**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN  
KHOA CÔNG NGHỆ THÔNG TIN  
MÔN HỆ THỐNG MÁY TÍNH**

**BÁO CÁO BÀI TẬP NHÓM**

**MÔN HỆ THỐNG MÁY TÍNH**

**BÀI TẬP 5.01**

**BÁO CÁO**

**Giảng viên lý thuyết :** **THÁI HÙNG VĂN**

**Lớp:** **21\_2**

**Thành viên nhóm thực hiện :**

**NGUYỄN HOÀI AN – 21120035**

**MÃ THÙY ANH – 21120037**

**NGUYỄN THANH HUỆ – 21120076**

**MỤC LỤC**

[**I. THÔNG TIN THÀNH VIÊN VÀ PHÂN CÔNG CÔNG VIỆC** 3](#_Toc133866539)

[**II. YÊU CẦU** 3](#_Toc133866540)

[**III. BÀI LÀM** 4](#_Toc133866541)

[**1. Giải thích chương trình** 4](#_Toc133866542)

[**2. Video thực hiện chương trình** 11](#_Toc133866543)

[**3. Kết luận** 12](#_Toc133866544)

[**IV. CÁC TÀI LIỆU THAM KHẢO** 13](#_Toc133866545)

[**1. Tài liệu tham khảo** 13](#_Toc133866546)

[**2. Công cụ sử dụng** 13](#_Toc133866547)

# **I. THÔNG TIN THÀNH VIÊN VÀ PHÂN CÔNG CÔNG VIỆC**

|  |  |  |  |
| --- | --- | --- | --- |
| **MSSV** | **Họ và tên** | **Công việc** | **Mức độ hoàn thiện** |
| 21120035 | Nguyễn Hoài An | Viết chương trình kiểm tra thời gian hợp lệ | 100% |
| Viết chương trình kiểm tra số nguyên tố hoặc 0/1 | 100% |
| Ghép các hàm tạo chương trình hoàn chỉnh | 100% |
| 21120037 | Mã Thùy Anh | Viết hàm vòng lặp in thời gian | 100% |
| Quay video demo chương trình | 100% |
| Viết báo cáo | 100% |
| 21120076 | Nguyễn Thanh Huệ | Viết chương trình lấy và in thời gian | 100% |
| Viết chương trình kiểm tra 2 số nguyên tố cùng nhau | 100% |
| Ghép các hàm tạo chương trình hoàn chỉnh | 100% |

# **II. YÊU CẦU**

Viết chương trình Hợp ngữ lấy giờ, phút & giây hiện tại của máy tính và hiển thị ra màn hình (dưới dạng hh:mm:ss, tại 1 vị trí cố định) cho đến khi có ít nhất 2 cặp trong 3 chỉ số đó là nguyên tố cùng nhau và không có chỉ số nào là nguyên tố hoặc là 0/1.

*Ví dụ, nếu bắt đầu lúc 22:23:00 thì thời điểm kết thúc là  22:24:09*

# **III. BÀI LÀM**

## **1. Giải thích chương trình**

Chương trình viết bằng Assembly lấy giờ, phút và giây hiện tại của máy tính hiển thị ra màn hình theo định dạng *hh:mm:ss* cho đến khi có ít nhất 2 cặp trong 3 chỉ số đó là nguyên tố cùng nhau và không có chỉ số nào là nguyên tố hoặc là 0/1.

**a. Dữ liệu đầu vào**

Text

Description automatically generated

- **INDEX** là biến bắt đầu vòng lặp dùng trong hàm CHECK\_VALID (kiểm tra giờ, phút, giây hợp lệ),

- **NUM\_HOUR, NUM\_MIN, NUM\_SEC** là các biến dùng để lưu dữ liệu giờ, phút, giây đầu vào để thực hiện so sánh trong hàm CHECK\_VALID

- **NUM1, NUM2** dùng để lưu dữ liệu giờ, phút, giây để thực hiện kiểm tra cặp số nguyên tố cùng nhau trong hàm COPRIME.

**b. Các hàm sử dụng**

* **GET\_TIME**: Xử lí giờ, phút, giây lấy từ máy và in ra màn hình tương ứng với chức năng của các hàm con **HOUR**, **MINUTES**, **SECONDS**.

Text

Description automatically generated

Text

Description automatically generated

Tại hàm **HOUR**, sau khi lấy được giờ từ hệ thống thì ta sẽ tiến hành gán số này vào AL rồi so sánh AL với 10 để chia làm 3 trường hợp: AL < 10, AL = 10, AL > 10.

