

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN
ୱୱୱୱ ୱୱୱୱ

ĐỒ ÁN
KIẾN TRÚC MÁY TÍNH VÀ HỢP NGỮ

THƯ VIỆN THỜI GIAN

Học kỳ II

Năm học: 2017-2018

LỜI CẢM ƠN

Trong quá trình thực hiện đồ án đề tài, nhóm em đã nhận được rất nhiều sự giúp đỡ, hỗ trợ từ các thầy cô Trường Đại học Khoa học Tự nhiên – ĐHQG TP.HCM và các bạn bè trong trường. Nhóm em xin bày tỏ lòng cảm ơn chân thành về sự hướng dẫn, chỉ bảo của mọi người.

Đặc biệt, nhóm em xin bày tỏ lòng biết ơn sâu sắc đến các thầy cô khoa Công nghệ thông tin, cụ thể hơn chúng em xin cảm ơn thầy Phạm Tuấn Sơn đã hỗ trợ, giảng dạy rất kĩ lưỡng từng phần nhỏ để chúng em có một đồ án thật hoàn chỉnh và phù hợp nhất. Chính những thứ tưởng chừng như nhỏ nhặt này đã góp phần to lớn giúp chúng em hoàn thành đồ án và bảng báo cáo này thật hoàn chỉnh nhất. Một lần nữa, chúng em xin bày tỏ lòng biết ơn sâu sắc đến với các thầy cô và bạn bè.

Ngoài ra, nhóm chúng em còn nhận được rất nhiều sự khích lệ tinh thần, động viên cổ vũ không chỉ từ các bạn đồng trang lứa mà còn đến từ các anh chị khoa Công nghệ thông tin khóa trước và các thầy cô.

Chúng em xin chân thành cảm ơn!

Mục lục

Mục lục.....	2
1 CHƯƠNG I: MỞ ĐẦU	3
1.1 Giới thiệu nhóm và phân công công việc.....	3
1.2 Mô tả đồ án.....	3
2 CHƯƠNG II: GIỚI THIỆU ĐỒ ÁN	3
2.1 Thời gian thực hiện	3
2.2 Các bước thực hiện đồ án	4
3 CHƯƠNG III: NỘI DUNG ĐỒ ÁN.....	4
3.1 char* Date(int day, int month, int year)	4
3.2 char* Convert(char* TIME, char type)	5
3.3 int Day(char * TIME).....	5
3.4 int Month(char * TIME).....	5
3.5 int Year(char* TIME).....	6
3.6 int LeapYear(char* TIME).....	6
3.7 int GetTime(char* TIME_1, char* TIME_2)	6
3.8 char* Weekday(char* TIME).....	6
3.9. NeareastLeapYears(char* TIME).....	7
4 TÀI LIỆU THAM KHẢO	7

1 CHƯƠNG I: MỞ ĐẦU

1.1 Giới thiệu nhóm và phân công công việc

- Số thành viên: 3 người

STT	MSSV	Họ và tên
1	1612840	Dương Nguyễn Thái Bảo
2	1612052	Phạm Minh Chiến
3	1612899	Hoàng Xuân Trường

