

**ĐẠI HỌC QUỐC GIA TP HCM
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN**

-----❧❧❧-----



BÁO CÁO

ĐỒ ÁN: THƯ VIỆN TIME VỚI MIPS

Lớp: 17CTT5

Thông tin thành viên nhóm:

1. Đỗ Tấn Tài	MSSV: 1712737
2. Huỳnh Công Sinh	MSSV: 1712724
3. Trần Thiên Quàng	MSSV: 1712709
4. Nguyễn Văn Thìn	MSSV: 1712787
5. Phan Đại Hải	MSSV: 1712410

GVHD: ThS. Lê Viết Long

Hồ Chí Minh, Tháng 04-2019

**ĐẠI HỌC QUỐC GIA TP HCM
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN**

-----❧❧❧-----



BÁO CÁO

ĐỒ ÁN: THƯ VIỆN TIME VỚI MIPS

Lớp: 17CTT5

Thông tin thành viên nhóm:

- | | |
|---------------------|---------------|
| 1. Đỗ Tấn Tài | MSSV: 1712737 |
| 2. Huỳnh Công Sinh | MSSV: 1712724 |
| 3. Trần Thiên Quàng | MSSV: 1712709 |
| 4. Nguyễn Văn Thìn | MSSV: 1712787 |
| 5. Phan Đại Hải | MSSV: 1712410 |

GVHD: ThS. Lê Viết Long

Hồ Chí Minh, Tháng 04-2019

MỤC LỤC

MỤC LỤC	1
NỘI DUNG.....	1
I. BẢNG TÓM TẮT CÁC HÀM SỬ DỤNG TRONG CHƯƠNG TRÌNH	1
II. CÁCH THỨC VÀ Ý TƯỞNG CÀI ĐẶT CÁC HÀM CHỨC NĂNG	5
1. Hàm DATE	5
2. Hàm CONVERT	5
3. Hàm DAY	6
4. Hàm MONTH.....	6
5. Hàm YEAR.....	6
6. Hàm ONLY_YEAR_IS_LEAP	6
7. Hàm LEAPYEAR	6
8. Hàm ATOI.....	6
9. Hàm ATOI_WHOLE	7
10. Hàm CHECK_VALID	7
11. Hàm GETTIME	7
12. Hàm INPUT_TIME	7
13. Hàm IS_ONLY_DIGITS	7
14. Hàm COUNT_LEAPYEAR.....	7
15. Hàm STRCPY	8
16. Hàm STRCAT	8
17. Hàm MONTH_IN_STRING.....	8
18. Hàm ABSOLUTE_DAY	8
19. Hàm DEFINE_DAY	8
20. Hàm LEAP_YEAR_CLOSER.....	8
21. Hàm CANCHI	8
22. Hàm READ_FILE	8
23. Hàm CONVERT_SPACE.....	9
24. Hàm WRITE_FILE	9
25. Hàm COUNT	9

26. Hàm ITOA	9
27. Hàm STRREV	9
28. Hàm CHECK_SYNTAX.....	9
29. Hàm CHECK_DATE_IN_FILE.....	9
III. QUI TẮC KHI VIẾT VÀ GỌI HÀM TRONG MIPS	10
1. MIPS hỗ trợ một số thanh ghi để lưu trữ các dữ liệu phục vụ cho thủ tục:	10
2. Cấu trúc của một thủ tục:	10
IV. ĐÁNH GIÁ MỨC ĐỘ HOÀN THÀNH: 100%	10
DANH MỤC TÀI LIỆU THAM KHẢO.....	11
BIÊN BẢN PHÂN CÔNG VIỆC NHÓM	12
I. THÔNG TIN CHUNG	12
II. BẢNG PHÂN CÔNG CÔNG VIỆC	12

