



Bài Tập Lớn 3

CHÚA TÊ NHỮNG CHIẾC NHẪN: SỰ TRỞ VỀ CỦA VỊ VUA

Phiên bản 1.0

1. Giới thiệu

Lại nói về Frodo, sau khi trốn thoát được các quái vật bóng tối, Frodo cùng với 1 người bạn của mình, Sam tiếp tục cuộc hành trình (và lẩn trốn các Ma nhân). Trên hành trình này, Gollum, người chủ sở hữu của chiếc nhẫn quyền lực trong 500 năm, đã trở nên gắn bó với Frodo và dẫn đường cho cả nhóm tìm đến Mordor, nơi chiếc nhẫn cần được huỷ trong ngọn núi lửa Doom (Mount Doom). Lúc này, sau khi đánh bại Saruman, các quái vật dừng hẳn tiếp tục tụ tập tại phía chân núi Mordor để bắt đầu một hành trình cam go nhất để huỷ bỏ chiếc nhẫn. Theo Gollum, có hai con đường đi đến ngọn núi lửa: con đường chính thức xuyên qua đại bản doanh của Chúa tể bóng tối và con đường kia xuyên qua một đường ngầm trong lòng các ngọn núi lửa. Do không thể xuyên thủng toàn bộ quân đội bóng tối để đi con đường chính thức, con đường ngầm đã được chọn. Và bây giờ, Frodo cùng cả đoàn quân đang đứng trước cổng vào con đường ngầm (Dark Portal), đối mặt với thử thách mới.

Mordor là một lục địa cổ đã chìm vào lòng đất từ lâu, bị bao bọc bởi các dòng dung nham nóng chảy. Rải rác giữa các dòng dung nham ấy là các mô băng nhỏ vừa đủ cho một người đặt chân, gọi là các *pole*. Tương truyền các *pole* này là các giọt lệ buồn vĩnh cửu của nàng *Arwen*, nên vĩnh viễn không tan, cho dù bị dung nham bao bọc. Vốn là người có khả năng điều khiển đại bàng, Gandalf bèn tổ chức dịch vụ *Eagle Express*, dùng đại bàng chở các hiệp sĩ di chuyển giữa các *pole*.

2. Yêu cầu

Trong bài tập lớn này, sinh viên sẽ được cung cấp một file chứa dữ liệu nhập chứa thông tin của một hiệp sĩ thám hiểm Mordor, cùng với địa hình Mordor bao gồm vị trí của Dark Portal và Mount Doom. Chương trình sẽ xuất ra một số nguyên tương ứng với nhiệm vụ của hiệp sĩ ở Mordor. Chi tiết cụ thể công việc sinh viên phải làm sẽ mô tả trong phần 4.

3. Dữ liệu nhập

Dữ liệu nhập của chương trình được chứa trong file mang tên `input.txt`. File này sẽ chứa các thông tin như sau:

-172

thông tin về hiệp sĩ thám hiểm Mordor

0100102

thông tin về địa hình Mordor

Như vậy file nhập sẽ bao gồm ít nhất hai dòng. Dòng đầu tiên sẽ có đúng một số nguyên N , giá trị tuyệt đối của số nguyên N này sẽ là chỉ số sức mạnh HP của hiệp sĩ đang thám hiểm Mordor. HP của hiệp sĩ sẽ là một số nguyên có giá trị từ 0 đến 999.



Kể từ dòng thứ hai của file nhập sẽ mô tả các đường bay của đại bàng giữa các pole. Mỗi chuyến bay sẽ được mô tả bằng một giá trị số, gọi là *mã phi tác*. Số chuyến bay là không cố định, có thể thay đổi tùy theo test case. Giữa hai pole có thể có nhiều chuyến bay khác nhau. Sẽ có tối đa 1000 chuyến bay. Nếu số chuyến bay nhiều, các mã phi tác có thể trình bày thành nhiều dòng.

Ý nghĩa của mã phi tác được diễn giải như sau: Nếu mã phi tác có dạng như sau $NNXXXMM$, có nghĩa là có đường đi từ pole NN đến pole MM của Mordor, tuy nhiên đường đi này bị canh giữ bởi một quái vật *Orc* có HP là XXX . Pole NN được gọi là *pole khởi hành*, pole MM được gọi là *pole đích đến*. Dark Portal sẽ nằm ở pole khởi hành của chuyến bay đầu tiên, Mount Doom nằm ở pole đích đến ở chuyến bay cuối cùng trong danh sách các chuyến bay. Giá trị XXX cũng được gọi là trọng số của chuyến bay.

