵn trên đường đi từ đỉnh đồi A đến đỉnh đồi B. Nhiệm vụ chú ong được giao là từ tổ (xuất phát) phải đi đến hết tất cả các đỉnh đồi mỗi đỉnh 1 lần và quay về tổ (vị trí) với lượng mật thu được là nhiều nhất. Dữ liệu được mô tả như sau

Ví dụ:

D	//D là đỉnh chứa tổ của Ong
A	B      100    //Lượng mật thu nhận khi đi từ đỉnh A đến đỉnh D hoặc ngược lại
A	D      200

.....

Chọn cấu trúc dữ liệu để mô tả dữ liệu trên.

Dữ liệu vào: Tập tin chứa bảng mô tả dữ liệu.

Phương pháp gợi ý: Sử dụng cây khung nhỏ nhất

Dữ liệu ra: Tập tin chứa hành trình tối ưu của Chú ong.

**Đề tài 31:** Viết chương trình minh họa các giải thuật tìm kiếm và sắp xếp trên mảng có kích thước n phần tử nguyên. Chương trình được mô tả với các yêu cầu như sau:

- Cài đặt hàm tìm kiếm:
  - Tìm kiếm tuần tự (tuyến tính) cho mảng bất kỳ
  - Tìm kiếm nhị phân cho mảng dữ liệu được sắp tăng
- Cài đặt các hàm sắp xếp (tăng) theo phương pháp:
  - Chọn trực tiếp
  - Chèn trực tiếp
  - Nổi bọt
  - Đổi chỗ trực tiếp
  - Shell Sort
  - Quick Sort

Các hàm sắp xếp phải minh họa trực quan: tại mỗi bước hoán vị  $a[i]$  và  $a[j]$

**Đề tài 32:** Minh họa chương trình quản lý sách đơn giản trong thư viện. Sử dụng cấu trúc dữ liệu danh sách liên kết đơn để cài đặt danh sách chứa nội dung các cuốn sách.

A. Thông tin liên quan đến một cuốn sách gồm: Mã số sách, Tên sách, Tên tác giả, Nhà xuất bản, Năm xuất bản, Trạng thái sách: {TRUE: chưa mượn/ FALSE: đã mượn}

B. Các thao tác trên danh sách này:

1. Khởi tạo danh sách

- a. Khởi tạo danh mục sách rỗng (chưa có sách)
- b. Đọc từ file: nhập vào tên file đã lưu danh mục sách ở lần làm việc trước đó.

2. Thêm một cuốn sách vào danh sách

- a. Thêm vào đầu danh sách: InsertFirst
- b. Thêm vào sau một cuốn sách nào đó: InsertAfter
- c. Thêm vào cuối của danh sách: InsertLast

3. Xóa một cuốn sách khỏi danh sách theo

- a. Mã số
- b. Tên sách
- c. Tên tác giả: (nếu tác giả có nhiều sách thì xóa hết)
- d. Xóa cuốn sách ở đầu danh sách
- e. Xóa cuốn sách ở sau cuốn sách có mã số nào đó
- f. Xóa cuốn sách ở cuối danh sách

4. Tìm kiếm sách theo: lưu ý kết quả có bao nhiêu thì liệt kê ra hết, không phải chỉ liệt kê kết quả đầu tiên

- a. Tên sách
- b. Tên tác giả.
- c. Tên nhà xuất bản

5. Chức năng xem danh sách

- a. Xem toàn bộ danh mục sách trong danh sách
- b. Xem những cuốn sách đang cho mượn
- c. Xem những cuốn sách chưa cho mượn
- d. Xem danh sách theo thứ tự:
  - i. Theo vần alphabet của tên sách
  - ii. Theo vần alphabet của tên tác giả
  - iii. Theo vần alphabet của tên nhà xuất bản
  - iv. Sách được xuất bản mới nhất (theo năm)

6. Chức năng mượn/trả sách

- a. Mượn sách: liệt kê những cuốn sách chưa cho mượn, cho user chọn một cuốn sách  $\Rightarrow$  rồi cập nhật lại trạng thái cho mượn của sách.
- b. Trả sách: nhập vào mã số sách được trả  $\Rightarrow$  cập nhật lại trạng thái đã trả sách cho cuốn sách đó.

