



ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
VNUHCM – UNIVERSITY OF SCIENCE

KIẾN TRÚC MÁY TÍNH VÀ HỢP NGỮ

BÀI BÁO CÁO ĐỒ ÁN THƯ VIỆN TIME

Nhóm thực hiện:

- | | |
|----------------------|---------|
| 1. Châu Xuân Tuấn | 1712868 |
| 2. Nguyễn Thọ Tuấn | 1712878 |
| 3. Trương Trọng Tuấn | 1712880 |
| 4. Nguyễn Thanh Tùng | 1712882 |
| 5. Võ Nhật Tường | 1712887 |

I. Mô tả về cách thức cài đặt các hàm quan trọng.

1. Hàm Date:

- Đặt hàm theo C/C++:
 - + `char* Date (int day, int month, int year, char* TIME)`
- Ý tưởng:
 - + Lần lượt load thứ tự các bit từ các tham số day, month, year với giữa là các dấu '/' và lưu vào
 - + mảng TIME: được tạo ở trên. Kết quả của space là DD/MM/YYYY
- Mô tả:
 - + Xuất chuỗi TIME theo định dạng mặc định DD/MM/YYYY
- Tham số:
 - + day, month, year: là các biến lưu giá trị ngày tháng năm
 - + TIME: trỏ đến vùng nhớ lưu trữ kết quả chuỗi ngày tháng đã định dạng
- Giá trị trả về:
 - + Trả về giá trị mà biến TIME đang giữ

2. Hàm Convert:

- Đặt hàm theo C/C++:
 - + `char* Convert (char* TIME, char type)`
- Ý tưởng:
 - + Sử dụng các hàm phụ trợ đã viết là strlen, strcpy và nameEachMonth. Với định dạng MM/DD/YYYY có thể sử dụng load lần lượt từ tháng, ngày và năm, cuối cùng theo dấu ngoặc. Ta cũng có thể dùng strlen và strcpy để sao chép từng nhóm tháng, ngày và năm. Với định dạng Month DD, YYYY; để có Month ta phải xác định tháng và từ đó dịch 4byte trên mảng 2 chiều đã tạo sẵn thông qua hàm nameEachMonth trả về tên của từng tháng, còn lại ta cứ sử dụng strlen và strcpy để sao chép ngày và năm; thêm các ký tự 'khoảng trắng' và ',' khi cần thiết. Với định dạng DD Month, YYYY thì cũng tương tự như định dạng trên nhưng theo thứ tự khác.
- Mô tả:
 - + Chuyển đổi kiểu định dạng của chuỗi TIME thành một trong các định dạng:
MM/DD/YYYY
Month DD, YYYY
DD Month, YYYY
- Tham số:
 - + TIME: trỏ đến vùng nhớ lưu giá trị chuỗi ngày tháng cần chuyển định dạng
 - + type: kiểu byte, định dạng muốn chuyển
 - type = 'A': định dạng MM/DD/YYYY
 - type = 'B': định dạng Month DD, YYYY
 - type = 'C': định dạng DD Month, YYYY
- Giá trị trả về:

- + Trả về định dạng TIME cần in ra
- Các hàm phụ trợ trong Convert:
 - + Strlen: độ dài chuỗi với tham số là địa chỉ mảng của chuỗi đó
 - + Strcpy: sao chép chuỗi với số ký tự cho trước với tham số là chuỗi nguồn, chuỗi đích, số ký tự
 - + NameEachMonth: trả về tên của tháng

3. Day

- Đặt hàm theo C/C++:
 - + int Day (char* TIME)
- Ý tưởng:
 - + Load 2 byte đầu của chuỗi TIME chuyển thành số (lấy mã ascii -48)
- Mô tả:
 - + Lấy giá trị ngày từ chuỗi TIME
- Tham số:
 - + TIME: trỏ đến vùng nhớ lưu giá trị ngày tháng cần xử lý
- Giá trị trả về:
 - + Trả về giá trị ngày trong chuỗi TIME (1 - 31)

4. Month

- Đặt hàm theo C/C++:
 - + int Month(char* TIME)
- Ý tưởng:
 - + Load các bit ở phần MM của biến TIME và chuyển sang số
- Mô tả:
 - + Lấy giá trị tháng từ chuỗi TIME
- Tham số:
 - + TIME: trỏ đến vùng nhớ lưu giá trị ngày tháng cần xử lý
- Giá trị trả về:
 - + Trả về giá trị tháng trong chuỗi TIME (1 - 12)

