



HỌC VIỆN KỸ THUẬT MẬT MÃ

**THỰC HÀNH LẬP TRÌNH
HỢP NGỮ TRÊN 8086**



NỘI DUNG 5

XỬ LÝ TẬP TIN

MỤC TIÊU

Viết được các chương trình xử lý tập tin như tạo tập tin, xóa tập tin, ghi tập tin, đọc nội dung của tập tin.

KIẾN THỨC CẦN CHUẨN BỊ

- Kết quả của các bài thực hành trước
- Các hàm 3dh, 3ch, 3eh, 3fh, 40h, 41h, 42h và 56h của INT 21h để xử lý tập tin.
- Các hàm 01, 02h, 06h, 08h, 09h, 0Ah của INT 21h và các lệnh của 8086.



5.1. Tạo tệp tin mới

Soạn thảo như đoạn chương trình mẫu phía dưới và lưu với tên là BAI_5A.ASM.

```
DSEG SEGMENT
tenfile db "d:\tt_asm\data.txt",0
thefile dw ?
DSEG ENDS
CSEG SEGMENT
ASSUME cs:cseg, ds:dseg
begin: mov ax, dseg
mov ds, ax
mov ah, 3ch ; tao tap tin moi
lea dx, tenfile
```

```
mov cx, 0 ; thuoc tinh tap tin
int 21h
mov thefile, ax ; cat the file
mov ah, 3eh ; dong tap tin
mov bx, thefile
int 21h
mov ah, 4ch ; thoat ve Dos
int 21h
CSEG ENDS
END begin
```



5.1. Tạo tệp tin mới

Soạn thảo như đoạn chương trình mẫu phía dưới và lưu với tên là BAI_5A.ASM.

```
DSEG SEGMENT
tenfile db "d:\tt_asm\data.txt",0
thefile dw ?
DSEG ENDS
CSEG SEGMENT
ASSUME cs:cseg, ds:dseg
begin: mov ax, dseg
mov ds, ax
mov ah, 3ch ; tao tap tin moi
lea dx, tenfile
```

```
mov cx, 0 ; thuoc tinh tap tin
int 21h
mov thefile, ax ; cat the file
mov ah, 3eh ; dong tap tin
mov bx, thefile
int 21h
mov ah, 4ch ; thoat ve Dos
int 21h
CSEG ENDS
END begin
```



5.1. Tạo tệp tin mới

- Biên dịch và cho chạy file BAI_5A.ASM để kiểm tra và xem kết quả. Gợi ý: Thư mục TT_ASM phải có sẵn trong ổ đĩa. Để biết chương trình chạy đúng hay sai, vào thư mục TT_ASM để xem có tệp tin Data.txt hay không, nếu có thì OK.
- Tại sao thẻ file phải được khai báo như dạng thefile DW ?
- Tại sao trong trường hợp này, chúng ta không phải dùng hàm 08h của int 21h đứng trước hàm 4ch của int 21h ?
- Tại sao phải cất thẻ file. Nếu chúng ta không cần đóng file thì chúng ta có cần cất thẻ file hay không ?



5.1. Tạo tệp tin mới

- Trong đoạn chương trình mẫu trên có cần thiết phải đóng tệp tin hay không ?
- Có thể bỏ biến thefile trong đoạn chương trình mẫu trên không ? Khi đó chúng ta phải dùng các lệnh gì để thay thế điều đó. Nếu có thay đổi, hãy biên dịch và cho chạy chương trình để kiểm chứng lại kết quả.



5.1. Tạo tệp tin mới

Hãy sửa đổi file BAI_5A.ASM và lưu với tên BAI_5A1.ASM để có thể thực hiện được yêu cầu sau: tạo một tệp tin mới, tên tệp tin được nhập từ bàn phím.

Gợi ý: dùng hàm 0ah của int 21h để nhập vào tên file, chú ý cuối chuỗi chứa tên file phải có zero, nhưng khi dùng hàm 0ah thì chúng ta không thể nào nhập zero vào cuối chuỗi được, nếu ta nhập ký tự '0' vào thì đó là mã ascii của ký tự '0' chứ không phải là zero (con số 0). Do đó, để thực hiện được điều này chúng ta hãy dùng giải thuật đưa 0 về cuối chuỗi như sau:



5.1. Tạo tệp tin mới

xor cx, cx ; dua zero ve cuoi chuoi
mov cl, len ; khai bao bien de dung ham 0ah cua int 21h dang
lea bx, tenfile; max db 250 ; so ky tu toi da duoc nhap
mov dl,0 ; len db ? ; chieu dai chuoi da nhap
mov [bx], dl ; tenfile db 250 dup(?); chua noi dung
duoc nhap



5.2. Ghi nội dung vào tập tin

Soạn thảo như đoạn chương trình mẫu phía dưới và lưu với tên là BAI_5B.ASM.