1.2 Mô tả đồ án

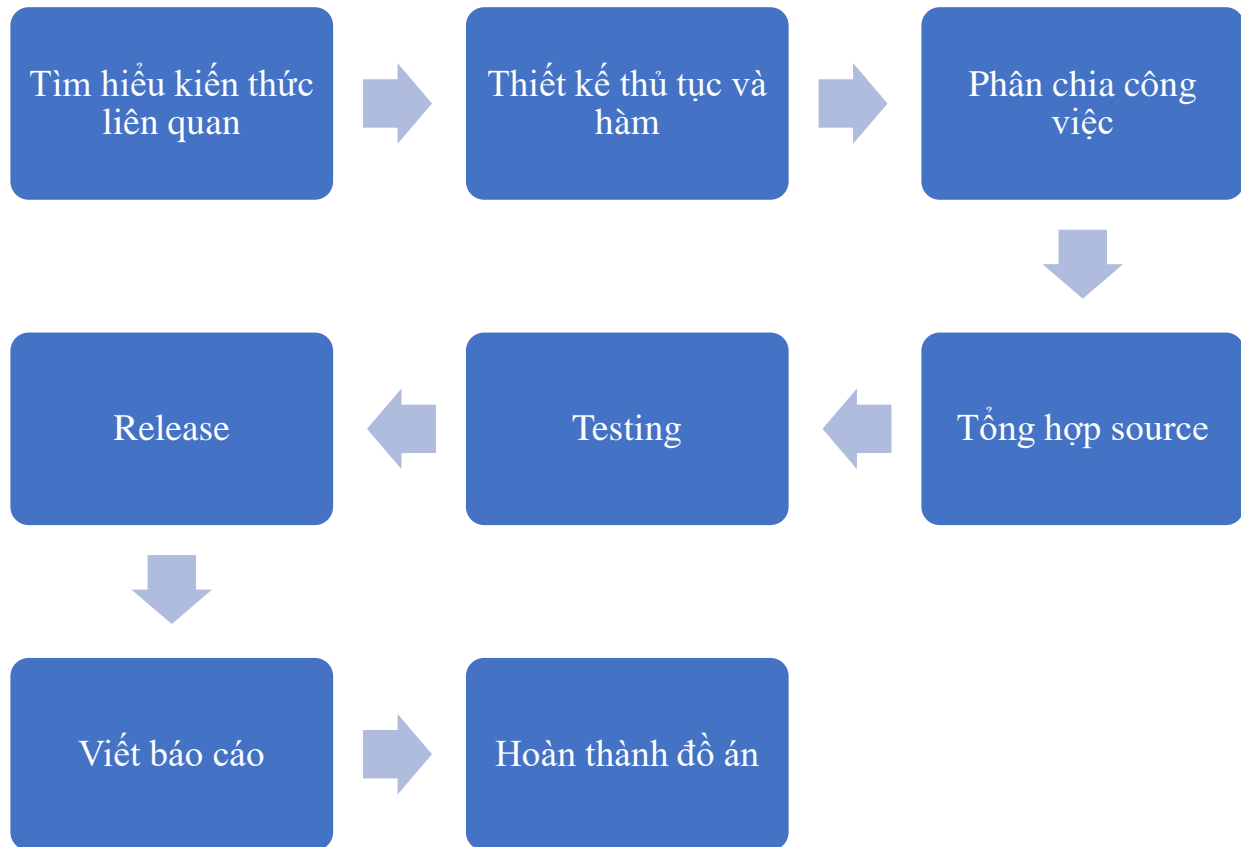
- Biểu diễn thời gian trên máy tính xử lý trong MIPS
- Viết chương trình cho phép người dùng nhập vào ngày tháng năm (kiểm tra định dạng đúng theo dd/mm/yyyy). Yêu cầu chương trình chuyển đổi các định dạng ngày tháng năm, xác định được năm nhuận, thứ trong tuần,...

2 CHƯƠNG II: GIỚI THIỆU ĐỒ ÁN

2.1 Thời gian thực hiện

- Đồ án bắt đầu từ ngày 1/4 đến ngày 15/4

2.2 Các bước thực hiện đồ án



3 CHƯƠNG III: NỘI DUNG ĐỒ ÁN

3.1 `char* Date(int day, int month, int year)`

- Xuất chuỗi TIME theo định dạng mặc định DD/MM/YYYY
- Mô tả: Người dùng nhập ngày tháng năm theo định dạng DD/MM/YYYY, sau khi kiểm tra dữ liệu nhập hợp lệ thì xuất ra màn hình.
 - Dữ liệu hợp lệ là ngày trong khoảng 1-31 tùy thuộc vào từng tháng trong năm (kể cả số ngày trong tháng 2 của năm nhuận), năm lớn hơn hoặc bằng 1900 và phải là số nguyên.

3.2 char* Convert(char* TIME, char type)

- Chuyển đổi kiểu dữ liệu của chuỗi TIME theo 2 lựa chọn:
 - A. Định dạng MM/DD/YYYY
 - B. Định dạng Month DD, YYYY
 - C. Định dạng DD Month, YYYY
- Tham số truyền vào: \$a0 (char* TIME), \$a1 (Type: A, B, C – đúng định dạng)
- Mô tả:
 - Chuỗi TIME sau khi người dùng nhập vào (đã kiểm tra tính hợp lệ) được ngày, tháng, năm lưu vào lần lượt các thanh ghi khác nhau \$a0, \$a1, \$a2.
 - Tùy vào từng lựa chọn của người dùng (A, B hoặc C) chương trình sẽ xuất ra kết quả tương ứng.
- Lưu ý: Month được thể hiện qua 3 ký tự đầu. Ví dụ: Jan – January, Jun - June

3.3 int Day(char * TIME)