Khi AL < 10 (hàm **hour1**), tức là AL là số có 1 chữ số, ta gán AH = 00 rồi nhảy đến hàm **next** để in BX (nhận giá trị sau khi được gán từ AX gồm AH và AL) ra màn hình thông qua hàm **DISP**.

Khi AL = 10 (hàm **hour2**), ta gán AH = 01, AL = 00 rồi thực hiện các bước in ra màn hình thông qua hàm **next**.

Khi AL > 10 (hàm **hour3**), ta gán AH = 01, AL = AL – 10 rồi so sánh AL với 10, nếu AL > 10 thì nhảy đến hàm **hour4** để gán AH = 02, AL = AL – 10 và nhảy đến hàm **next** để in ra màn hình, ngược lại thì giữ nguyên giá trị của AH, AL rồi in.

Text

Description automatically generated

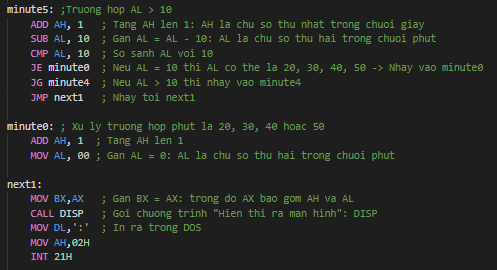
Tại hàm **MINUTES**, sau khi lấy phút từ máy, ta cũng tiến hành gán số vừa lấy được vào AL và so sánh AL với 10.

Đối với trường hợp AL < 10, ta nhảy đến hàm **minute1** để gán AH = 00 và nhảy tiếp đến hàm **next1** để in BX được gán từ AX gồm AH và AL ra màn hình.

Đối với trường hợp AL = 10, ta nhảy đến hàm **minute2** để gán AH = 01, AL = 00 và nhảy tiếp đến hàm **next1** để in ra màn hình

Đối với trường hợp AL > 10, ta nhảy đến hàm **minute3** để gán AH = 01, AL = AL–10 rồi so sánh AL với 10. Nếu AL vẫn lớn hơn 10 thì nhảy đến hàm **minute4**, còn khi AL = 10 thì nhảy đến **minute0** và khi AL < 10 thì nhảy đến **next1** để in ra màn hình.

Tại **minute4**, ta gán AH = AH + 1, AL = AL – 10 rồi so sánh AL với 10. Khi   
AL > 10 thì nhảy đến hàm **minute5**, khi AL = 10 thì nhảy đến hàm **minute0** và ngược lại, nhảy đến hàm **next1** để in ra màn hình.



Tại **minute5**, gán AH = AH + 1, AL = AL – 10 rồi so sánh AL với 10. Với   
AL = 10 thì nhảy đến hàm **minute0**, AL > 10 thì nhảy đến **minute4**, AL < 10 thì nhảy đến **next1**.

Tại **minute0**, gán AH = AH + 1, AL = 00 để xử lí các trường hợp số phút nhận được là 20, 30, 40 hoặc 50.

Tiếp tục lấy giây từ máy bằng cách dùng hàm **SECONDS**. Ta cũng thực hiện các bước gán, so sánh và in ra tương tự như đã làm ở hàm **MINUTES**.

Text

Description automatically generated

Text

Description automatically generated

* **ASSIGN**: Dùng để gán lại giờ, phút, giây vào các thanh ghi CH, CL, DH.
* **EXIT**:Kết thúc chương trình
* **LOOP\_TIME**: Lặp lấy thời gian và in ra màn hình

Text

Description automatically generated

* **DISP**: In thời gian ra màn hình

Text

Description automatically generated

* **CHECK\_VALID**: Kiểm tra giờ, phút, giây hợp lệ (không phải số nguyên tố hoặc 0 hoặc 1).

Đầu tiên ta so sánh số nhận được có bằng 0 hay không. Nếu có thì nhảy đến hàm **INVALID** để trả về không hợp lệ, không thì nhảy đếm hàm **CHECK\_1** để so sánh số đó với 1. Nếu bằng 1 thì nhảy đến hàm **INVALID** để trả về không hợp lệ, không thì nhảy đến hàm **CHECK\_2** để so sánh số đó với 2. Khi số đó bằng 2 thì nhảy đến **INVALID**, không thì nhảy đến hàm **CHECK\_PRIME** để kiểm tra số đó có phải số nguyên tố hay không.