- Biên dịch và cho chạy file BAI_5B.ASM để kiểm tra và xem kết quả. (vào D:\TT_ASM để xem tập tin DATA.TXT có trong đó hay chưa và có nội dung hay chưa ?, nếu có là OK.)
- Xem xét đoạn chương trình mẫu, hãy đưa ra giải thuật ghi nội dung của vùng dữ liệu vào một tập tin vừa tạo.
- Lệnh *len db \$ - string1* được dùng để làm gì ?.
- Lệnh *XOR CX, CX* có ý nghĩa gì? Sau khi thực hiện xong lệnh này, thanh ghi CX có giá trị bằng bao nhiêu? Có thể thay thế nó bằng lệnh nào khác được không ?



5.2. Ghi nội dung vào tập tin

- Tại sao dùng lệnh *MOV CL, LEN* mà không dùng *MOV CX, LEN* hay *MOV CH, LEN* ? Khi dùng lệnh *MOV CX, LEN* thì cần phải thay đổi khai báo biến **len** như thế nào ? nếu không thay đổi thì sẽ có vấn đề gì xảy ra hay không ?
- Hãy thay đổi, biên dịch và chạy chương trình để kiểm chứng lại kết quả so với chương trình mẫu.



5.2. Ghi nội dung vào tập tin

- Hãy sửa đổi file BAI_5B.ASM và lưu với tên BAI_5B1.ASM để có thể thực hiện được yêu cầu sau: nhập từ một chuỗi ký tự bất kỳ, sau đó lưu vào tập tin có tên là “d:\tt_asm\solieu.txt”. Gợi ý: dùng hàm 0Ah của int 21h để nhập vào một chuỗi ký tự, sau đó áp dụng toàn bộ giải thuật của BAI_5B.ASM.
- Hãy sửa đổi file BAI_5B1.ASM và lưu với tên BAI_5B2.ASM để có thể thực hiện được yêu cầu sau: tạo tên tập tin mới, tên tập tin được nhập từ bàn phím

Sau đó nhập vào một chuỗi ký tự bất kỳ và lưu chuỗi ký tự đã nhập vào tập tin vừa tạo. Gợi ý: xem lại BAI_5A1.ASM để lấy lại giải thuật nhập vào tên file và đưa zero (con số 0) về cuối chuỗi đối với trường hợp tên file được nhập từ bàn phím và các vấn đề còn lại thì xem lại file BAI_5B1.ASM.



5.2. Ghi nội dung vào tập tin

- Hãy sửa đổi file BAI_5B2.ASM và lưu với tên BAI_5B3.ASM để có thể thực hiện được yêu cầu sau: tạo tên tập tin mới, tên tập tin được nhập từ bàn phím.

Sau đó nhập vào một chuỗi ký tự thường, sau đó đổi hoa ký tự đầu của mỗi từ và lưu chuỗi ký tự đã thay đổi vào tập tin vừa tạo. Gợi ý: xem lại các bài tập đã làm trong bài 5 xử lý ký tự và các file BAI_5B2.ASM.



5.2. Ghi nội dung vào tập tin

```
dseg segment
string1 db "Chao em co gai Lam Hong"
len db $ - string1
tenfile db "d:\tt_asm\data.txt",0
thefile dw ?
dseg ends
cseg segment
assume cs:cseg, ds:dseg
begin: mov ax, dseg
mov ds, ax
mov ah, 3ch ; tao tap tin moi
lea dx, tenfile
mov cx, 0 ; tap tin co thuoc tinh binh thuong
int 21h
```

```
mov thefile, ax ; cat the file
mov ah, 40h ; ghi file
mov bx, thefile
xor cx, cx
mov cl, len
lea dx, string1
int 21h
mov ah, 3eh ; dong tap tin
mov bx, thefile
int 21h
mov ah, 4ch ; thoat ve Dos
int 21h
cseg ends
end begin
```



5.3. Đọc nội dung tập tin

Soạn thảo như đoạn chương trình mẫu phía dưới và lưu với tên là BAI_5C.ASM.

- Biên dịch và cho chạy file BAI_5C.ASM để kiểm tra và xem kết quả.



5.3. Đọc nội dung tập tin

```
dseg segment
tenfile db "d:\tt_asm\data.txt",0
thefile dw ?
buffer db 251 dup ('$')
dseg ends
cseg segment
assume cs:cseg, ds:dseg
begin: mov ax, dseg
mov ds, ax
mov ah, 3dh ; mở tập tin đã có
lea dx, tenfile
mov al, 2 ; thuộc tính tập tin
int 21h
```

```
mov thefile, ax ; cat the file
mov ah, 3fh ; đọc nội dung file vào vùng đệm
mov bx, thefile
lea dx, buffer
mov cx, 250 ; số byte cần đọc từ file đã mở
int 21h
mov ah, 3eh ; đóng tập tin
mov bx, thefile
int 21h
mov ah, 09h ; in nội dung của file ra màn hình
lea dx, buffer
int 21h
mov ah, 08h ; dùng màn hình để xem kết quả
int 21h
mov ah, 4ch ; thoát về Dos
int 21h
cseg ends
end begin
```



5.3. Đọc nội dung tập tin

Xem lại đoạn chương trình mẫu, hãy đưa ra giải thuật đọc nội dung của tập tin và hiển thị nội dung đó ra màn hình.