NỘI DUNG

I. BẢNG TÓM TẮT CÁC HÀM SỬ DỤNG TRONG CHƯƠNG TRÌNH

Tên hàm	Chức năng	Tham số	Kết quả trả về
1. Date	Hàm trả về chuỗi TIME theo định dạng DD/MM/YYYY	\$a0 DAY (int) \$a1 MONTH (int) \$a2 YEAR (int) \$a3 địa chỉ TIME	\$v0 chứa địa chỉ chuỗi TIME
2. Convert	Hàm trả về chuỗi TIME với định dạng do người dùng quy định	\$a0 TIME \$a1 type	\$v0 chứa địa chỉ chuỗi TIME
3. Day	Hàm trả về ngày trong chuỗi TIME	\$a0 TIME	\$v0: day(int)
4. Month	Hàm trả về tháng trong chuỗi TIME	\$a0 TIME	\$v0: month(int)
5. Year	Hàm trả về năm trong chuỗi TIME	\$a0 TIME	\$v0: year(int)
6. only_year_is_leap	Hàm kiểm tra năm nhuận	\$a0 year(int)	\$v0: 1 - năm nhuận, 0 - năm không nhuận
7. LeapYear	Hàm kiểm tra năm nhuận	\$a0 TIME	\$v0: 1 - năm nhuận, 0 - năm không nhuận

Tên hàm	Chức năng	Tham số	Kết quả trả về
8. atoi	Chuyển từ kí tự thứ x đến kí tự thứ y thành số	\$a0 Chuỗi \$a1 Vị trí bắt đầu \$a2 Vị trí kết thúc	\$v0: int
9. atoi_whole	Chuyển chuỗi thành số	\$a0 Chuỗi	\$v0: int
10. check_valid	Hàm kiểm tra tính hợp lệ logic của TIME	\$a0 TIME	\$v0: 1 – hợp lệ, 0 – không hợp lệ
11. GetTime	Hàm trả về khoảng cách giữa hai ngày, kết quả trả về là số ngày.	\$a0 TIME_1 \$a1 TIME_2	\$v0: number of day (int)
12. input_time	Hàm nhập TIME	\$a0 TIME \$a1 Chuỗi tạm	\$v0: Địa chỉ TIME \$v1: Tính hợp lệ
13. is_only_digits	Hàm kiểm tra một chuỗi có toàn các chữ số hay không	\$a0 string	\$v0: Tính hợp lệ
14. count_leapYear	Hàm đếm số năm nhuận từ năm 0 đến năm truyền vào	\$a0: Năm hiện tại \$a1: Tháng hiện tại	\$v0: Số năm (int)

Tên hàm	Chức năng	Tham số	Kết quả trả về
15. strcpy	Sao chép chuỗi y vào chuỗi x	\$a0: Địa chỉ string x \$a1: Địa chỉ string y	\$v0: Địa chỉ string x
16. strcat	Nối chuỗi y vào chuỗi x	\$a0: Địa chỉ string x \$a1: Địa chỉ string y	\$v0: Địa chỉ string x
17. Month_in_String	Hàm trả về tên tháng	\$a0: month (integer)	\$v0 month (string)
18. absolute_day	Hàm đếm số ngày tuyệt đối	\$a0 năm (int) \$a1 tháng (int) \$a2 ngày (int)	\$v0: số ngày tuyệt đối (int)
19. define_day	Hàm xác định một ngày là thứ mấy	\$a0 TIME	\$v0: Địa chỉ chuỗi thông báo thứ tương ứng
20. leap_year_closer	Hàm trả về hai năm nhuận gần nhất	\$a0 TIME	\$v0: Năm nhuận 1 \$v1: Năm nhuận 2
21. CanChi	Tính can chi của một năm	\$a0: TIME	\$v0: Địa chỉ chuỗi biểu diễn can \$v1: Địa chỉ chuỗi biểu diễn chi

