

Лабораторна робота №7
ВИВЕДЕННЯ СПИСКУ ФАЙЛІВ В ПОТОЧНОМУ КАТАЛОЗІ

Мета: Навчитися користуватися командами пошуку файлів, закріпити навички по виводу рядків, що містять крім літерних, ще й цифрові дані.

Обладнання: будь-який текстовий редактор що не використовує службових символів, компілятор асемблера (TASM або MASM), дебагер (рекомендується Turbo Debugger).

Хід роботи

1. Постановка задачі та розробка алгоритму.

Іноді буває важче знайти потрібний файл, ніж відкрити його і забезпечити введення-виведення інформації. Тому на під час виконання цієї лабораторної роботи ми ознайомимося з функціями пошуку файлу.

Завдання наше буде наступним: потрібно, користуючись функціями пошуку файлів, вивести на екран перелік всіх файлів, що містяться в поточному каталозі (аналог команди DIR). Крім імені файлу треба вивести час та дату створення файлу.

Для забезпечення пошуку файлів ми можемо використовувати дві функції: знайти перший файл, знайти наступний файл.

Алгоритм задачі буде наступний:

1. Задаємо маску пошуку “*.*”.
2. Шукаємо перший файл.
3. Виводимо ім'я файлу, дату та час його створення.
4. Шукаємо наступний файл.
5. Якщо файлів більше нема – йдемо на крок 8.
6. Виводимо ім'я файлу, дату та час його створення.
7. Повертаємося на крок 4.
8. Кінець програми.

За цим алгоритмом потрібно скласти блок-схему та написати програму.

2. Написання програми.

Дві функції операційної системи нам знадобляться, щоб виконати наше завдання. Перша – це функція 4Eh (INT 21h) “Знайти перший файл”, а друга – функція 4Fh (INT 21h) “знайти наступний файл”.

Щоб знайти всі файли, потрібно для пошуку використовувати маску “*.*”. Після кожної з вказаних вище процедур треба перевіряти флаг CF – якщо він =1, значить файлів більше нема і пошук можна завершити.

Для того, щоб отримати ім'я файлу, дату та час його створення, нам потрібно звернутися до відповідних полів DTA. Зазвичай ця область даних розташована зі зміщенням 0080h від початку блоку PSP. Але ми можемо перепризначити адресу цієї області за допомогою функції 1Ah.

Крім того нам потрібно буде перетворити дату та час створення файлу з двоїчного формату в коді ASCII, щоб вивести на екран у вигляді рядків.

Виводити данні можна за допомогою будь-якої відомої нам функції виводу на екран. Циклічність виводу досягається за допомогою команд умовних переходів.

Робота з файлами. Однією з важливих задач операційної системи є керування розміщенням даних у зовнішній пам'яті (пам'яті довгострокового збереження). У сучасних ПК як пристрої зовнішньої пам'яті найчастіше використовуються дискові нагромаджувачі, у першу чергу — нагромаджувач на твердому магнітному диску (HDD або «вінчестер»). Порція інформації (найчастіше однорідної), що зберігається в зовнішній пам'яті під визначеним ім'ям, називається файлом.

Підсистема обліку розміщення інформації на пристрої зовнішньої пам'яті називається файловою системою. Від неї залежить, яка одиниця простору пам'яті є найменшою, чи можуть файли займати незв'язані ділянки пам'яті, як іменуються файли, якими властивостями може володіти файл і які операції над ним можна робити і т.п. У DOS для дискових нагромаджувачів використовується файлова система FAT. Вона дозволяє розділяти дисковий простір HDD на кілька розділів і організовувати в кожному з розділів ієрархічну структуру каталогів, що містять файли. Каталог, що представляється звичайно у виді іменованої сукупності файлів, також по суті є файлом визначеної структури, що містить список файлів, що зберігаються в ньому. (Для роботи з каталогами в DOS мають спеціальні функції).