- Hãy thử thay đổi thuộc tính tập tin trong lệnh `mov al, 2` lần lượt thành các giá trị khác như 0, 1, 3 hoặc 4. Biện dịch và cho chạy chương trình để xem kết quả. Có nhận xét gì về các giá trị này ?.
- Thẻ file có vai trò như thế nào trong xử lý tập tin.



5.3. Đọc nội dung tập tin

Nếu số byte cần đọc (giá trị chứa trong thanh ghi cx) lớn hơn kích thước thật sự của tập tin thì có gây ra lỗi gì hay không ?

Sau khi đọc nội dung của tập tin vào vùng đệm bằng hàm 3fh của INT 21h, thanh ghi ax sẽ có giá trị thay đổi hay không và nó chứa (giá trị) gì ?

Làm sao xác định được khi nào đọc xong nội dung thành tập tin ?
Hãy thử đưa ra hướng giải quyết?



5.3. Đọc nội dung tập tin

Hãy sửa đổi file BAI_5C.ASM và lưu với tên BAI_5C1.ASM để có thể thực hiện được yêu cầu sau: đọc nội dung của một tập tin và hiển thị nội dung đó lên màn hình. Tên tập tin được nhập từ bàn phím.

Gợi ý: sử dụng lại giải thuật nhập tên file từ bàn phím và đưa zero về cuối chuỗi



5.3. Đọc nội dung tập tin

Hãy sửa đổi file BAI_5C1.ASM và lưu với tên BAI_5C2.ASM để có thể thực hiện được yêu cầu sau: copy nội dung của một tập tin bất kỳ sau đó paste sang một vị trí khác.

Gợi ý: các giải thuật nhập tên file từ bàn phím thì có sẵn, giải thuật copy và paste như sau: trước hết phải mở tập tin đã có bằng hàm 3dh, chúng ta định nghĩa sẵn đường dẫn chứa tên file cần mở, đọc nội dung của tập tin vào vùng đệm bằng hàm 3fh, nhớ cất trữ file; tạo tên tập tin mới bằng hàm 3ch, chúng ta định nghĩa sẵn đường dẫn chứa tên file cần tạo, nhớ cất trữ file; ghi nội dung của vùng đệm vào tập tin mới vừa tạo bằng hàm 40h; đóng hai tập tin lại bằng hàm 3eh, tất cả các hàm này đều của int 21h.



5.3. Đọc nội dung tập tin

Hãy sửa đổi file BAI_5C2.ASM và lưu với tên BAI_5C3.ASM để có thể thực hiện được yêu cầu sau: copy nội dung của một tập tin bất kỳ sau đó paste sang một vị trí khác. Tên tập tin được copy và tập tin sau khi paste đều nhập từ bàn phím.

Gợi ý: hoàn toàn tương tự như BAI_5C2.ASM, nhưng chúng ta cần tận dụng lại giải thuật nhập tên file từ bàn phím từ BAI_5B3.ASM.



5.3. Đọc nội dung tập tin

Hãy sửa đổi file BAI_5C3.ASM và lưu với tên BAI_5C4.ASM để có thể thực hiện được yêu cầu sau: save as nội dung của một tập tin. Tên tập tin được copy và save as đều nhập từ bàn phím. Gợi ý: bài này giống tương tự như BAI_5C3.ASM.



5.3. Đọc nội dung tập tin

Hãy sửa đổi file BAI_5C3.ASM và lưu với tên BAI_5C4.ASM để có thể thực hiện được yêu cầu sau: đọc nội dung của tập tin và sau đó nhập một chuỗi ký tự bất kỳ và ghi tiếp theo sau nội dung của tập tin vừa mở. Tên tập tin nhập từ bàn phím.

Gợi ý: xem lại BAI_5C1.ASM, BAI_5B1.ASM, BAI_5B2.ASM và hàm 42h của int 21h (dời vị trí con trỏ tập tin).



5.3. Đọc nội dung tập tin

Hãy sửa đổi file BAI_5C4.ASM và lưu với tên BAI_5C5.ASM để có thể thực hiện được yêu cầu sau: mã hóa nội dung của tập tin. Tên tập tin cần mã hóa được nhập từ bàn phím.

