

# Homework 6

## Greedy Methods (Phương pháp tham lam)

Họ tên: Phạm Vũ Anh Quân

MSV: 19000470

Gói C

### 1. Lý thuyết phương pháp

#### a. Ý tưởng

- Thuật toán tham lam làm việc tại từng bước. Ở mỗi bước, ta tìm lựa chọn tối ưu cục bộ với giả định rằng một lựa chọn cục bộ tốt tạo ra một lựa chọn tối ưu toàn cục.

#### b. Lược đồ tổng quát

```
Greedy(A)  $\equiv$   
  S =  $\emptyset$ ;  
  while (A  $\neq \emptyset$  and !IsSolution(S))  
    x = BestSelect(A);  
    A = A \ {x};  
    if (Acceptable(S, x))  
      Integrate(S, x);  
  endwhile;  
  return S;  
End.
```

#### c. Ưu điểm

- Dễ triển khai
- Độ phức tạp thời gian thấp
- Có thể được sử dụng để giải quyết các bài toán tối ưu

#### d. Nhược điểm

- Không phải lúc nào cũng tìm thấy được lựa chọn tối ưu toàn cục

### 2. Lập trình

- Làm 2 bài: Đổi tiền và Lập lịch sử dụng tài nguyên
- *ChangeCoins.py* và *ActivitySelection.py*
- Đánh giá: chọn k để đánh giá
  - Với bài Đổi tiền, k được dùng để tạo ra các tổng số tiền một cách ngẫu nhiên (k tổng số tiền được tạo, số chữ số của các tổng số tiền lần lượt từ 1 đến k).
  - Với bài Lập lịch sử dụng tài nguyên, k được dùng để tạo ra các tập hoạt động một cách ngẫu nhiên (k tập được tạo, số lượng hoạt động trong mỗi tập lần lượt là  $10^i$  với  $i=1..k$ ).

### 3. Chương 9 sách ‘Introduction to the Design and Analysis of Algorithms – Anany Levitin’

- Exercise 9.1:
  - Câu 10. Thuật toán Prim tự kiểm tra khả năng kết nối của đồ thị, do đó ta không phải tự mình làm điều đó.
  - Câu 11. Có 2 trở ngại nghiêm trọng mà ta gặp phải nếu xây dựng cây bao trùm tối thiểu bằng tìm kiếm đầy đủ:
    - Số lượng cây bao trùm phát triển theo cấp số nhân với kích thước đồ thị.
    - Việc tạo ra tất cả các cây bao trùm cho một đồ thị là không dễ dàng, nó khó hơn việc tìm một cây bao trùm tối thiểu bằng cách sử dụng một số thuật toán hiệu quả.
- Exercise 9.2:
  - Câu 1.
    - a.
      - Danh sách cạnh đã được sắp xếp:  $[bc, de, bd, cd, ab, ad, ce]$ .
      - $bc \rightarrow de \rightarrow bd \rightarrow ab \rightarrow \textbf{end}$ .
    - b.
      - Danh sách cạnh đã được sắp xếp:  
 $[de, cd, ef, ab, be, gh, ij, ad, cg, ej, ac, dh, fj, jl, bf, hi, gk, hk, kl, jl]$
      - $de \rightarrow cd \rightarrow ef \rightarrow ab \rightarrow be \rightarrow gh \rightarrow ij \rightarrow cg \rightarrow il \rightarrow gk \rightarrow \textbf{end}$ .
  - Câu 3. Thay vì lặp cho đến khi toàn bộ đỉnh được tìm thấy, ta duyệt qua toàn bộ cạnh trong danh sách cạnh đã được sắp xếp bởi vì với đồ thị không liên thông, thuật toán sẽ không bao giờ đạt được đủ số đỉnh.

(\*)

Source code đặt tại thư mục ‘src’.

Thực thi chương trình: `./filename` hoặc `python filename`

Cài đặt thư viện

- Matplotlib: `pip install matplotlib`
- Numpy: `pip install numpy`