

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ  
MINH  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN



BÀI TẬP NHÓM 7  
BỞI NHÓM 12

---

Giáo viên hướng dẫn: *Nguyễn Thanh Sơn*

Nhóm 12:

- *Hoàng Minh Thái - 23521414*
- *Nguyễn Trọng Tất Thành - 23521455*



## Mục lục

# GIẢI BÀI TẬP BỞI NHÓM 12

## 1. Bài toán Set Cover

### Mô tả bài toán

Cho một tập  $U$ , là một tập hợp các phần tử, và một tập hợp  $S = \{S_1, S_2, \dots, S_m\}$  gồm các tập con của  $U$ . Mỗi tập con  $S_i$  là một tập con của  $U$ , và mục tiêu là chọn một số ít các tập con sao cho tất cả các phần tử trong  $U$  đều được bao phủ, tức là mỗi phần tử của  $U$  phải xuất hiện ít nhất một lần trong các tập con đã chọn.

- Tập  $U = \{u_1, u_2, \dots, u_n\}$  là tập hợp các phần tử cần được bao phủ.
- Tập con  $S = \{S_1, S_2, \dots, S_m\}$  là tập hợp các tập con của  $U$ , mỗi tập con  $S_i$  chứa một số phần tử của  $U$ .
- Mục tiêu: Chọn một số ít các tập con từ  $S$  sao cho mỗi phần tử trong  $U$  xuất hiện ít nhất một lần trong các tập con được chọn. Tối thiểu hóa số lượng tập con được chọn.

### Định nghĩa chính thức

Cho:

- $U = \{u_1, u_2, \dots, u_n\}$ .
- $S = \{S_1, S_2, \dots, S_m\}$  với  $S_i \subseteq U$  cho mọi  $i$ .

Mục tiêu là tìm một tập con các tập con  $S' \subseteq S$  sao cho:

$$\bigcup_{S_i \in S'} S_i = U$$

và tối thiểu hóa số lượng các tập con được chọn, tức là:

$$\min |S'|.$$

## 2. Bài toán TSP (Travelling Salesman Problem)

### Mô tả bài toán

Bài toán TSP (Bài toán người du lịch) là một bài toán tối ưu hóa cổ điển trong lý thuyết đồ thị và tổ hợp. Mục tiêu của bài toán này là tìm một chu trình Hamilton (có thể hiểu là một con đường) trong đồ thị sao cho:

- Mỗi đỉnh được thăm đúng một lần.
- Tổng chi phí (hoặc tổng quãng đường) đi qua tất cả các đỉnh là nhỏ nhất.



## Định nghĩa chính thức

Cho một đồ thị  $G = (V, E)$ , trong đó:

- $V = \{v_1, v_2, \dots, v_n\}$  là tập hợp các đỉnh, mỗi đỉnh tương ứng với một thành phố.
- $E$  là tập hợp các cạnh nối các đỉnh.
- Mỗi cạnh  $(v_i, v_j) \in E$  có một trọng số  $c(v_i, v_j)$ , có thể là chi phí, thời gian hoặc khoảng cách di chuyển giữa các thành phố  $v_i$  và  $v_j$ .

Mục tiêu là tìm một chu trình Hamiltonian (một chuỗi các đỉnh mà bắt đầu từ một thành phố và quay lại thành phố ban đầu) sao cho tổng trọng số của các cạnh trong chu trình này là nhỏ nhất.

## 3. Thuật toán gần đúng

### Bài toán Set Cover

[label=)]**Thuật toán Greedy:**

1. (a) Khởi tạo  $S' = \emptyset$  và  $C = \emptyset$ , trong đó  $C$  là tập các phần tử đã được bao phủ.  
(b) Lặp lại cho đến khi  $C = U$ :
  - Chọn tập  $S_i \in S$  sao cho  $S_i$  bao phủ nhiều phần tử chưa được bao phủ nhất.
  - Thêm  $S_i$  vào  $S'$  và cập nhật  $C$ .  
(c) Trả về tập  $S'$ .

### 2. Thuật toán dựa trên lập trình tuyến tính:

- (a) Mô hình hóa bài toán dưới dạng bài toán quy hoạch tuyến tính.
- (b) Sử dụng các kỹ thuật làm tròn nghiệm để chuyển nghiệm liên tục thành nghiệm nguyên.

### Bài toán TSP

[label=)]**Thuật toán Greedy:**

1. (a) Bắt đầu từ một đỉnh bất kỳ.  
(b) Tại mỗi bước, chọn cạnh có trọng số nhỏ nhất kết nối đến một đỉnh chưa được thăm.  
(c) Lặp lại cho đến khi tất cả các đỉnh được thăm và quay lại đỉnh ban đầu.

### 2. Thuật toán 2-Approximation:

- (a) Tìm cây khung nhỏ nhất (Minimum Spanning Tree, MST) của đồ thị.
- (b) Tạo chu trình Euler bằng cách duyệt cây khung nhỏ nhất.
- (c) Biến chu trình Euler thành chu trình Hamilton bằng cách bỏ qua các đỉnh đã thăm nhiều hơn một lần.