Tên hàm	Chức năng	Tham số	Kết quả trả về
22. read_file	Đọc chuỗi trong file và lưu vào TIME1 và TIME2	\$a0: TIME1 \$a1: TIME2 \$a2: buffer	\$v0: Địa chỉ TIME1 \$v1: Địa chỉ TIME2
23. convert_space	Chuyển chuỗi có dạng “DD MM YYYY” thành “DD/MM/YYYY”	\$a0: TIME	\$v0: Địa chỉ TIME1
24. write_file	Ghi kết quả ra file output	\$a0: TIME1 \$a1: TIME2	\$v0: file descriptor
25. count	Trả về độ dài chuỗi truyền vào	\$a0: Địa chỉ string	\$v0: int
26. ItoA	Chuyển số thành chuỗi và đảo ngược chuỗi	\$a0: int \$a1: Địa chỉ lưu chuỗi	\$v0: Địa chỉ lưu chuỗi
27. strrev	Đảo ngược chuỗi	\$a0: Địa chỉ chuỗi \$a1: Địa chỉ chuỗi kết quả sau đảo ngược	\$v0: Địa chỉ chuỗi kết quả sau đảo ngược
28. check_syntax	Kiểm tra chuỗi có dạng “XX/XX/XXXX” với X là kí tự số hay không	\$a0: TIME	\$v0: 1 - hope, 0 - khonghope
29. check_date_in_file	Kiểm tra chuỗi trong file input có hợp lệ syntax và logic hay không	\$a0: TIME1 \$a1: TIME2	\$v0: 1 - hope, 0 - khonghope

II. CÁCH THỨC VÀ Ý TƯỞNG CÀI ĐẶT CÁC HÀM CHỨC NĂNG

1. Hàm DATE

Lấy ngày từ thanh ghi \$a0. Chia cho 10, lấy phần dư lưu vào \$t3 và phần nguyên lưu vào trong \$t2. Tăng giá trị của \$t2, \$t3 thêm 48 để hiển thị số lấy từ ngày chuyển sang dạng kí tự số. Rồi lưu \$t2, \$t3, kí tự “/” lần lượt vào các ô 0,1,2 của thanh ghi \$a3(dùng để lưu TIME)

Lấy tháng từ thanh ghi \$a1. Chia cho 10, lấy phần dư lưu vào \$t3 và phần nguyên lưu vào trong \$t2. Tăng giá trị của \$t2, \$t3 thêm 48 để hiển thị số lấy từ ngày chuyển sang dạng kí tự số. Rồi lưu \$t2, \$t3, kí tự “/” lần lượt vào các ô 3,4,5 của thanh ghi \$a3(dùng để lưu TIME)

Lấy năm từ thanh ghi \$a2. Chia cho 1000, lấy phần nguyên lưu vào \$t2 và tăng \$s2 thêm 48 chuyển sang dạng kí tự số rồi lưu vào ô thứ 6 của thanh ghi \$a3. Lấy phần dư chia cho 100, lấy phần nguyên lưu vào \$t2 và tăng \$s2 thêm 48 chuyển sang dạng kí tự số rồi lưu vào ô thứ 7 của thanh ghi \$a3. Lấy phần dư chia cho 10, lấy phần nguyên lưu vào \$t2 và tăng \$s2 thêm 48 chuyển sang dạng kí tự số rồi lưu vào ô thứ 7 của thanh ghi \$a3. Lấy phần dư lưu vào \$t0 và tăng \$s0 thêm 48 chuyển sang dạng kí tự số rồi lưu vào ô thứ 9 của thanh ghi \$a3.

Thuật toán chung: Chuyển các số từ các thanh ghi ngày, tháng, năm chuyển sang kí tự số lưu vào TIME.

2. Hàm CONVERT

Sử dụng thanh ghi \$a1 để xác định kiểu định dạng mà người dùng cần chuyển đổi.

Với Convert_A: lấy các kí tự ngày ô 0,1 của \$a0 (lưu TIME dạng DD/MM/YYYY) lưu tạm vào thanh ghi tạm \$t0,\$1, lấy các kí tự tháng ô 3,4 lưu tạm vào thanh ghi tạm \$t2,\$3. Rồi sau đó lấy giá trị của thanh ghi tạm \$t2,\$3 lưu lại vào kí tự ngày ô 0,1 của \$a0, lấy giá trị của thanh ghi tạm \$t0,\$1 lưu lại vào kí tự ngày ô 3,4 của \$a0. Kiểu giá trị TIME \$a0 trả về có dạng MM/DD/YYYY.