7. Chức năng chỉnh sửa nội dung của sách: cho phép chọn các thông tin của sách để sửa và sau đó cập nhật lại.

8. Chức năng lưu file: nhập vào một tên file rồi lưu toàn bộ trạng thái hiện tại của danh mục sách vào đó.

Yêu cầu: Sinh viên phải thực hiện theo các yêu cầu sau:

1. Cài đặt cấu trúc dữ liệu Book theo mô tả như phần A

2. Cài đặt cấu trúc BookNode là phần tử của danh sách liên kết chứa danh mục sách

3. Cài đặt toàn bộ các chức năng mô tả trong phần B từ 1 -> 8: thể hiện các chức năng này theo *menu* chọn.

Ngoài ra sinh viên có thể bổ sung những chức năng mở rộng tùy ý. Tất cả các chức năng nâng cao này đều được đánh giá cao!

**Đề tài 33:** Trò chơi NIM: Có  $n$  đồng sỏi, mỗi đồng có một số viên sỏi. Hai người chơi luân phiên nhau chơi như sau: Đến lượt người nào, người đó tùy chọn một đồng sỏi để bốc, và bốc ra một số viên sỏi (ít nhất 1 viên hoặc nhiều nhất là hết các viên sỏi trong đồng đã chọn). Ai bốc cuối cùng là thua. Lập chương trình tổ chức chơi giữa người và máy tính theo các yêu cầu:

- Số đồng sỏi lúc đầu được nhập từ bàn phím
- Số lượng viên sỏi trong mỗi đồng được sinh ngẫu nhiên
- Máy sẽ gieo xu để xác định người hay máy đi trước
- Có thông báo số lượng trong mỗi đồng trước và sau mỗi lượt đi
- Thông báo kết quả cuối cùng

**Đề tài 34:** Một quầy trả tiền có  $N$  loại tiền với các mệnh giá (giá trị tiền ghi trên tờ tiền) là  $A[1], A[2], \dots, A[N]$  (Các  $A[i]$  là nguyên dương và khác nhau từng đôi). Giả thiết loại tiền mệnh giá  $A[i]$  có  $B[i]$  tờ ( $1 \leq i \leq N$ ). Có  $M$  khách (được đánh số hiệu từ 1 đến  $M$ ) cần lấy tiền. Số tiền khách  $j$  cần lấy là  $K[j]$ ,  $K[j]$  nguyên dương,  $1 \leq j \leq M$ ). Quy định rằng với mỗi khách hoặc quầy từ chối chưa trả tiền hoặc quầy phải trả đúng số tiền mà khách cần lấy. Dữ liệu vào được cho trong file văn bản có tên INP.TXT trong đó dòng đầu ghi giá trị  $N$  ( $N \leq 10$ ), dòng tiếp theo ghi các giá trị  $A[1], A[2], \dots, A[N]$ , dòng tiếp theo ghi các giá trị  $B[1], B[2], \dots, B[N]$ , sau đó là dòng ghi giá trị  $M$  ( $M \leq 20$ ), cuối cùng là dòng ghi các giá trị  $K[1], K[2], \dots, K[M]$ , tất cả các giá trị đều nguyên dương.

- Đọc file dữ liệu và đưa ra màn hình nội dung file dữ liệu (theo thứ tự trên).
  - Tìm cách trả tiền sao cho trả được nhiều khách nhất. Thông báo kết quả ra file văn bản với tên OUT2.TXT trong đó dòng đầu ghi số khách được trả tiền, trong các dòng tiếp theo, mỗi dòng ghi thông tin về một khách được trả tiền gồm số hiệu của khách, số tiền phải trả và dãy các số  $X[1], X[2], \dots, X[N]$ , trong đó  $X[i]$  là số tờ của loại tiền mệnh giá  $A[i]$ ,  $1 \leq i \leq N$ , được trả cho khách.
  - Tìm cách trả tiền sao cho trả được nhiều tiền nhất. Thông báo kết quả ra file văn bản với tên OUT3.TXT trong đó dòng đầu ghi tổng số tiền đã trả được, trong các dòng tiếp theo, mỗi dòng ghi thông tin về một khách được trả tiền theo quy cách giống câu 2
- Chú ý: Hai giá trị liền nhau trên một dòng của các file văn bản cách nhau ít nhất một dấu trắng.

