BÀI TẬP 1: TỐI ƯU HÓA CHƯƠNG TRÌNH

Giả sử cho trước 1 danh sách 7000 tiếng trong tiếng Việt (danh sách A kèm theo) và 1 file văn bản tiếng Việt (file B.txt, nhập vào, font unicode) không giới hạn kích thước văn bản).

1. Hãy đếm có bao nhiêu tiếng trong danh sách A xuất hiện trong file B (không lặp lại)
2. Tính Tỷ lệ phần trăm: (đếm trong câu a / 7000) \*100.

Yêu cầu :

- Thời gian xử lý nhanh nhất

- Viết thành đoạn chương trình C++, report cách làm trong 1 trang A4 Word.

Ý tưởng: Ta tận dụng bộ nhớ để lưu tất cả các tiếng trong File A trong một map<string, int>. Khi đó mỗi tiếng là 1 key, value tương ứng = 1.

Khi đó ta chỉ cần kiểm tra tiếng x trong File B có thuộc File A hay không bằng cách kiểm tra trực tiếp điều kiện map[x] == 1.

Cách làm:

- Tạo file văn bản tiếng Việt (file B.txt) với nội dung bất kì được nhập vào bằng Unicode.

- Do file A và file B là file tiếng Việt font Unicode, ra cần thiết lập **\_setmode(\_fileno(stdout), \_O\_U16TEXT);** để đọc file dưới dạng UTF-16. Dùng biến **locale** để tránh lỗi khi in ra console.



- Ta viết hàm xử lý để tách từng tiếng đọc được từ file B và lưu chúng lại trong 1 **set<wstring>** (đặt tên là list) để không bị lặp lại. Do kí tự ở dạng UTF-16, khoảng giá trị nhị phân của mỗi kí tự ước lượng từ 65 - 8000, tiếng sẽ được tách ra khi gặp dấu phẩy, dấu chấm hay dấu cách.

- Với file A do mỗi tiếng nằm trên 1 dòng nên không cần tách, ta lưu tất cả chúng trong 1 **map<wstring, int>** với int = 1 (đặt tên là m). Dùng biến đếm kiểu int (khởi tạo = 0) để lưu kết quả câu a (đặt tên là dem)

- Kiểm tra xem tất cả phần tử của list có nằm trong m hay không, tức là kiểm **tra m[list[i]] == 1**. Nếu thỏa, tăng biến đếm lên 1.

- Tính tỉ lệ phần trăm: **dem / float(map<wstring, int>) \* 100**