TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**Báo cáo Combinatorial Optimization**

*Người hướng dẫn*: **CÔ THÂN LÊ**

*Người thực hiện*: **PHẠM HOÀNG KHÔI NGUYÊN - 51703143**

Lớp: 17050301

Khoá  **: 21**

LỜI CẢM ƠN

Đầu tiên tôi xin gửi lời cảm ơn đến các thầy cô trong Khoa Công Nghệ Thông Tin vì đã mang dạy cho chúng tôi những kiến thức hay và có ích ,với sự nhiệt tình giảng dạy của thầy cô đã mang chúng tôi đến một vùng trời kiến thức.Đặc biệt tôi xin gửi lời cảm ơn trân trọng nhất đến thầy Lê vì thầy đã mang đến rất nhiều kiến thức cơ bản cũng như nâng cao của môn cấu trúc dữ liệu,với những kiến thức đó tôi đã có thể hoàn thành bài tập lớn lần này và cũng như có được những kiến thức này để giúp ích cho tương lai tôi khi đi ra xã hội.

Sau cùng,tôi xin kính chúc quý Thầy Cô trong Khoa Công Nghệ Thông Tin và thầy Lê dồi dào sức khỏe và niềm tin để tiếp tục thực hiện sứ mệnh cao đẹp của mình là truyền đạt kiến thức cho thế hệ mai sau.

TIỂU LUẬN ĐƯỢC HOÀN THÀNH TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG

Tôi xin cam đoan đây là sản phẩm tiểu luận của riêng tôi và được sự hướng dẫn của Thầy Thắng;. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình.** Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

TP. Hồ Chí Minh, ngày 17 tháng 10 năm 2018

Tác giả

(ký tên và ghi rõ họ tên)

Nguyên

Phạm Hoàng Khôi Nguyên

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

**Phần xác nhận của GV hướng dẫn**

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

**Phần đánh giá của GV chấm bài**

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

Tóm tắt

Phân tích và thiết kế chương trình phù hợp với 3 bài toán:tìm đường đi ngắn nhất,tìm cây khung ngắn nhất ,tìm đường đi có tải lớn nhất.

Mục Lục

[LỜI CẢM ƠN 1](#_Toc6520854)

[PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN 3](#_Toc6520855)

[Tóm tắt 4](#_Toc6520856)

[Mục Lục 5](#_Toc6520857)

1. Introduction

Trong bài báo cáo sẽ dùng 3 thuật toán khác nhau để giải quyết 3 bài toán khác nhau:

1. Cây khung ngắn nhất: Kruskal -Phạm Hoàng Khôi Nguyên

tìm cây khung ngắn nhất là bài toán được ứng dụng trong việc tìm kiếm các đường đi từ 1 điểm bất kì trên cây khung tới 1 điểm khác ngắn nhất.

1. Đường đi ngắn nhất: Dijkstras

Tìm đường đi ngắn nhất được ứng dụng trong các lọa tìm đường cho xe buýt ,ô tô, xe máy trên bản đồ với việc xác định khoảng cách và đường đi ngắn nhất tới điểm cần tìm

1. Tải lớn nhất và ngắn nhất của cây: Ford Fulkerson

Việc tìm tải lớn nhất và nhỏ nhất trên cây là việc được ứng dụng phổ biến trong việc tìm kiếm số lượng xe tải , xe máy hoặc các thứ khác đi từ 1 điểm tới 1 điểm trong cây với số lượng lớn nhất và nhỏ nhất được truyền qua cây đó.

1. State Of The Art

Các thư viện dùng trong chương trình là các thư viện có sẵn trong java và các class phụ bao gồm Edge.java , ArrayList (thư viện) , Main.java.

Các thuật toán được viết lại theo các bài thuật toán trên mạng để phù hợp hơn với yêu cầu đề bài.

1. Approaches
2. bài toán tìm cây khung ngắn nhất: qua việc tìm hiểu trên mạng về các thuật toán tìm cây khung ngắn nhất thì em chọn thuật toán Kruskal vì nó thông dụng va nằm trong quá trình giảng dạy của nhà trường.