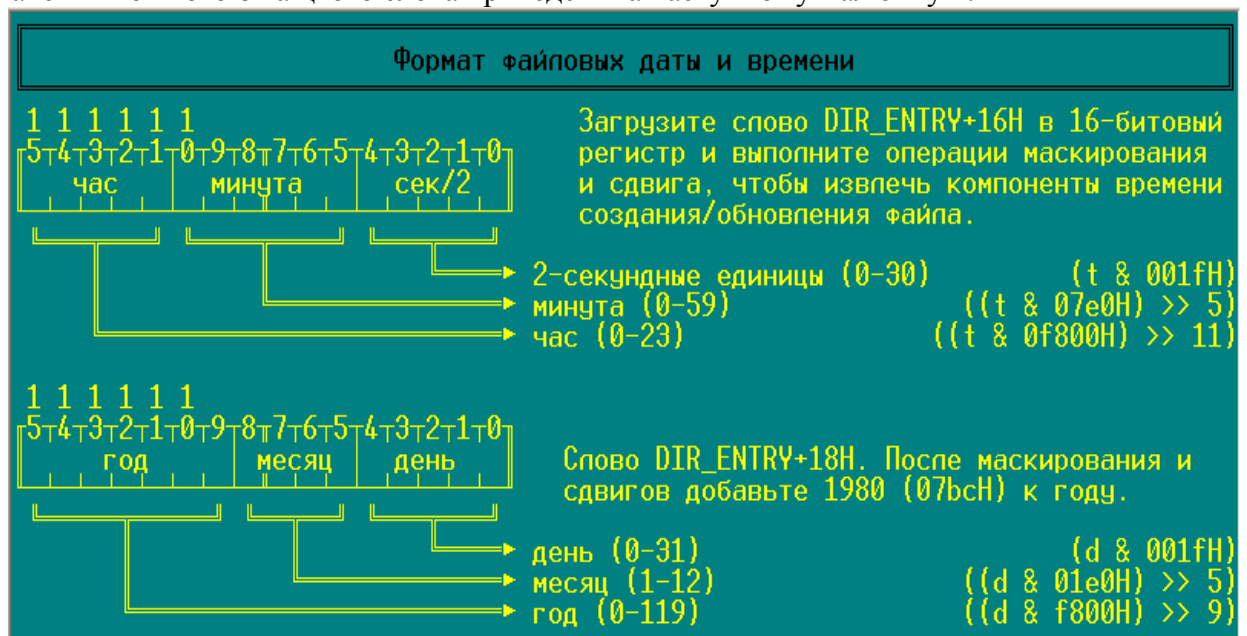
Ім'я файлу (і, відповідно, каталогу) може містити від 1 до 8 символів безпосередньо в імені і від 0 до 3 символів — у розширенні файлу, що звичайно пояснює його тип. Ім'я і розширення відокремлюються крапкою. Сумарна довжина імені, таким чином, складає максимум 12 байт.

DOS підтримує ряд наступних функцій для роботи з файлами:

№ у АН	Опис	Вхід	Вихід
1Ah	Встановити адрес DTA	DS:DX – адрес для DTA	
4Eh	Знайти перший файл	DS:DX → ASCIIZ-ім'я/маска; CX = атрибути	[DTA+15h] - атрибут; [DTA+16h] - дата; [DTA+18h] - час; [DTA+1Ah] - розмір; [DTA+1Eh] - ASCIIZ-ім'я або CF = 1 (не знайдений)
4Fh	Знайти наступний файл	Дані записуються в DTA після попереднього пошуку	

*ASCIIZ – '(шлях\)'ім'я',0 (якщо шлях не зазначений, використовується поточний каталог).

Наприклад, для того, щоб отримати час необхідно до адресу DTA додати відповідне зміщення (для часу 16h), тепер ми отримали адрес двухбайтного слова у якому записаний час. Значення кожного біта цього слова приведені на наступному малюнку 1:



Мал. 1

Усі функції у випадку помилки встановлюють CF в 1 і повертають у AX код помилки:

He	De	Значення	Hex	Dec	Значення
1	1	Невірний номер функції	0Ah	10	Невірне оточення
2	2	Файл не знайдений	0Bh	11	Невірний формат
3	3	Шлях не знайдений	0Ch	12	Невірний код доступу
4	4	Занадто багато відкритих	0Dh	13	Невірна дата
5	5	файлів	0Eh	14	(не використовується)
6	6	Доступ не дозволений	0Fh	15	Задано невірний диск
7	7	Невірний дескриптор	10h	16	Не можна видаляти тек.
8	8	Зруйновано блоки упр.	11h	17	каталог
9	9	пам'яттю	12h	18	Не той пристрій
		Недостатньо пам'яті			Більше немає шуканих файлів
		Невірна адреса блоку пам'яті			

Стандартні дескриптори:

0 CON - стандартне введення (клавіатура);

1 CON - стандартний висновок (екран);

- 2 стандартний пристрій помилок (екран);
- 3 AUX - асинхронний адаптер (COM1);
- 4 стандартний принтер (LPT1).