**Đề tài 35:** Ở một nước nọ do các đội bóng đá thiếu tinh thần thi đấu nên số trận đấu hoà rất nhiều. Ban tổ chức quyết định thay đổi luật để kích thích các đội thi đấu tích cực hơn. Giải sẽ tổ chức thi đấu vòng tròn nghĩa là mỗi đội đều phải đấu với tất cả các đội khác đúng một trận. Nếu hoà, cả hai đội đều bị 0 điểm. Do sơ xuất của ban tổ chức, bảng kết quả thi đấu của tất cả các đội bị nhòe một số chỗ không đọc được. Ban tổ chức quyết định gửi bảng đó đến một trung tâm Tin học nhờ xem xét việc khôi phục lại các thông tin bị mất.

Giả sử số đội bóng bằng  $N$  ( $N \leq 20$ ), bảng kết quả được cho bởi một mảng số nguyên  $A[1..N, 1..N+1]$ , trong đó  $A[i, j]$  bằng số điểm đội  $i$  đạt được trong trận đấu với đội  $j$  ( $1 \leq i, j \leq N$ ) và  $A[i, N+1]$  bằng tổng số điểm của đội  $i$  trong toàn giải. Để đầy đủ, ta quy ước  $A[i, j] = 0$  ( $1 \leq i \leq N$ ).

Dữ liệu vào được cho trong một file văn bản với tên là INP.TXT, trong đó dòng đầu tiên ghi giá trị  $A[i, 1], A[i, 2], \dots, A[i, N], A[i, N+1]$ . Các giá trị này cách nhau ít nhất một dấu trắng và quy ước rằng giá trị nào bị nhòe thì tại đó được thay bằng dấu?..

Lời giải cần đưa ra một file văn bản với tên là OUT.TXT trong đó liệt kê tất cả các khả năng có thể có. Với mỗi bảng kết quả được khôi phục, cần ghi số thứ tự của bảng trên

một dòng, các dòng tiếp theo, ghi các dòng của bảng giống như trong file dữ liệu vào, trong đó các dấu? được thay bằng các giá trị tương ứng đã khôi phục.

Ví dụ với file dữ liệu vào INP.TXT.

```
4
0 1 3 ? 7
? 0 ? ? 6
1 ? 0 1 ?
? ? ? 0 ?
```

File in kết quả OUT.TXT sẽ là :

```
1
0 1 3 3 7
3 0 3 0 6
1 1 0 1 3
1 0 3 0 4
2
0 1 3 3 7
3 0 0 3 6
1 0 0 1 2
1 1 3 0 5
```

**Đề tài 36:** Xây dựng dịch vụ làm mai qua máy tính: Nhập họ tên, giới tính, sở thích, rồi tìm một hay nhiều người khác trong tệp chứa các bản ghi gồm những mục tin nói trên có nhiều sở thích chung nhất

**Đề tài 37:** Có n loại bánh ga tô, mỗi loại m chiếc. Tất cả nxm chiếc bánh trên xếp đều trong n thùng, mỗi thùng m chiếc. Lập thuật toán để chọn ra từ n thùng trên, mỗi thùng một chiếc sao cho thu được n chiếc thuộc n loại bánh khác nhau.

**Đề tài 38:.** Tìm tất cả các số có sáu chữ số sao cho tất cả các tích của nó với 2, 3, 4, 5, 6 đều viết được từ số ban đầu bằng cách hoán vị các chữ số.

**Đề tài 39:** Bài toán mã đi tuần: con mã xuất phát từ một vị trí bất kỳ trên bàn cờ vua, đi qua tất cả các vị trí trên bàn cờ, cuối cùng trở về vị trí xuất phát

**Đề tài 40:** Xây dựng chương trình quản lý bài báo khoa học sử dụng danh sách liên kết Thông tin bài báo khoa học bao gồm: Mã bài báo, Tên bài báo, Tên tạp chí, Loại tạp chí (SCI, SCIE, ISI, SCOPUS & Tạp chí khác), Số xuất bản, Tập xuất bản, Năm xuất bản, Nhà xuất bản, Tác giả chính (mỗi bài báo yêu cầu phải có ít nhất 1 tác giả chính, có thể có nhiều hơn 1 tác giả chính), Đồng tác giả (mỗi bài báo có thể có hoặc không có đồng tác giả, số lượng đồng tác giả có thể là nhiều người).