Với Convert_B: từ \$a0 lưu TIME dưới dạng DD/MM/YYYY, ta copy “DD” from \$a0 sang TEMP_1 dưới dạng “_D1D2_”, copy “YYYY” sang TEMP_2 dưới dạng “Y1Y2Y3Y4”. Gọi hàm strcpy để lấy chuỗi tháng rồi trả về \$a1, sau đó nối chuỗi trong \$a1 với TEMP_1 và TEMP_2 lần lượt bởi hàm strcat. Trả về kiểu TIME dạng Month DD, YYYY.

Với Convert_C: Tương tự như Convert_B, ta copy “DD” từ \$a0 lưu TIME dưới dạng DD/MM/YYYY sang TEMP_1 dưới dạng “D1D2_”, copy “YYYY” sang TEMP_2 dưới dạng “_Y1Y2Y3Y4”. Ta copy “D1D2_” trong TEMP_1 vào TIME \$a1, nối với month, TEMP_2 lần lượt bởi hàm strcat. Kiểu giá trị TIME trả về có dạng DD Month, YYYY.

3. Hàm DAY

Gán \$a1 =1, \$a2 =2 để truyền tham số vào hàm atoi để biến kiểu ngày trong TIME dạng DD/MM/YYYY chuyển sang kiểu số với các kí tự số trong TIME được biến đổi từ vị trí 1 tới 2 (DD). Kết quả trả về là \$v0 là giá trị số của ngày trong TIME.

4. Hàm MONTH

Gán \$a1 =3, \$a2 =4 để truyền tham số vào hàm atoi để biến kiểu ngày trong TIME dạng DD/MM/YYYY chuyển sang kiểu số với các kí tự số trong TIME được biến đổi từ vị trí 3 tới 4 (MM). Kết quả trả về là \$v0 là giá trị số của tháng trong TIME.

5. Hàm YEAR

Gán \$a1 =6, \$a2 =9 để truyền tham số vào hàm atoi để biến kiểu ngày trong TIME dạng DD/MM/YYYY chuyển sang kiểu số với các kí tự số trong TIME được biến đổi từ vị trí 6 tới 9 (YYYY). Kết quả trả về là \$v0 là giá trị số của năm trong TIME.

6. Hàm ONLY_YEAR_IS_LEAP

Lấy năm chia lấy dư cho 4, lấy phần dư lưu vào \$t2. Nếu \$t2 khác 0, nhảy tới hàm only_year_is_leap_false, nếu \$t2 bằng 0, tiếp tục kiểm tra năm.

Lấy năm chia lấy dư cho 100, lấy phần dư lưu vào \$t2. Nếu \$t2 bằng 0, nhảy tới hàm only_year_is_leap_false, nếu \$t2 khác 0, đi tới hàm only_year_is_leap_true.

Trong hàm only_year_is_leap_true, cho \$v0 giá trị 1 và nhảy tới hàm only_year_is_leap_exit.

Trong hàm only_year_is_leap_false, cho \$v0 giá trị 0 và đi tới hàm only_year_is_leap_exit.

Thuật toán chung: Kiểm tra năm nhuận là năm chia hết cho 400 hoặc là năm chia hết cho 4 nhưng không chia hết cho 100.

7. Hàm LEAPYEAR

Từ \$a0 lưu input là TIME, nhảy và lưu địa chỉ tới hàm Year để lấy giá trị của năm và lưu trong \$v0.

Gán giá trị của năm từ \$v0 sang \$a0.

Tiếp tục nhảy và lưu địa chỉ tới hàm only_year_is_leap để kiểm tra năm nhuận, kết quả trả về lưu trong \$v0.

8. Hàm ATOI

Gán \$v0 bằng 0 để chứa kết quả. Khởi tạo \$t0, \$t1 là địa chỉ của vị trí bắt đầu và kết thúc chuyển.

Kiểm tra \$t0 < \$t1, nếu không hợp lệ thì nhảy tới hàm atoi_exit, còn nếu có thì lần lượt lấy byte đầu (kí tự đầu) của \$t0, chuyển sang số rồi lưu vào \$t3, sau đó lấy \$v0 * 10 + \$t3 để lấy kết quả.