5. Year

- Đặt hàm theo C/C++:
 - + int Year (char* TIME)
- Ý tưởng:
 - + Load các bit ở phần yyyy (4 bit cuối) của biến TIME và chuyển sang số
- Mô tả:
 - + Lấy giá trị năm từ chuỗi TIME
- Tham số:
 - + TIME: trỏ đến vùng nhớ lưu giá trị ngày tháng cần xử lý
- Giá trị trả về:
 - + Trả về giá trị năm trong chuỗi TIME

6. LeapYear

- Đặt hàm theo C/C++:
 - + bool LeapYear (int year)
- Ý tưởng:
 - + Theo lịch Gregorius - loại lịch tiêu chuẩn hiện nay được dùng trên hầu khắp thế giới thì những năm nào chia hết cho 4 được và không chia hết cho 100 được coi là năm nhuận. Những năm chia hết cho 100 chỉ được coi là năm nhuận nếu chúng cũng chia hết cho 400
 - + Bước 1: Kiểm tra năm có chia hết cho 400 không. Nếu có trả về True, ngược lại False.
 - + Bước 2: Kiểm tra năm có chia hết cho 4 không. Nếu có tiếp tục bước 3, ngược lại trả về False.
 - + Bước 3: Kiểm tra năm có chia hết cho 100 không. Nếu có trả về False, ngược lại True.
- Mô tả:
 - + Kiểm tra năm nhuận
- Tham số:
 - + TIME: trỏ đến vùng nhớ lưu giá trị ngày tháng cần xử lý
- Giá trị trả về:
 - + 0: năm trong chuỗi TIME không phải là năm nhuận
 - + 1: năm trong chuỗi TIME là năm nhuận

7. Weekday

- Đặt hàm theo C/C++:
 - + char* Weekday (char* TIME)
- Ý tưởng:
 - + Sử dụng công thức: $(d + m + y + y/4 - y/100 + y/400) \% 7 = \{0, 1, 2, 3, 4, 5, 6\}$, ánh xạ qua tập { Sun, Mon, Tues, Wed, Thurs, Fri, Sat }. VD 0->Sun.
 - + d: ngày hiện tại trong chuỗi TIME.
 - + m: được ánh xạ từ tập tháng $T = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\}$ sang tập $R = \{0, 3, 2, 5, 0, 3, 5, 1, 4, 6, 2, 4\}$. VD: Tháng 1 -> m=0.
 - + y: năm hiện tại trong chuỗi TIME nếu tháng hiện tại trong chuỗi TIME > 3, Ngược lại y = năm hiện tại trong chuỗi TIME -1.
- Mô tả:
 - + Cho biết giá trị ngày trong chuỗi TIME là thứ mấy trong tuần
- Tham số:
 - + TIME: trỏ đến vùng nhớ lưu giá trị ngày tháng cần xử lý
- Giá trị trả về:
 - + Trả về thứ trong tuần thuộc tập giá trị sau: { Mon, Tues, Wed, Thurs, Fri, Sat, Sun }
- Các hàm phụ trợ trong Weekday:
 - + Day -> Lấy ngày hiện tại trong chuỗi TIME
 - + Month -> Lấy tháng hiện tại trong chuỗi TIME
 - + Year -> Lấy năm hiện tại trong chuỗi TIME

8. CountDay

- Đặt hàm theo C/C++:
 - + int CountDay (char* TIME)
- Ý tưởng:
 - + Dùng phương pháp cộng dồn để đếm năm sau đó đếm tháng và cộng với Day của chuỗi TIME
- Mô tả:
 - + Cho biết ngày vừa nhập là ngày thứ mấy kể từ ngày 1/1/1
- Tham số:
 - + TIME: trỏ đến vùng nhớ lưu giá trị ngày tháng cần xử lý
- Giá trị trả về:
 - + Trả về khoảng thời gian (ngày) từ TIME kể từ 1/1/1
- Các hàm phụ trợ trong Weekday:
 - + Day
 - + Month
 - + Year

9. CanChi

- Ý tưởng:
 - + Lưu tất cả các can và chi thành 2 mảng 1 chiều char rồi theo công thức để lb và sb thích hợp đưa ra kết quả.
- Mô tả:
 - Cho biết can chi của năm trong TIME
- Tham số:
 - TIME: trỏ đến vùng nhớ lưu giá trị ngày tháng cần xử lý
- Giá trị trả về:
 - Trả về giá trị cantri ứng với TIME
 - + can: {"Giap", "At", "Binh", "Dinh", "Mau", "Ky", "Canh", "Tan", "Nham", "Quy"}
 - + chi: {"Ty", "Suu", "Dan", "Meo", "Thin", "Ty", "Ngo", "Mao", "Thân", "Dau", "Tuat", "Hoi"}