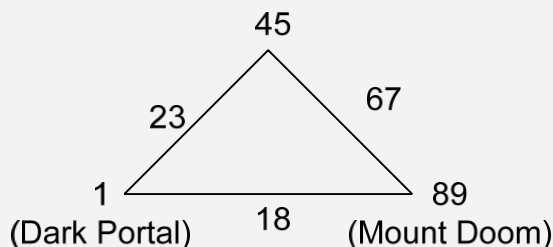
Nếu số N là số dương, hiệp sĩ khởi hành vào thời điểm các núi lửa ở Mordor không phun trào, các đại bàng có thể tự do bay đi về hai chiều giữa các pole. Tuy nhiên nếu N là số âm, lúc này các núi lửa đang phun, nên đại bàng chỉ có thể bay từ pole khởi hành đến pole đích đến, không thể bay theo chiều ngược lại. Muốn bay về theo chiều ngược lại, các hiệp sĩ phải tìm các chuyến bay khác.

Ví dụ 1: Với dữ liệu nhập là

125

0102345 4506789 0101889

Thì hiệp sĩ có *HP* là 125, địa hình Mordor sẽ được biểu diễn như hình vẽ sau:

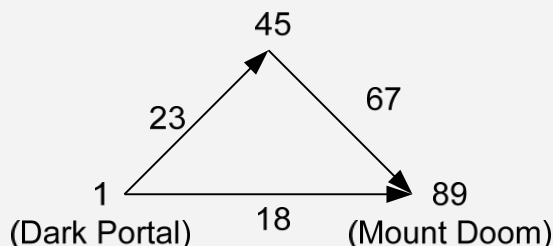


Ví dụ 2: Với dữ liệu nhập là

-125

0102345 4506789 0101889

Thì hiệp sĩ có *HP* là 125, địa hình Mordor sẽ được biểu diễn như hình vẽ sau:





4. Hiện thực chương trình

Sinh viên sẽ hiện thực một hàm *travel* có prototype như sau:

```
int travel(int N, eventList* pEvent)
```

Thông số N là giá trị của số nguyên N như đã mô tả ở phần 3. Thông số $pEvent$ là một con trỏ trỏ đến danh sách liên kết của các sự kiện được đọc từ file input, được định nghĩa như sau:

```
struct eventList {  
    long nEventCode;  
    eventList* pNext;  
}
```

5. Tính kết quả trả về

Kết quả trả về của hàm *travel* sẽ được tính như sau:

S1) Nếu HP của hiệp sĩ nằm trong khoảng $(0,200)$, đây là một hiệp sĩ tập sự được cử đến trinh sát Mordor. Hiệp sĩ tập sự này sẽ báo cáo lại thông tin Mordor dưới dạng một số nguyên I như sau:

$$I = HP_K + N_P + \sum_{Mordor} HP_O$$

Trong đó HP_K là HP của hiệp sĩ, N_P là số pole có dịch vụ đại bàng (đi hoặc đến) ở Mordor và HP_O là HP của các quái vật Orc đang canh gác Mordor

Ví dụ 3: Với dữ liệu nhập là

-125

0102345 4506789 0101889

thì kết quả trả về là $125 + 3 + (23+67+18) = 236$.

S2) Nếu HP của hiệp sĩ nằm trong khoảng $[200, 600)$, hiệp sĩ là một *Ringbearer*, lãnh nhiệm vụ mang chiếc nhẫn đi tiêu hủy. Kết quả trả về của hàm *travel* sẽ là **trọng số của con đường có trọng số nhỏ nhất từ Dark Portal đến Mount Doom**. Mặt khác, trọng số của con đường này phải nhỏ hơn HP của hiệp sĩ, nếu không hiệp sĩ sẽ không đủ sức đánh thắng các quái vật Orc chặn đường để mang chiếc nhẫn quyền lực đến Mount Doom. Trong trường hợp không tìm ra con đường như vậy, hàm trả về -1.

Định nghĩa 1: Trọng số của một con đường là tổng trọng số của tất cả các chuyến bay trên con đường đó.



Ví dụ 4: Với dữ liệu nhập là

250

0102345 4506789 0101889

thì kết quả trả về là **18**.

Ví dụ 5: Với dữ liệu nhập là

-250

0102345 4506789 0111889

thì kết quả trả về là $23+67 = \mathbf{90}$.

Ví dụ 6: Với dữ liệu nhập là

250

0109845 4509889 8909898

thì kết quả trả về là **-1**.

S3) Nếu *HP* của hiệp sĩ nằm trong khoảng $[600,999)$, hiệp sĩ này từng tham dự trận chiến ở Toà tháp đôi, được gọi là *Ranger*. Tính chiến đấu của Ranger rất cao. Trong số các con đường dẫn từ Dark Portal đến Mount Doom, Ranger sẽ **chọn con đường có trọng số cao nhất**. Kết quả trả về sẽ là trọng số của con đường được chọn. Nếu con đường được chọn có trọng số cao hơn HP của Ranger, Ranger sẽ chết trên đường đi và kết quả trả về là -1.