Tiếp tục vòng lặp cho tới khi vị trí bắt đầu lớn hơn vị trí kết thúc (\$t0 > \$t1).

Giá trị của \$v0 sau khi thoát vòng lặp là kết quả chuyển của string.

Thuật toán chung: Lấy từng ký tự của string cộng với kết quả nhân 10.

9. Hàm ATOI_WHOLE

Gán \$v0 bằng 0 để chứa kết quả. Khởi tạo \$t0 là địa chỉ bắt đầu của string.

Tương tự như hàm 8, lần lượt lấy từng byte đầu của string, lưu vào \$t1 và cộng vào ($\$v * 10$).

Lặp lại cho tới khi \$t1 bằng 10 (tức là tới ký tự kết thúc chuỗi).

10. Hàm CHECK_VALID

Đầu tiên ta trích xuất tháng từ chuỗi TIME và kiểm tra tính hợp lệ của tháng ($1 \leq \text{tháng} \leq 12$).

Tiếp đó tùy vào tháng trong TIME là tháng nào để ta kiểm tra số ngày có phù hợp tương ứng với tháng hay không bằng cách cho nhảy đến nhãn check_31_day hoặc check_30_day, riêng với tháng 2 phải kiểm tra tiếp xem năm đó có là năm nhuận hay không để có được số ngày trong tháng 2 tương ứng là 28 hay 29 ngày.

11. Hàm GETTIME

Trích xuất riêng ngày tháng năm của 2 chuỗi thời gian và lưu vào các thanh ghi.

Sử dụng hàm ABSOLUTE_DAY để tính số ngày tuyệt đối của 2 TIME với các tham số đã được lưu vào các thanh ghi ở trên.

Trừ hai kết quả nhận được và kiểm tra xem kết quả có dương hay không, nếu âm thì kết quả là số đối của số hiện hành.

12. Hàm INPUT_TIME

Cho nhập từng thành phần ngày tháng năm theo trình tự nhập chuỗi, kiểm tra tính hợp lệ cú pháp của chuỗi vừa nhập, chuyển chuỗi vừa nhập thành số và lưu vào 3 thanh ghi. Nếu cả 3 thành phần ngày tháng năm đều nhập hợp lệ ta sử dụng hàm Date để có chuỗi dạng “DD/MM/YYYY”. Ngược lại, nếu chuỗi nhập không hợp lệ về cú pháp hoặc logic sẽ yêu cầu nhập lại đến khi nhận được chuỗi hợp lệ.

13. Hàm IS_ONLY_DIGITS

Kiểm tra một chuỗi có bao gồm toàn các ký tự số hay không bằng cách duyệt từng ký tự trong chuỗi và sử dụng mã ASCII để kiểm tra tính hợp lệ. Kiểm tra đến khi gặp ký tự kết thúc “\0” hoặc ký tự xuống dòng “\n”.

14. Hàm COUNT_LEAPYEAR

Hàm đếm số năm nhuận từ năm 0 đến năm hiện tại theo công thức được cung cấp. Nếu tháng nhỏ hơn 3 trừ năm hiện tại đi 1. Kết quả sẽ bằng $(\text{nam} / 4 - \text{nam} / 100 + \text{nam} / 400)$

15. Hàm STRCPY

Duyệt từng kí tự của chuỗi cần sao chép bằng cách lb từng kí tự và sb các kí tự đó vào chuỗi đích cho đến khi gặp kí tự kết thúc ở chuỗi cần sao chép thì dừng lại.

16. Hàm STRCAT

Tìm ra vị trí kết thúc của chuỗi cần nối chuỗi vào bằng cách duyệt từng kí tự đến khi gặp kí tự “\0”. Từ vị trí đó ta lb từng kí tự ở chuỗi cần nối và sb vào chuỗi đích.

17. Hàm MONTH_IN_STRING

Với số nguyên đại diện cho tháng là tham số đầu vào, ta sẽ kiểm tra và nhảy đến nhãn tương ứng với tháng truyền vào. Sau đó gán \$v0 là địa chỉ của chuỗi biểu diễn string của tháng tương ứng.

