**Báo cáo Mini-Project**

Họ và tên:

MSSV:

Bài 22

**Cách thực hiện**

- Nhận input từ bàn phím

- Thực hiện kiểm tra chuỗi nhập vào hợp lệ, chỉ chứa:

+ Các kí tự trong bảng chữ cái từ a-z hoặc A-Z

+ Ký tự phân cách (Space)

- Chuyển đổi tất cả từ chữ hoa về chữ thường

- Thực hiện phân tách các từ theo dấu phân cách (Space) / Tách từ

- Kiểm tra từng từ đó xem từ đó có phải là “cyclone word” không

**Các thanh ghi**

**$a0:** dùng để chứa địa chỉ của chuỗi nhập vào

**$a0, $a1:** dùng để truyền tham số vào chương trình con

**$v0, $v1:** dùng để chứa kết quả trả về của chương trình con

**$t0, $t1, $t2, $t3, $t8, $s0:** dùng trong chương trình con

**Chương trình con**

**1) check**

- Kiểm tra chuỗi đầu vào hợp lệ

- Chuyển từ ký tự in hoa sang ký tự thường

Duyệt lần lượt từ đầu chuỗi đến khi gặp ký tự \0:

+ Kiểm tra xem ký tự đó có thuộc: a-z, A-Z hoặc space (dấu cách) hay không => Nếu không thì báo lỗi và kết thúc chương trình

+ Nếu ký tự hợp lệ => Chuyển ký tự A-Z về a-z

Sau khi duyệt xong sẽ nhận được một chuỗi hợp lệ

**$a0**: địa chỉ của chuỗi nhập vào

**$t0**: chỉ mục để duyệt từng ký tự của chuỗi (i)

**$t1**: lưu địa chỉ phần tử thứ i

**$t2**: lưu giá trị phần tử thứ i

**2) load\_word**

- Tách các từ theo dấu phân cách (Space)

- Trả về địa chỉ và độ dài của từ

Duyệt từ vị trí đầu (hoặc tiếp tục từ vị trí sau dấu space):

+ Sau khi duyệt, nếu duyệt đến ký tự space hoặc \0 => Ta được 1 từ

+ Nếu độ dài của từ = 1 (chỉ có ký tự \0) => Kết thúc chuỗi

+ Nếu độ dài từ > 1 => Ta được một từ hợp lệ

**$a0**: địa chỉ của chuỗi nhập vào

**$t8**: vị trí ký tự đang duyệt

**$v0**: trả về địa chỉ của từ

**$v1**: trả về độ dài của từ

**3) is\_cyclone\_phrase**

- Kiểm tra xem từ có phải là cyclone word hay không

Hai con trỏ left và right:

+ Con trỏ left: Bắt đầu từ 0 và tăng dần

+ Con trỏ right: Bắt đầu từ length-1 và giảm dần (length – độ dài của từ)

Với mỗi lần lặp, ta kiểm tra:

+ Nếu left >= right => Kết thúc lặp

+ So sánh ký tự tại vị trí left (string[left]) với ký tự tại vị trí right (string[right])

\* Nếu string[left] > string[right] => Từ không phải là cyclone word, trả về 0

\* Nếu string[left] >= string[right] => Tăng left lên 1

+ So sánh ký tự tại vị trí left (string[left]) với ký tự tại vị trí right (string[right])

\* Nếu string[right] > string[left] => Từ không phải là cyclone word, trả về 0

\* Nếu string[right] >= string[left] => Giảm right đi 1

=> Sau khi kết thúc lặp: trả về 1 (cyclone word)

**$a0**: địa chỉ của từ (Address(string[0]))

**$a1**: độ dài của từ (length)

**$t0**: chỉ mục trái

**$t1**: chỉ mục phải

**$t2**: giá trị tại chỉ mục trái

**$t3**: giá trị tại chỉ mục phải

**Bài 21**

**Cách thực hiện:**

- Nhận input từ bàn phím

- Kiểm tra chuỗi số hợp lệ (chỉ gồm các chữ số 0 – 9), loại bỏ ký tự thừa ‘\n’ ở cuối

- Khởi tạo degree = 0

- Lặp:

+ Nếu độ dài chuỗi là 1 => Kết thúc lặp

+ Tính tổng các chữ số (chuyển từng ký tự sang chữ số và cộng)

+ Chuyển tổng vừa tìm được sang chuỗi mới => Tiếp tục áp dụng với chuỗi mới

- In degree ra màn hình