Chức năng hệ thống:

- Import dữ liệu từ file .txt;
- Hiện thị dữ liệu các bài báo;
- Thêm, sửa, xóa bài báo khoa học (lưu ý: có thể thêm cuối danh sách hoặc lựa chọn 1 vị trí bất kỳ để chèn vào)
- Tìm kiếm bài báo theo: Tên bài báo

- Filter bài báo theo: Năm xuất bản, Tên tác giả (có thể là Tác giả chính hoặc Đồng tác giả), Tên tạp chí & Loại tạp chí.
- Sắp xếp danh sách bài báo theo: Năm xuất bản và Tên bài báo.
- Export dữ liệu ra file .txt.

**Đề tài 41:** Xây dựng chương trình tìm đường đi ngắn nhất trong ma trận (A\* hoặc Dijkstra)

Ngày xưa có một Quốc vương muốn gả công chúa với yêu cầu ai đến sớm nhất thì sẽ cưới được công chúa. Có 1 chàng hiệp sĩ vô cùng mừng rỡ và hăm hở lên đường, tuy nhiên do chàng rất nghèo chỉ có thể đi bộ nên chàng phải tìm con đường ngắn nhất để có thể đến kịp và rước được công chúa.

Bản đồ của vương quốc là một hình chữ nhật gồm  $m \times n$  ô vuông ( $m, n < 5000$ ), hiệp sĩ có thể đi theo bất cứ đường nào từ một ô đến các ô lân cận (ngang dọc hoặc chéo) thời gian để đi từ ô này sang ô kia là một canh giờ.

Các ô trên bản đồ được đánh tọa độ theo số dòng và số cột, ô ở góc dưới cùng bên tay trái bản đồ có tọa độ là (0, 0) và ô ở góc trên cùng bên tay phải bản đồ có tọa độ ( $m-1, n-1$ ). Hiệp sĩ không được đi ra khỏi vương quốc, nếu không sẽ bị nước lãng giếng tóm cổ tổng giam không lấy được công chúa. Trên bản đồ có một số ô là núi cao, vực sâu, rừng thiêng nước độc không thể đi vào được. Hãy viết chương trình cho biết nếu hiệp sĩ đi suốt ngày suốt đêm thì phải mất bao nhiêu canh giờ hiệp sĩ mới lên được kinh đô để cưới công chúa.

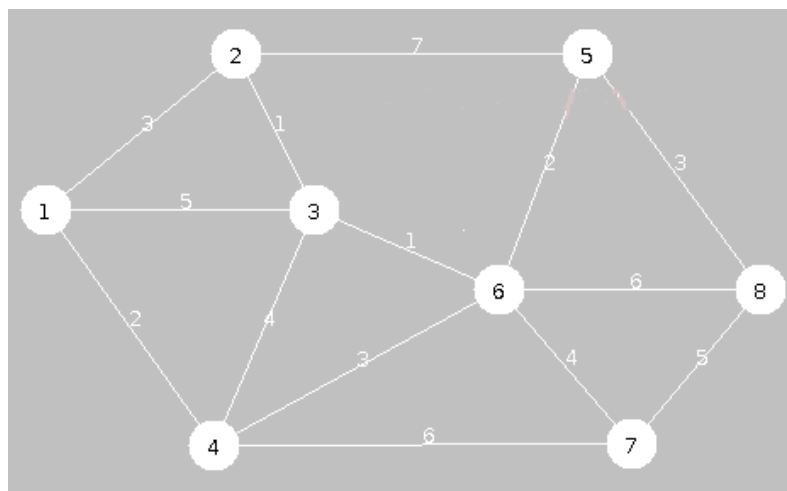
INPUT: là 1 file dữ liệu .txt, trong đó dòng đầu tiên của input chứa 6 con số cách nhau bởi khoảng trắng, hai con số đầu lần lượt là  $m, n$ , hai số tiếp theo là tọa độ nhà của hiệp sĩ và hai con số của cùng là tọa độ của hoàng cung nơi công chúa đang mỗi mắt trông một tấm chồng.  $m$  dòng tiếp theo trong input, mỗi dòng chứa  $n$  con số cách nhau bởi khoảng trắng. Đây là toàn bộ bản đồ vương quốc với con số 0 tượng trưng cho những ô có thể đi vào được và con số 1 tượng trưng cho những ô không thể đi vào được.