18. Hàm ABSOLUTE_DAY

Áp dụng công thức tính số ngày tuyệt đối để tính với sự hỗ trợ của hàm COUNT_LEAPYEAR.

19. Hàm DEFINE_DAY

Áp dụng công thức để tính được giá trị số biểu diễn thứ, nếu kết quả là 0, cộng thêm 7 vào kết quả. Sau đó xét số nguyên kết quả để nhảy đến nhãn tương ứng. Tại đó, \$v0 sẽ được gán địa chỉ của chuỗi biểu diễn thứ tương ứng.

20. Hàm LEAP_YEAR_CLOSER

Kiểm tra năm hiện tại truyền vào có phải năm nhuận hay không, nếu là năm nhuận, hai năm nhuận gần nhất chính là năm hiện tại cộng trừ 4. Nếu năm hiện tại không phải năm nhuận, ta tăng năm hiện tại lên từng đơn vị và kiểm tra cho đến khi nhận được năm nhuận. Lấy giá trị năm nhu cho đến khi nhận được năm nhuận. Lấy giá trị năm nhuận vừa nhận được -4 và xét xem đó có phải năm nhuận hay không. Nếu không ta lấy giá trị năm nhuận vừa nhận được -8 sẽ nhận được kết quả năm nhuận thứ 2.

21. Hàm CANCHI

Trích xuất năm từ chuỗi TIME truyền vào. Áp dụng công thức để tính can của năm đó, xét kết quả để nhảy đến nhãn tương ứng và gán địa chỉ chuỗi biểu diễn cho \$v0, làm tương tự với chi.

22. Hàm READ_FILE

Mở file, đọc chuỗi vào vùng nhớ buffer, sau đó xử lí chuỗi bằng cách duyệt từng kí tự trong chuỗi buffer và lưu vào vùng nhớ chuỗi TIME1 cho đến khi gặp kí tự “\r”. Từ vị trí đó cộng 1, đọc tiếp từng kí tự trong chuỗi buffer và lưu vào vùng nhớ chuỗi TIME2 cho đến khi gặp kí tự “\0” trong chuỗi buffer thì kết thúc.

23. Hàm CONVERT_SPACE

Duyệt từng kí tự của chuỗi cần chuyển đổi, xét xem nếu đó là kí tự khoảng trắng thì đổi thành kí tự “/”.

24. Hàm WRITE_FILE

Kiểm tra tính hợp lệ TIME1, nếu không hợp lệ thì in thông báo vào file. Sau đó dù TIME1 hợp lệ hay không hợp lệ, ta vẫn tiếp tục kiểm tra tính hợp lệ của TIME2, nếu không hợp lệ thì in thông báo. Sau đó với sự hỗ trợ của hàm CHECK_DATE_IN_FILE, ta sẽ quyết định có in kết quả các yêu cầu hay không dựa vào kết quả kiểm tra lưu trong thanh ghi \$v0. Nếu hợp lệ, ta sẽ in lần lượt các kết quả theo yêu cầu với sự hỗ trợ của các hàm chức năng nêu trên bằng cách truyền các tham số phù hợp cho các hàm đã có.

25. Hàm COUNT

Duyệt từng kí tự trong chuỗi truyền vào và từng biến đếm lên 1 cho đến khi gặp kí tự kết thúc chuỗi.

26. Hàm ITOA

Chuyển số nguyên thành chuỗi bằng cách chia dư số đó cho 10 để lấy ra từng kí tự từ phải sang và chuyển số đó thành kí tự số bằng cách cộng thêm 48. Lặp lại cho đến khi thương bằng 0 có nghĩa là đã hết chữ số. Ta sẽ nhận được số đó đảo ngược ở dạng chuỗi.