Các bước trong quá trình hoạt động của thuật toán:

* 1. tìm cạnh nhỏ nhất trong các cạnh ngoài trừ các cạnh đã tìm
  2. kiểm tra việc thêm cạnh vào có khiến tạo vòng hay không
  3. thực hiện lại bước 1 nếu số cạnh thêm vào không bằng số đỉnh trừ 1 hoặc cạnh trên không là cạnh cuối trong dãy

1. Đường đi ngắn nhất:việc dùng thuật toán Dijkstras cho bài toán được em cân nhắc rất nhiều và quyết định dùng nó bởi tính thông dụng và nó khá tương tự với thuật toán Kruskal được dùng để làm bài tìm cây khung ngắn nhất.

Các bước của thuật toán:

* 1. xác định đỉnh được chọn làm mốc (đỉnh được chọn đầu tiên hoặc đỉnh có khoảng cách nhỏ nhất tới đỉnh được chọn đầu tiên) trong cây và đánh dấu đỉnh đó đã được chọn.
  2. Xét đỉnh được chọn với các đỉnh còn lại trong cây ngoài các đỉnh đã được chọn :
     1. Nếu đỉnh còn lại có khoảng cách tới đỉnh đầu qua các điểm lớn hơn đỉnh chọn cộng khoảng cách tới điểm còn lại thì ta sẽ đổi điểm chọn là điểm nằm trước điểm còn lại và cập nhật lại khoảng cách điểm còn lại bằng kiểm chọn + khoảng cách tới điểm còn lại.
     2. Nếu ngược lại thì ta bỏ qua đỉnh còn lại này
  3. Chạy lại bước 1 thuật toán sẽ dừng lại cho tới khi chạy xong n-1 lần trong đó n là số đỉnh của cây.

1. Tìm tải lớn nhất của cây :việc tìm tải trọng lớn nhất của cây được tìm theo nhiều cách trong các cách đó thì thuật toán của Ford Fulkerson là thuật toán thông dụng.

Quá trình hoạt động:

* 1. tìm đường đi có qua các điểm có tải lớn nhất (các điểm chứa nhiều tải nhất) từ điểm đầu tới điểm cuối.
  2. xác định nơi gây nghẽn tải (nơi có tải truyền qua là nhỏ nhất)
  3. lấy tải tại nơi nghẽn làm tải lớn nhất cho cả đường đi.Trừ tải tại nơi nghẽn vào các tải khác trên đường đi.
  4. Thực hiện lại bước a cho tới khi không tìm thấy đường đi từ điểm đầu tới điểm cuối.

1. Tìm tải nhỏ nhất cho cây: việc tìm tải ngắn nhất cho cả cây được nhóm em suy nghĩ ra và trình bày cách hoạt động như sau:
   1. tìm đường đi qua các tải nhỏ nhất ( các đoạn đường có thể chứa tải nhỏ nhất) từ điểm đầu tới điểm cuối
   2. tìm đoạn nghẽn trên đường đi của cây
   3. giá trị của đoạn nghẽn chính là tải nhỏ nhất của cây
2. Experiment and Result

Để tìm khung ngắn nhất của cây ta chạy file Main.java .Kết quả:

Source Destion Weight

3 5 1

1 4 2

0 1 3

0 2 4

1 3 4

Tìm đường đi ngắn nhất từ 1 điểm tới các điểm còn lại chạy file ShortestPath.java với điểm chọn là 0.Kết quả:

Vertex Distance from Source

0 0

1 4

2 12

3 19

4 21

5 11

6 9

7 8

8 14

Tìm tải lớn nhất của cây chạy MaximumCut.java. Kết quả:

Maximun flow in network: 7

1. Conclusion

Qua việc làm project này em đã biết thêm về việc phát triển 1 ứng dụng đáp ứng và phù hợp với yêu cầu người dùng là rất khó khăn nếu chỉ làm 1 người vì thế cần các thành viên trong team phối hợp và phát triển.

Các file trong bài thuyết trình đều được viết lại từ những code mẫu trên mạng hoặc ý tưởng sau khi xem xét thuật toán trên mạng với mỗi file code được chỉnh sữa hơn 50%.