OUTPUT: Một con số duy nhất cho biết thời gian hiệp sĩ cần để đến được chỗ công chúa, tính theo canh giờ. Nếu hiệp sĩ không thể đến được chỗ công chúa xuất ra -1

**Đề tài 42:** xây dựng chương trình tìm đường đi ngắn nhất với Dijkstra và Floyd

Input: file .txt chứa dữ liệu: 3 chữ số dòng đầu tiên là số đỉnh của đồ thị, đỉnh xuất phát, đỉnh kết thúc; các dòng tiếp theo sẽ chứa các số nguyên biểu diễn ma trận kề

Ví dụ:



8 4 8

0 3 5 2 0 0 0 0

```

3 0 1 0 7 0 0 0
5 1 0 4 0 1 0 0
2 0 4 0 0 3 6 0
0 7 0 0 0 2 0 3
0 0 1 3 2 0 4 6
0 0 0 6 0 4 0 5
0 0 0 0 3 6 5 0

```

Lưu ý: khi chương trình đọc dữ liệu vào thì phải hiển thị Ma trận kề của đồ thị ra màn hình (nếu hiển thị được cả đồ thị thì càng tốt) và người dùng có thể thay đổi trọng số trên giao diện chương trình (sau khi thay đổi thì có thể lưu lại vào file dữ liệu ban đầu)  
 Kết quả có thể hiển thị trên màn hình và xuất ra file .txt

**Đề tài 43:** Xây dựng chương trình tìm chuỗi con chung lớn nhất của 2 hay nhiều chuỗi (Longest common substring) sử dụng quy hoạch động.

Mô tả bài toán tổng quát: Cho một tập chuỗi  $S = \{S_1, \dots, S_k\}$  trong đó  $|S_i| = n_i$  và  $\sum n_i = N$ . Với mỗi giá trị  $t$  thỏa mãn  $2 \leq t \leq k$ , tìm các chuỗi con chung dài nhất của  $k$  chuỗi.

**Đề tài 44:** Một công ty bưu điện chịu trách nhiệm chuyển phát thư một cách nhanh chóng giữa các thành phố. Có tất cả  $N$  thành phố được đánh số từ  $1..N$ , với  $M$  con đường hai chiều nối giữa các thành phố. Việc di chuyển trên mỗi con đường mất một khoảng thời gian là  $T$ . Bức thư từ thành phố  $A$  muốn gửi đến thành phố  $B$  phải đi theo một quy trình do công ty đặt ra. Đầu tiên nhân viên bưu điện mất một khoảng thời gian  $T_1$  vận chuyển thư từ thành phố  $A$  đến công ty bưu điện đặt tại thành phố 1 để kiểm tra tính hợp lệ của các bức thư, sau đó mất thêm một khoảng thời gian  $T_2$  để đưa các bức thư từ công ty đến thành phố  $B$ . Như vậy thời gian tổng cộng để gửi 1 bức thư từ thành phố  $A$  đến thành phố  $B$  là  $T_1 T_2$ .

Yêu cầu:

- Biết rằng có rất nhiều yêu cầu gửi thư giữa các thành phố khác nhau.
- Hãy tính thời gian ngắn nhất để một bức thư được gửi từ thành phố này đến thành phố khác.

- Sử dụng BFS & DFS

Input: file .txt.

- Dòng đầu tiên gồm 3 số nguyên  $N$  ( $2 \leq N \leq 50000$ ),  $M$  ( $N-1 \leq M \leq 100000$ ) và  $K$  ( $1 \leq K \leq 25000$ ) cho biết số lượng thành phố, số con đường hai chiều, và số yêu cầu gửi thư giữa các thành phố.

-  $M$  dòng tiếp theo, mỗi dòng ghi 3 số nguyên dương  $L, P, T$  ( $1 \leq L \neq P \leq N, 1 \leq T \leq 2000$ ) biểu diễn cho việc di chuyển trên con đường hai chiều nối thành phố  $L$  và  $P$ , sẽ mất khoảng thời gian là  $T$ .

-  $K$  dòng tiếp theo, mỗi dòng gồm 2 số nguyên  $A, B$  biểu diễn cho yêu cầu gửi thư từ thành phố  $A$  đến thành phố  $B$ .

- Output: Ghi ra file .txt gồm  $K$  dòng tương ứng với  $K$  yêu cầu gửi thư trong input, mỗi dòng gồm một số nguyên duy nhất biểu diễn thời gian ngắn nhất gửi một bức thư từ thành phố này đến thành phố khác.

Ví dụ:

Input	Output
6 7 31 2 35 4 33 1 16 1 93 4 21 4 43 2 22 45 13 6	6610