Атрибути (біти):

	a	d	s	v	h	r
--	---	---	---	---	---	---

- a - (archive) архівний;
d - (directory) каталог;
v - (volume) мітка тому;
s - (system) системний;
h - (hidden) схований;
r - (read only) тільки для читання.

Розглянемо основні поняття, які необхідно мати при використанні цих функцій.

Починаючи з версії 2 у MS-DOS використовується дескрипторний метод роботи з файлами. При створенні або відкритті файлові привласнюється 16-розрядний двійковий номер, називаний дескриптором. Надалі при виконанні операцій читання, запису й інших необхідно вказувати привласнений файлові дескриптор.

Мається п'ять визначених дескрипторів, що відповідають стандартним пристроям введення-висновку: клавіатурі, екранові і комунікаційним портам. Використання визначених дескрипторів дозволяє здійснювати введення-вивід з цими пристроями точно так само, як і з файлами, а також організовувати потоки даних, що не залежать від типу джерел і приймачів даних.

Імена файлів, каталогів або шаблонів для пошуку повинні записуватися в ASCIIZ-форматі, тобто наприкінці імені вказується нульовий байт.

При читанні або записі файлу операційна система підтримує двохсловну змінну — покажчик файлу, що вказує на байт у файлі, до якого здійснюється доступ. В міру обробки файлу покажчик файлу зміщується до наступних байтів. Спеціальна функція DOS дозволяє задавати відносне або абсолютне положення покажчика у файлі.

Пошук групи файлів, імена яких задовольняють визначеному шаблону, виконується в два етапи. На першому етапі використовується функція 'Знайти перший файл', при цьому вказується шаблон, що може містити шлях до оброблюваного каталогу і шаблонні символи '*' (послідовність припустимих символів довільної довжини) і '?' (будь-який припустимий символ у даній позиції). Якщо шлях не зазначений, мається на увазі поточний каталог (це справедливо й в інших подібних випадках). Якщо файл не знайдений, встановлюється флаг CF, інакше інформація про знайдений файл записується в область DTA. На другому етапі викликається функція 'Знайти наступний файл', що використовує інформацію з DTA після попереднього пошуку. Ця функція при кожному новому виклику знаходить наступний файл, що задовольняє первісному шаблону. Файли знаходяться в порядку їхнього проходження в записі каталогу. Підкаталоги не обробляються.

Поряд зі службовою інформацією, що забезпечує можливість пошуку чергового файлу, у DTA записуються зведення про атрибути, дату і час модифікації, розмірі знайденого файлу, а також ім'я файлу (без шляху) у форматі ASCIIZ. Якщо довжина імені з розширенням складає менш 12 байт, залишок полючи заповнюється нулями.

Ім'я файла ми можемо вивести на екран одразу, а ось інформацію про даний файл, вивести складно. Для виведення місяця дивіться лабораторну роботу № 5, а для виведення числової інформації потрібно зробити наступне:

1) Спочатку необхідно виділити необхідні біти, для цього наше двохбайтне слово потрібно логічною операцією AND скласти з відповідною маскою, і здвинути вправо на певну кількість біт (див. Мал. 1).

2) Тепер отримане число ділимо на 10, в наслідок цього в одному регістрі ми отримаємо результат, а в іншому залишок від ділення, до результату ми додаємо 30h або це дорівнює '0', додавши виводимо на екран або записуємо в пам'ять.

3) Далі результат від ділення знову ділимо на 10 і виконуємо дії, які описані в П.2

4) Коли число буде менше 10, ми його вже не ділимо, а просто до нього додаємо 30h і виводимо на екран або записуємо в пам'ять.

3. Тестування програми. Аналіз результатів.

Після написання програми необхідно перевірити її працездатність.

Особливу увагу треба приділити правильності роботи функцій зчитування імені, часу та дати знайдених файлів. Також треба слідкувати, щоб не виникало нескінченних циклів.

По закінченню написання та тестування ми повинні отримати програму, що буде при запуску виводити на екран список файлів поточного каталогу.

4. Висновки.

Під час виконання цієї лабораторної роботи ми ознайомились з двома новими функціями: функціями пошуку файлів, а також з функцією переадресації області DTA. Також ми закріпили і отримали нові навички по переведенню числових даних в рядок ASCII символів, коли готували до виводу дату та час створення файлу.

5. Контрольні запитання.