- Tham số truyền vào: \$a0
- Mô tả: Sử dụng \$t0 để lưu kết quả, giá trị trả về là một số được lưu trong \$v0. Hàm trả về kết quả là giá trị ngày trong chuỗi char* TIME
- Cách thực hiện:
 - Đọc từng byte của \$a0 và lưu vào \$t2
 - Chuyển ký tự \$t2 thành số
 - $\$t0 = \$t0 * 10 + \$t2$
 - Lặp tới khi gặp ký tự '/'

3.4 int Month(char * TIME)

- Tham số truyền vào: \$a0
- Mô tả: hàm trả về kết quả là giá trị tháng trong chuỗi char* TIME (dd/mm/yyyy)
- Cách thực hiện:
 - Chạy đến ký tự '/' đầu tiên
 - Đọc từng byte của \$a0 và lưu vào \$t2
 - Chuyển ký tự \$t2 thành số
 - $\$t0 = \$t0 * 10 + \$t2$
 - Lặp tới khi gặp ký tự '/' thứ hai

3.5 int Year(char* TIME)

- Tham số truyền vào: \$a0
- Mô tả: hàm trả về kết quả là giá trị năm trong chuỗi char* TIME (dd/mm/yyyy)
- Cách thực hiện:
 - Chạy đến ký tự '/' thứ 2
 - Đọc từng byte của \$a0 và lưu vào \$t2
 - Chuyển ký tự \$t2 thành số
 - $\$t0 = \$t0 * 10 + \$t2$
 - Lặp tới khi gặp ký tự kết thúc chuỗi

3.6 int LeapYear(char* TIME)

- Tham số: \$a0: chuỗi time
- Giá trị trả về: \$v0: 1 nếu năm trong chuỗi time là năm nhuận, ngược lại 0.
- Thuật toán:
 - Đầu tiên gọi hàm Year để tách lấy năm trong chuỗi time.
 - Sử dụng hàm trung gian is_leap nhận tham số là năm và kiểm tra nó có phải năm nhuận không:
 - Nếu năm chia hết cho 400 thì nó là nhuận.
 - Ngược lại, nếu năm không chia hết cho 4 thì nó không nhuận.
 - Ngược lại, nếu năm không chia hết cho 100 thì nó là nhuận.

3.7 int GetTime(char* TIME_1, char* TIME_2)

- Tham số: \$a0: chuỗi time1, \$a1: chuỗi time2.
- Giá trị trả về: \$v0: khoảng cách giữa 2 năm trong 2 chuỗi.
- Thuật toán:
 - Đầu tiên gọi hàm Year để tách lấy năm trong 2 chuỗi time.
 - Trừ 2 năm này. Nếu số này âm thì lấy 0 trừ đi chính nó.

3.8 char* Weekday(char* TIME)

- Tham số: \$a0: chuỗi time.
- Giá trị trả về: \$v0: con trỏ tới chuỗi thuộc một trong các kết quả {Sun, Mon, Tue, Wed, Thurs, Fri, Sat}.
- Thuật toán:
 - Tách ngày, tháng, năm từ chuỗi time bằng hàm Day, Month, Year.
 - Công thức: $(d + m + y + [y / 4] + c) \% 7$. Với:
 - d: ngày.
 - m: tháng tương ứng trong bảng Month ở dưới.
 - y: 2 số cuối của năm.

- c: thế kỉ.
- Kết quả của công thức trên sẽ cho ra giá trị số thuộc tập {0,1,2,3,4, 5, 6} tương ứng các ngày trong tuần {Sun, Mon, Tues, Wed, Thurs, Fri, Sat}.

Month	m	Leap years
January	0	6
February	3	2
March	3	
April	6	
May	1	
June	4	
July	6	
August	2	
September	5	
October	0	
November	3	
December	5	

3.9. NeareastLeapYears(char* TIME)

- Tham số đầu vào: \$a0: chuỗi time.
- Kết quả: \$v0: năm nhuận gần nhất với năm trong chuỗi time, \$v1: năm nhuận gần nhì với năm trong chuỗi time.
- Thuật toán:
 - Tách năm trong chuỗi time bằng hàm Year rồi gán cho \$s0 và \$s1. Khởi tạo biến đếm số năm nhuận đã tìm thấy.
 - Lặp trong khi chưa tìm đủ 2 năm nhuận gần nhất: \$s0 giảm đi 1, \$s1 tăng lên 1. Nếu \$s0 là năm nhuận thì lưu lại và tăng biến đếm lên, tương tự với \$s1.

4 TÀI LIỆU THAM KHẢO

- [1] <http://www.scribd.com/doc/3577342/MIPS-Assembly-Language-Programming>
- MIPS Assembly Language Programming, Robert Britton.