10. DayDiff

- Đặt hàm theo C/C++:
 - + int DayDiff (char* TIME1, char *TIME2)
- Ý tưởng:
 - + Sử dụng lại hàm CountDay tính TIME1 và TIME2 . Sau đó trừ ra kết quả.
- Mô tả:
 - + Tính khoảng thời gian cách biệt giữa giá trị năm của chuỗi TIME_1 và TIME_2
- Tham số:
 - + TIME_1 và TIME_2: trỏ đến vùng nhớ lưu giá trị ngày tháng cần xử lý
- Giá trị trả về:
 - + Số ngày cách biệt giữa TIME_1 và TIME_2 (≥ 0)

11. PreNextLeapYear

- Đặt hàm theo C/C++:
 - + `int* PreNextLeapYear(char* TIME)`
- Ý tưởng:
 - + Lấy năm hiện tại trong chuỗi TIME
 - + Chạy lùi về trước năm hiện tại, để tìm một năm nhuận bên trái-> break
 - + Chạy tiến trước năm hiện tại, để tìm một năm nhuận bên phải -> break
- Mô tả:
 - + Cho biết 2 năm nhuận gần nhất so với TIME
- Tham số:
 - + TIME: trỏ đến vùng nhớ lưu giá trị ngày tháng cần xử lý
- Giá trị trả về:
 - + Xuất ra 2 năm nhuận gần nhất với TIME
- Các hàm phụ trợ trong hàm PreNextLeapYear:
 - + LeapYear: kiểm tra xem có là năm nhuận hay không

12. ReadFile

- Đặt hàm theo C/C++:
 - + `char* ReadFile (char* infile)`
- Ý tưởng:
 - + Đọc file lấy vào tên file và chuỗi trả về nội dung trong lưu trong file
- Mô tả:
 - + Mở file và lưu file descriptor vào thanh ghi của file input, tiếp theo lưu dữ liệu trong file vào một chuỗi, cuối cùng đóng file trả file descriptor vào thanh \$a0 và trả về nội dung của file
- Tham số:
 - + infile: biến lưu file input
- Giá trị trả về:
 - + Chuỗi lưu nội dung của file

13. ProcessFile

- Đặt hàm theo C/C++:
 - + `void ProcessFile (char* buffer)`
- Ý tưởng:
 - + Xử lý file là tách chuỗi đo thành dữ liệu lưu địa chỉ vào các biến
- Mô tả:
 - + Trong phần xử lý file, đầu tiên là lấy địa chỉ của nội dung file từ đọc file, sau đó tiến hành xử lý chuỗi đó bằng việc thêm ký tự '/', copy chuỗi vào 2 biến TIME1 và TIME2.
- Tham số:
 - + buffer: lưu nội dung của chuỗi
- Giá trị trả về:

- + Lưu thời gian hai ngày vào TIME1 và TIME2

14. WriteFile

- Đặt hàm theo C/C++:
 - + void WriteFile (char* outfile, char* TIME1, char* TIME2)
- Ý tưởng:
 - + Lấy các tham số thông qua các biến, thực hiện các yêu cầu và xuất ra từng nội dung
- Mô tả:
 - + Viết vào file cũng mở file và đóng file giống thủ tục của phần đọc file, ta lấy tên file qua tham số và load các biến vào trong thanh ghi, tiếp đến xử lý file và xuất theo yêu cầu.
- Tham số:
 - + outfile: biến lưu file output
 - + TIME1: biến lưu thông tin ngày thứ nhất
 - + TIME2: biến lưu thông tin ngày thứ hai

II. Giải thích về qui tắc khi viết và gọi hàm trong MIPS.

- Qui tắc khi viết hàm:
 - + Đầu tiên khi vào hàm là ta phải xác định lưu những thanh ghi cần thiết, tập trung vào thanh lưu địa chỉ trả về của hàm \$ra, các thanh ghi tham số \$a0,.. \$a3 nếu thay đổi, các thanh lưu giá trị từ \$s0... \$s7 và các thanh \$t0 ... \$t8 nếu cần thiết.
 - + Tạo stack lưu các thanh ghi đó.
 - + Sử dụng các phép toán logic và luận lý để thực hiện chức năng của hàm. Xuất kết quả trả về vào \$v0,\$v1.
 - + Load lại các thanh ghi đã lưu ở trên đồng thời hủy stack.
- Qui tắc khi đọc hàm:
 - + Load các địa chỉ biến cần đưa vào hàm vào các thanh ghi tham số.

-HẾT-