Ví dụ 7: Với dữ liệu nhập là

650

0109845 4509898 0117298

thì kết quả trả về là $98+98 = \mathbf{196}$.

Ví dụ 8: Với dữ liệu nhập là

650

0109845 4509898 0177298

thì kết quả trả về là -1. Lý do là đường đi có trọng số cao nhất là $01 \rightarrow 98$, trọng số $772 > 650$.

Chú ý là mặc dù còn con đường $01 \rightarrow 45 \rightarrow 98$ có trọng số 196 (nhỏ hơn HP của hiệp sĩ), nhưng đường trọng số cao nhất là $01 \rightarrow 98$ (trọng số 772) vẫn được chọn (và dẫn đến là hiệp sĩ đã tử nạn trên đường đi).



S4) Nếu mã phi tác có dạng XX000YY, thì chuyển bay cho biết là đường đi từ đỉnh XX đến đỉnh YY giam giữ nữ vương Galadriel (chứ không phải là một quái vật Orc có HP là 000). Nếu Galadriel xuất hiện ở Mordor, Ringbearer và Ranger sẽ chọn con đường dẫn từ Dark Portal đến Mount Doom đi qua nơi Galadriel bị giam để giải cứu nàng. Con đường được chọn vẫn phải thoả mô tả như đã mô tả trong mục S2 và S3. Nếu không tồn tại đường đi vừa giải cứu được Galadriel vừa đến được Mount Doom, kết quả sẽ trả về là -1. Chú ý: Galadriel chỉ xuất hiện tối đa 1 lần duy nhất.

Ví dụ 9: Với dữ liệu nhập là

650

0111145 0119846 4699845 0100002 0211103 0312145 0210145

Có bốn con đường dẫn từ Dark Portal đến Mount Doom là $01 \rightarrow 45$, $01 \rightarrow 46 \rightarrow 45$, $01 \rightarrow 02 \rightarrow 03 \rightarrow 45$ và $01 \rightarrow 02 \rightarrow 45$. Trong đó chỉ có 2 đường cuối là đi ngang qua nơi Galadriel bị giam giữ là $01 \rightarrow 02 \rightarrow 03 \rightarrow 45$ có trọng số là $(0+111+121)$ và $01 \rightarrow 02 \rightarrow 45$ có trọng số là $(000+101)$. Tuy nhiên do hiệp sĩ là một Ranger, hiệp sĩ sẽ chọn con đường có trọng số cao hơn, kết quả trả về sẽ là $0+111+121 = 232$.

Ví dụ 10: Với dữ liệu nhập là

650

0111145 0119846 4699845 0100002 0211103 0399845 0299845

Tương tự ví dụ 8, có 2 con đường có thể giải cứu được Galadriel. Cả 2 đường này đều có trọng số lớn hơn HP của hiệp sĩ. Kết quả trả về là -1.

S5) Nếu HP của quái vật Orc là 099, quái vật đó chính là Saruman đã biến thành Orc. Do mới biến từ người thành Orc nên HP của Saruman vẫn còn rất thấp. Tuy vậy các hiệp sĩ đều e ngại không dám giao chiến với Saruman. **Các Ringbearer và Ranger không chọn đường đi qua nơi Saruman canh giữ.** Nếu không có đường nào thỏa mãn các điều kiện như vậy (thỏa mãn điều kiện của Ringbearer/Ranger đồng thời không đi qua nơi Saruman canh giữ), kết quả trả về là -1. Chú ý: Saruman cũng chỉ xuất hiện tối đa 1 lần duy nhất.

Ví dụ 11: Với dữ liệu nhập là

250

0111102 0200145 0109945

Kết quả trả về là 112.

S6) Nếu HP của hiệp sĩ thám hiểm Mordor là 888, hiệp sĩ đó chính là Legolas. Legolas sẽ hành xử như một Ranger được mô tả ở trên. Tuy nhiên nếu cả nữ vương Galadriel lẫn Saruman cùng xuất hiện ở Mordor, Legolas sẽ chọn đường đi có chạm trán với Saruman và đồng thời cứu được Galadriel. Nếu không tồn tại đường đi như vậy, Legolas vẫn mang chiếc nhẫn quyền lực đến Mount Doom tiêu hủy như bình thường nhưng không cho biết đi bằng cách nào. (Kết quả trả về là -1)



Ví dụ 12: Với dữ liệu nhập là

888

0111102 0200145 0109945 4500246 0100246

Kết quả trả về là $111+1+2 = 114$ do Legolas bản thân cũng là một Ranger, chàng sẽ chọn đường đi có trọng số cao nhất cho phép (đồng thời tránh Saruman).