- 1). Яку функцію потрібно використати для пошуку 1-го файлу?
- 2). Яку функцію ми використовуємо для пошуку наступного файлу?
- 3). Як ми можемо перевірити, що файлів в каталозі більше немає?
- 4). Яку структуру має область DTA? Що ми можемо з неї взяти?
- 5). Де розташовується зазвичай DTA? За допомогою якої функції ми можемо переадресувати цю область в інше місце?

```
model tiny
.code
.startup
;-----
        lea dx,DTA                                ;встановлення адреси зміщення для
                                                ;локального буфера

        mov ah,1ah
        int 21h

        call Find_First                          ;пошук першого файла
        jc exit1                                 ;якщо нема файлів перейти до exit1
        call Out_Info                            ;виведення даних про файл

cukl:    call Find_Next                          ;пошук наступного файла
        jc exit2                                 ;якщо нема файлів перейти до exit2
        call Out_Info                            ;виведення даних про файл
        loop cukl                               ;перейти до cukl

exit1:   lea dx,info                             ;виведення даних info
        mov ah,09h
        int 21h

exit2:   mov ah,4ch                             ;завершення програми
        int 21h
;-----
Find_First proc                                ;пошук першого файла
        lea dx,Path                             ;загрузка в dx адреса строки Path
        mov cx,00                               ;встановлення атрибутів
        mov ah,4eh                             ;виклик функції для пошуку
        int 21h
        ret
Find_First endp

Find_Next proc                                ;пошук наступного файла
        lea dx,Path                             ;загрузка в dx адреса строки Path
        mov cx,00                               ;встановлення атрибутів
        mov ah,4fh                             ;виклик функції для пошуку
        int 21h
```

```

    ret
Find_Next endp

Out_Info proc                                     ;виведення даних про файл
;-----
    lea dx, strk                                ;вивід імені файла
    mov ah, 9h                                  ;загрузка в dx адреса строки strk
    int 21h                                     ;для переведення курсора на
                                                ;наступний рядок

    mov cx, 0dh                                  ;запис в cx 0dh - кількість байтів,
                                                ;які буде виводити ф-я 40h
    mov bx, 1eh                                  ; імені
                                                ; імені файла відносно DTA
    lea dx, DTA + bx                            ; загрузка в dx адреса зміщення
                                                ; імені файла
    mov bx, 1                                    ;запис в bx=1, див. використання
                                                ;функції 40h
    mov ah, 40h                                  ;виклик функції 40h
    int 21h
;-----

    lea dx, sozdan                              ;вивід на екран слова sozdan
    mov ah, 09h
    int 21h

    mov cx, 2                                    ;запис в cx 0dh - кількість байтів,
    mov bx, 18h                                  ;загрузка в bx адреса зміщення
                                                ;для визначення дати
    lea bx, DTA + bx                            ;загрузка в bx адреса зміщення дати
    mov ax, [bx]                                ;запис в ax значення, що знаходиться
                                                ;по адресі в bx
    mov di, ax                                  ;запис в di значення ax (в di будемо
                                                ;зберігати два байта, що зберігають
                                                ;дані про дату
    mov si, 10                                  ;Vuvid chusla
    and ax, 001fh                               ;запис в si значення 10
                                                ;лог. операція складання числа в ax
    mov dx, 0                                    ;з маскою 001fh результат в ax
    div si                                       ;запис в dx значення 0
    add dx, 30h                                  ;AX/10 результат в AX, залишок в DX
    lea bx, chuslo                              ;додавання до залишку 30h
    mov [bx+1], dl                              ; загрузка в bx адреса зміщення
                                                ;строки chuslo
    add ax, 30h                                  ;запис в строку із адресом зміщення
    lea bx, chuslo                              ; [bx+1] значення, регістра dl
    mov [bx], al
    lea dx, chuslo
    mov ah, 09h
    int 21h
;-----

    mov ax, di                                  ;визначення місяця
    and ax, 01e0h                              ;запис в ax значення di
    shr ax, 5                                   ;лог. операція складання числа в ax
    sub al, 1                                   ;з маскою 01e0h, результат в ax
    mov ah, 11                                  ;здви́г ax на 5 біт вправо
    mul ah                                       ; al=al-1
                                                ; ah=11 (11 символів між початками
                                                ; місяців строки Mis)
    mul ah                                       ; al=al*ah