Gợi ý: nhập vào tên file cần mã hóa, mở một file đã có bằng hàm 3dh; đọc nội dung của tập tin vào vùng đệm bằng hàm 3fh; mã hóa vùng đệm đọc được bằng một trong các phép toán cộng, trừ, nhân, chia, and, or, not ..., vừa mã hóa vừa lưu nội dung trở lại vùng đệm; dời vị trí con trỏ tập tin về đầu tập tin bằng hàm 42h; sau đó ghi nội dung của vùng đệm trở lại tập tin ban đầu thông qua thẻ file của nó. Tất cả các hàm sử dụng ở đây đều của int 21h.



5.3. Đọc nội dung tập tin

Hãy sửa đổi file BAI_5C5.ASM và lưu với tên BAI_5C6.ASM để có thể thực hiện được yêu cầu sau: giải mã nội dung của tập tin đã mã hóa. Tên tập tin cần giải mã được nhập từ bàn phím.

Gợi ý: giải mã là trường hợp ngược lại của mã hóa, nếu mã hóa theo phương thức nào thì giải mã phải làm ngược lại phương thức mã hoá đó.



5.4. Xóa tập tin

Soạn thảo như đoạn chương trình mẫu phía dưới và lưu với tên là BAI_5D.ASM.

- Biên dịch và cho chạy file BAI_5D.ASM để kiểm tra và xem kết quả.
- Hãy sửa đổi file BAI_5D.ASM và lưu với tên BAI_5D1.ASM để có thể thực hiện được yêu cầu sau: xóa tên một tập tin. Tên tập tin cần được nhập từ bàn phím.

Gợi ý: xem lại các bài trước để lấy giải thuật nhập tên file từ bàn phím.



5.4. Xóa tập tin

```
dseg segment  
tenfile db "d:\tt_asm\data.txt",0  
dseg ends  
cseg segment  
assume cs:cseg, ds:dseg  
begin: mov ax, dseg  
mov ds, ax  
mov ah,41h ; xoa tap tin da co  
lea dx, tenfile  
int 21h  
mov ah, 4ch ; thoat ve Dos  
int 21h  
cseg ends  
end begin
```



5.5. Đổi tên tập tin

Soạn thảo như đoạn chương trình mẫu phía dưới và lưu với tên là BAI_5E.ASM.

- Biên dịch và cho chạy file BAI_5E.ASM để kiểm tra và xem kết quả



5.5. Đổi tên tập tin

```
dseg segment
oldfile db "d:\tt_asm\data.txt",0
newfile db "d:\tt_asm\solieu.txt",0
dseg ends
cseg segment
assume cs:cseg, ds:dseg, es: dseg
begin: mov ax, dseg
mov ds, ax
mov es, ax
mov ah,56h ; rename/remove tên file cũ thành mới
lea dx, oldfile
lea di, newfile
int 21h
mov ah, 4ch ; thoát về Dos
int 21h
cseg ends
end begin
```



5.5. Đổi tên tập tin

Hãy tạo một thư mục con có tên là baitap nằm trong thư mục tt_asm. Sửa đổi lệnh oldfile db “d:\tt_asm\data.txt”,0 và newfile db “tt_asm\solieu.txt”,0 lại thành oldfile db “d:\tt_asm\solieu.txt”,0 và newfile db “tt_asm\baitap\data.txt”,0. Biên dịch lại và cho chạy chương trình để xem xét kết quả. Có nhận xét gì về kết quả nhận được.

Hãy sửa đổi file BAI_5E.ASM và lưu với tên BAI_5E1.ASM để có thể thực hiện được yêu cầu sau: đổi tên một tập tin. Tên tập tin cũ và mới được nhập từ bàn phím.

Gợi ý: xem lại các bài tập trước để lấy giải thuật nhập tên file từ bàn phím



5.6. Bài tập

1. Viết chương trình sử dụng hàm 41h/ INT 21h để xóa tập tin trên đĩa. Tên tập tin cần xóa được nhập từ bàn phím khi thực hiện chương trình.
2. Viết chương trình nhập 1 chuỗi từ bàn phím, sau đó ghép chuỗi nhận được vào cuối của nội dung tập tin có trên đĩa. Tên tập tin nhập từ bàn phím khi chạy chương trình.
3. Viết chương trình nhập 1 chuỗi từ bàn phím, sau đó chèn chuỗi nhận được vào đầu của nội dung tập tin có trên đĩa. Tên tập tin nhập từ bàn phím khi chạy chương trình.
4. Viết chương trình ghép nội dung 2 tập tin có sẵn trên đĩa thành 1 tập tin mới. Tên của các tập tin được nhập từ bàn phím khi chạy chương trình.
5. Viết chương trình đọc nội dung tập tin trên đĩa, sau đó đổi tất cả ký tự HOA thành ký tự thường và lưu lại vào tập tin đó. Tên tập tin phải được nhập từ bàn phím