27. Hàm STRREV

Sử dụng hàm COUNT để biết số kí tự của chuỗi, ta sẽ biết được kí tự kết thúc của chuỗi kết quả nằm ở vị trí nào, sau đó ta duyệt từng kí tự của chuỗi truyền vào và ghi sb từng kí tự vừa lấy ra vào vị trí vùng nhớ từ vị trí kết thúc của chuỗi kết quả ngược về trước cho đến khi gặp kí tự kết thúc ở chuỗi truyền vào.

28. Hàm CHECK_SYNTAX

Duyệt từng kí tự, nếu là kí tự “/” thì bỏ qua, tiếp tục kiểm tra các kí tự tiếp theo, ở mỗi kí tự kiểm tra xem có phải kí tự chữ số hay không, nếu có kí tự khác kí tự chữ số thì chuỗi truyền vào không hợp lệ. Ta sử dụng mã ASCII của các kí tự để kiểm tra.

29. Hàm CHECK_DATE_IN_FILE

Kiểm tra tính hợp lệ của chuỗi trong file bằng cách kiểm tra syntax và logic của từng chuỗi TIME1 và TIME2, chỉ cần một điều kiện không thỏa thì dữ liệu trong file là không hợp lệ và sẽ không được in kết quả các yêu cầu ra file output.

III. QUI TẮC KHI VIẾT VÀ GỌI HÀM TRONG MIPS

1. MIPS hỗ trợ một số thanh ghi để lưu trữ các dữ liệu phục vụ cho thủ tục:

Đối số: \$a0, \$a1, \$a2, \$a3

Kết quả trả về: \$v0, \$v1

Biến cục bộ: \$s0, \$s1, ... , \$s7

Địa chỉ quay về \$ra

2. Cấu trúc của một thủ tục:

Đầu thủ tục

entry_label:

addi \$sp,\$sp, -framesize # khai báo kích thước cho stack

sw \$ra, framesize-4(\$sp) # cất địa chỉ trả về của thủ tục trong \$ra vào ngăn xếp

Lưu tạm các thanh ghi khác (nếu cần)

Thân thủ tục ...

(có thể gọi các thủ tục khác...)

Cuối thủ tục

Phục hồi các thanh ghi khác (nếu cần)

lw \$ra, framesize-4(\$sp) # lấy địa chỉ trả về ra \$ra

addi \$sp,\$sp, framesize

jr \$ra

Gọi thủ tục: jal entry_label

IV. ĐÁNH GIÁ MỨC ĐỘ HOÀN THÀNH: 100%

DANH MỤC TÀI LIỆU THAM KHẢO

- [1] <https://stackoverflow.com/questions/25953681/create-and-write-to-file-on-mips>
- [2] <http://www4.comp.polyu.edu.hk/~comp212/lab/lab9.pdf>

BIÊN BẢN PHÂN CÔNG VIỆC NHÓM

I. THÔNG TIN CHUNG

Các thành viên có tham gia làm đồ án:

STT	MSSV	Họ và tên
1	1712737	Đỗ Tấn Tài
2	1712724	Huỳnh Công Sinh
3	1712709	Trần Thiên Quàng
4	1712787	Nguyễn Văn Thìn
5	1712410	Phan Đại Hải

II. BẢNG PHÂN CÔNG CÔNG VIỆC

STT	Người phụ trách	Mô tả nội dung công việc	Kết quả
1	Đỗ Tấn Tài	Tổng hợp báo cáo, tổng hợp code. Code hàm 21 – 29, viết báo cáo phần hàm 21 – 29.	Hoàn thành 100%, đúng deadline nhóm
2	Huỳnh Công Sinh	Code hàm 06 – 10, viết báo cáo phần hàm 06 – 10.	Hoàn thành 100%, đúng deadline nhóm
3	Trần Thiên Quàng	Code hàm 01 – 05, viết báo cáo phần hàm 01 – 05.	Hoàn thành 100%, đúng deadline nhóm
4	Nguyễn Văn Thìn	Code hàm 11 – 15, viết báo cáo phần hàm 11 – 15.	Hoàn thành 100%, đúng deadline nhóm
5	Phan Đại Hải	Code hàm 16 – 20, viết báo cáo phần hàm 16 – 20.	Hoàn thành 100%, đúng deadline nhóm