```

```

    lea dx,Mis                ; загрузка в dx адреса зм.строки Mis
    add dx,ax                 ; dx=dx+ax

    mov ah,9h                ;вивід на екран строки,яка знаходить-
                              ;ся по адресу dx

    int 21h
;-----
                              ;визначення року
;визначення року відбувається аналогічно визначенню числа, тільки
;потрібно додати 1980, так як вказано у теоретичних відомостях
;і крім того потрібно ділити рік на 10 3 рази, тому що він складається
;з чотирьох цифр (для числа ділити 1 раз, воно склад. з двох цифр),
;а все інше за тим самим принципом, що і для визначення числа
    mov ax,di
    and ax,0fe00h
    shr ax,9                  ;Vuvod roky
    add ax,1980

    mov bp,0
    mov di,ax
    lea bx,rik+4

cukl2:    mov ax,di
    mov dx,0
    div si
    add dx,'0'
    mov di,ax

    mov [bx],dl
    sub bx,1

    mov cx,4
    add bp,1
    sub cx,bp
    loop cukl2

    mov dx,di
    add dx,'0'
    mov [bx],dl

    lea dx,rik
    mov ah,9h
    int 21h

    lea dx,OB
    mov ah,09h
    int 21h
;час визначається аналогічно як і число, тільки для визначення для
;байтів, які містять інформацію про час для даного файла використовуємо
;зміщення 16h і і використовуємо відповідні маски, у випадку визначення
;секунди множимо кінцевий результат на 2 (див мал.1 )
    mov cx,2
    mov bx,16h
    lea bx,DTA + bx
    mov ax,[bx]               ;di - інформація про час
    mov di,ax

                              ;визначення години

    mov si,10
    and ax,0f800h
    shr ax,11
    mov dx,0
    div si
    add dx,'0'
    lea bx,chas
    mov [bx+1],dl

    add ax,'0'

```

```

    lea bx,chas
    mov [bx],al

;ВИЗНАЧЕННЯ ХВИЛИНИ

    mov ax,di
    mov si,10
    and ax,07e0h
    shr ax,5
    mov dx,0
    div si
    add dx,'0'
    lea bx,chas
    mov [bx+4],dl

    add ax,'0'
    lea bx,chas
    mov [bx+3],al

;ВИЗНАЧЕННЯ СЕКУНДИ

    mov ax,di
    mov si,10
    and ax,01fh

    mov ah,al
    mov al,2
    mul ah
;результат ділення в al

    mov dx,0
    div si

    add dx,'0'
    lea bx,chas
    mov [bx+7],dl

    add ax,'0'
    lea bx,chas
    mov [bx+6],al

    lea dx,chas
    mov ah,09h
    int 21h
    ret
Out_Info endp
;-----
    ret
DTA db 50 dup(?),'$' ;локальний буфер інформації про файл
Path db "D:\*.*",0 ;шлях і маска пошуку файлів
strk db 0ah,0dh,'$' ;перев. курсора на поч. наступного рядка
rik db 5 dup(?),' roky$' ;рядок, в який записується рік
Mis db ' Sichen $ Lyutogo $ Bereznya $ Kvitnya $ Travnnya $ Chervnya $ Lupnya
$ Serpnya $ Veresnya $ Ghovtnya $ Lustopada$ Grudnya $'
chuslo db ' '$' ;рядок, в який записується число

Sozdan db ' sozdan: $' ;рядок в пам'яті із словом sozdan:
OB db ' y $' ;рядок в пам'яті із словом y
chas db ' : : $' ;рядок, в який записується час
Info db 'V daniy deruktoriyi fayliv nema!!!$'
end

```

Додаткові завдання за варіантами:

0. Написати на мові асемблер com-програму, яка дозволить вивести данні про рік створення першого файлу у каталозі (в текстовому форматі).
1. Написати на мові асемблер exe-програму, яка дозволить вивести данні про рік створення першого файлу у каталозі (в текстовому форматі).
2. Написати на мові асемблер com-програму, яка дозволить вивести данні про місяць створення першого файлу у каталозі (в текстовому форматі).
3. Написати на мові асемблер exe-програму, яка дозволить вивести данні про місяць створення першого файлу у каталозі (в текстовому форматі).
4. Написати на мові асемблер com-програму, яка дозволить вивести данні про час створення (години і хвилини) першого файлу у каталозі.
5. Написати на мові асемблер exe-програму, яка дозволить вивести данні про час створення (години і хвилини) першого файлу у каталозі.
6. Написати на мові асемблер com-програму, яка дозволить вивести назву першого файлу у каталозі.
7. Написати на мові асемблер exe-програму, яка дозволить вивести назву першого файлу у каталозі.
8. Написати на мові асемблер com-програму, яка дозволить вивести назву другого файлу у каталозі.
9. Написати на мові асемблер exe-програму, яка дозволить вивести назву другого файлу у каталозі.