Ví dụ 13: Với dữ liệu nhập là

888

0111102 0200145 0109945 4500046 4600147 4500547

Kết quả trả về là $099+000+001 = 100$ (chọn đồng thời đường qua Saruman and Galadriel).

S7) Nếu HP của hiệp sĩ là 999, hiệp sĩ đó chính là pháp sư hùng mạnh Gandalf. Vốn không hề biết sợ, Gandalf muốn thám hiểm tất cả các pole ở Mordor. Tuy nhiên, tư duy quân sự tuyệt hảo trong ông khiến Gandalf chọn một cách vi hành (có thể đi qua một pole nhiều lần) sao cho tránh đụng độ với các Orc có HP lớn càng nhiều càng tốt.

Kết quả trả về trong trường hợp này là tổng trọng số trên tất cả các cạnh của cây phủ tối thiểu đồ thị (minimum spanning tree) các chuyên bay ở Mordor (tham khảo trong slide bài giảng hoặc tại địa chỉ http://en.wikipedia.org/wiki/Minimum_spanning_tree).

(Nếu không có tồn tại cây phủ tối thiểu nào, kết quả trả về là -1.)

Ví dụ 14: Với dữ liệu nhập là

999

0111102 0200145 0109945 4511246 0100146

Kết quả trả về là $099+001+001 = 101$.

S8) (*bonus*) Nếu HP của hiệp sĩ là 777, hiệp sĩ đó chính là Aragorn. Không bằng lòng với việc tiêu diệt chiếc nhẫn quyền lực, Aragorn còn muốn chiếm giữ vùng trọng tâm của Mordor khi núi lửa phun trào. Vùng trọng tâm của Mordor là một vùng liên thông đầy đủ (có khi là bao gồm toàn bộ Mordor), gọi là strongly connected component (http://en.wikipedia.org/wiki/Strongly_connected_component). Vùng này, **khi núi lửa phun trào**, bất kỳ đỉnh nào của vùng này cũng có thể đi đến được bất kỳ đỉnh nào khác trong vùng. (Xem thêm mục 6.)

6. Cách dịch và thực thi chương trình

Sinh viên download file *assignment3.zip* từ trang Web của môn học. Khi giải nén file này, sẽ có được các file sau:



input.txt	Một file input ví dụ.
main.cpp	Chương trình chính
doom.cpp	Chương trình hiện thực bởi sinh viên
defs.h	File định nghĩa các cấu trúc và hàm dùng chung

File `input.txt` là một file nhập mẫu như được mô tả ở phần 3. File `main.cpp` là chương trình khởi tạo, bao gồm các hàm viết sẵn như sau:

- `main()`: chương trình chính sẽ thực thi
- `readFile()`: hàm đọc file input
- `display()` : hàm xuất dữ liệu ra màn hình.

Trong hàm này sẽ có lời gọi đến một hàm `checkSCC` để kiểm tra liệu kế hoạch của Aragorn có thực hiện được hay không. Hàm `checkSCC` được khai báo trong file `defs.h` và được hiện thực trong file `doom.cpp`. Sinh viên cần viết lại hàm này nếu muốn có điểm bonus được mô tả trong phần S8.

Lưu ý rằng **sinh viên không được phép thay đổi file `main.cpp` và `defs.h` khi hiện thực chương trình**. Ngoài ra, các hàm do sinh viên viết không được xuất bất kỳ dữ liệu nào ra màn hình khi thực thi.

XỬ LÝ GIAN LẬN

Bài tập lớn phải được sinh viên TỰ LÀM. Sinh viên sẽ bị coi là gian lận nếu:

- Có sự giống nhau bất thường giữa mã nguồn của các bài nộp. Trong trường hợp này, **TẤT CẢ** các bài nộp đều bị coi là gian lận. *Do vậy sinh viên phải bảo vệ mã nguồn bài tập lớn của mình.*
- Sinh viên không hiểu mã nguồn do chính mình viết, trừ những phần mã được cung cấp sẵn trong chương trình khởi tạo. *Sinh viên có thể tham khảo từ bất kỳ nguồn tài liệu nào, tuy nhiên phải đảm bảo rằng mình hiểu rõ ý nghĩa của tất cả những dòng lệnh mà mình viết. Trong trường hợp không hiểu rõ mã nguồn của nơi mình tham khảo, sinh viên được đặc biệt cảnh báo là **KHÔNG ĐƯỢC** sử dụng mã nguồn này; thay vào đó nên sử dụng những gì đã được học để viết chương trình.*

KHÔNG CHẤP NHẬN BẤT KỲ GIẢI THÍCH NÀO - KHÔNG CÓ BẤT KỲ NGOẠI LỆ NÀO!

Sau mỗi bài tập lớn được nộp, sẽ có một số sinh viên được gọi phỏng vấn ngẫu nhiên để chứng minh rằng bài tập lớn vừa được nộp là do chính mình làm.