

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



BÀI TẬP NHÓM 6
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

Bài 1: Thuật toán Prim

Thuật toán:

Input: $G = (V, E)$ là đồ thị vô hướng liên thông, biểu diễn dưới dạng danh sách kề với trọng số c_e cho mỗi cạnh $e \in E$.

Output: Một cây khung nhỏ nhất (MST) của G .

Thuật toán Prim [1] $X \leftarrow \{s\}$ s là một đỉnh bất kỳ $T \leftarrow \emptyset$ T chứa các cạnh của MST tồn tại (v, w) với $v \in X$ và $w \notin X$ $(v^*, w^*) \leftarrow$ cạnh có trọng số nhỏ nhất với $v^* \in X$ và $w^* \notin X$ Thêm đỉnh w^* vào X Thêm cạnh (v^*, w^*) vào T T

Phân tích độ phức tạp:

- Tìm cạnh có trọng số nhỏ nhất trong mỗi bước lặp: $O(E)$.
- Số vòng lặp tối đa: $O(V)$.
- Do đó, độ phức tạp tổng quát là $O(V \cdot E)$.
- Nếu sử dụng hàng đợi ưu tiên, độ phức tạp giảm xuống $O((V + E) \log V)$.

Phương pháp tối ưu:

Sử dụng cấu trúc dữ liệu hàng đợi ưu tiên để quản lý các cạnh, giảm độ phức tạp xuống $O((V + E) \log V)$.

Bài 2: Thuật toán Kruskal

Thuật toán:

Input: $G = (V, E)$ là đồ thị vô hướng liên thông, biểu diễn dưới dạng danh sách kề với trọng số c_e cho mỗi cạnh $e \in E$.

Output: Một cây khung nhỏ nhất (MST) của G .

Thuật toán Kruskal [1] $T \leftarrow \emptyset$ Sắp xếp các cạnh trong E theo trọng số tăng dần. mỗi cạnh $e \in E$ $T \cup \{e\}$ không tạo chu trình $T \leftarrow T \cup \{e\}$ T

Phân tích độ phức tạp:

- Sắp xếp các cạnh: $O(E \log E)$.
- Kiểm tra chu trình: sử dụng Union-Find với độ phức tạp $O(\alpha(V))$ cho mỗi thao tác, trong đó α là hàm nghịch đảo của hàm Ackermann.
- Tổng quát: $O(E \log E)$, và do $E \leq V^2$, độ phức tạp là $O(E \log V)$.



Phương pháp tối ưu:

Không cần tối ưu thêm, do $O(E \log V)$ đã là tốt nhất trong trường hợp tổng quát.

Bài 3: Thuật toán Huffman

Thuật toán:

Input: Một bảng ký tự α với xác suất xuất hiện p_a cho mỗi ký tự $a \in \alpha$.

Output: Cây mã Huffman.

Thuật toán Huffman [1] **Khởi tạo:** mỗi ký tự $a \in \alpha$ Tạo cây T_a chứa một nút duy nhất, gán nhãn " a ". $P(T_a) \leftarrow p_a$. $F \leftarrow \{T_a \mid a \in \alpha\}$. $|F| \geq 2$ Lấy hai cây T_1, T_2 trong F có $P(T_1), P(T_2)$ nhỏ nhất. Gộp T_1 và T_2 thành cây mới T_3 : gốc của T_1, T_2 là con trái, con phải của T_3 . $P(T_3) \leftarrow P(T_1) + P(T_2)$. Cập nhật $F \leftarrow F \setminus \{T_1, T_2\} \cup \{T_3\}$. $F[0]$.

Phân tích độ phức tạp:

- Tìm hai cây nhỏ nhất trong F : $O(|F| \log |F|)$ nếu sử dụng hàng đợi ưu tiên.
- Số vòng lặp: $O(|\alpha|)$.
- Tổng độ phức tạp: $O(|\alpha| \log |\alpha|)$.

Phương pháp tối ưu:

Không cần tối ưu thêm vì sử dụng hàng đợi ưu tiên đã đạt độ phức tạp tối ưu $O(|\alpha| \log |\alpha|)$.