Text

Description automatically generated

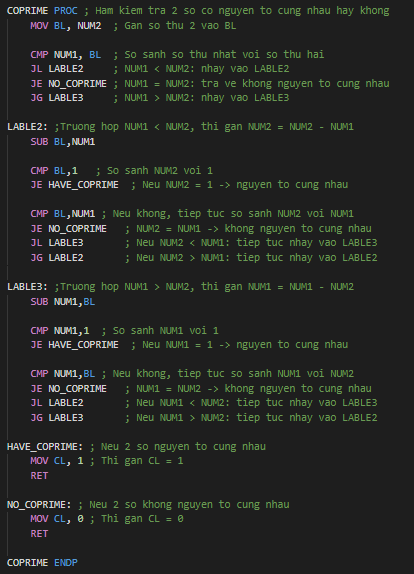
* **CHECK\_PRIME**: Kiểm tra số nguyên tố.

Tạo vòng lặp từ với biến INDEX bắt đầu từ 2 đến số cần kiểm tra. Nếu số cần kiểm tra bằng biến INDEX thì nhảy đến **INVALID** để báo không hợp lệ. Nếu không thì nhảy đến **CHECK\_SAME\_PRIME** để kiểm tra số đó có nguyên tố cùng nhau với biến INDEX hay không. Nếu số cần kiểm tra nguyên tố cùng nhau với tất cả các biến INDEX trong vòng lặp thì là số nguyên tố nhảy đến **INVALID** để báo không hợp lệ. Ngược lại, số đó sẽ hợp lệ nếu không phải số nguyên tố, nhảy đến **VALID** để báo hợp lệ.

Text

Description automatically generated

* **COPRIME**: Sau khi gọi hàm **ASSIGN** để gán, ta tiếp tục gán phút, giây vào NUM1, NUM2 và gọi **COPRIME** để kiểm tra hai số nguyên tố cùng nhau hay không. Kết quả trả về là 0 (sai) và 1 (đúng) sẽ được lưu vào thanh ghi CL. Ta tiếp tục kiểm tra với NUM1 = giờ, NUM2 = giây và NUM1 = giờ, NUM2 = phút.

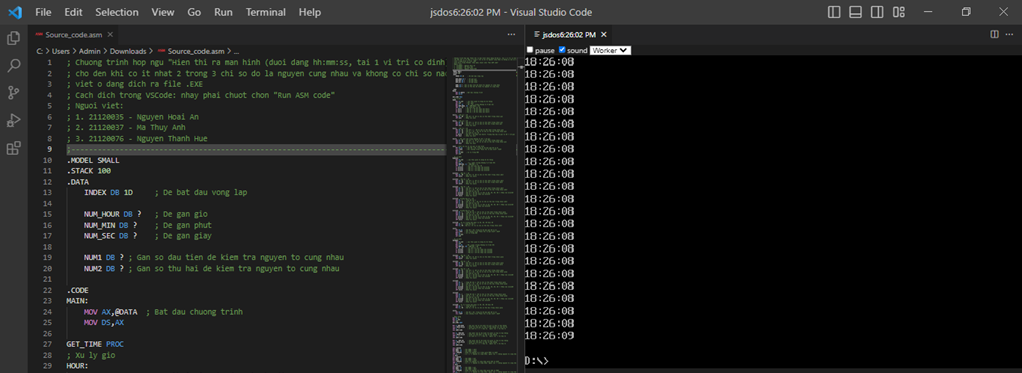


## **2. Video thực hiện chương trình**

Để xem bản chạy thử của chương trình, [nhấn vào đây](https://youtu.be/Y4NMrEnzlWo).

Link Youtube: [Demo bài tập 5.01 - YouTube](https://www.youtube.com/watch?v=Y4NMrEnzlWo&feature=youtu.be)

## **3. Kết luận**



Ta thấy rằng, thời gian bắt đầu là 18:26:02, thời gian kết thúc là tại 8:26:09 do có 2 cặp số nguyên tố cùng nhau là (8,9) và (26,9).

# **IV. CÁC TÀI LIỆU THAM KHẢO**

## **1. Tài liệu tham khảo**

[1] File bài giảng: KTMT-HN.Ch05.Assembly Basics.pdf

[2] File bài giảng: KTMT-HN.Ch06.Assembly Language Programming.pdf

[3] Github: [8086 Program to display Time in hh/mm/ss format](https://gist.github.com/Suman2593/8c4ecff6f37b811fc1f89b043b0f5e92?fbclid=IwAR05iq5j1C5QBoDm75KV7xnP75_KEqpWuKFt8vAZCOMK_8dUU-_8zeAvNug)

## **2. Công cụ sử dụng**

[1] Visual Code 2022

[2] MASM/TASM