# File IO

CSC 236

# Terminal input and output

- Input
  - Is a sequence of bytes
  - Often ASCII character codes
  - Compare to getchar() in C
- Must convert for numeric input
  - Integer
  - Real
  - ...

- Output
  - A sequence of bytes
  - Often ASCII character codes
  - Compare to putchar() in C.
- For numeric output, we would have to:
  - Convert binary/hex byte/word
  - ASCII codes for digits

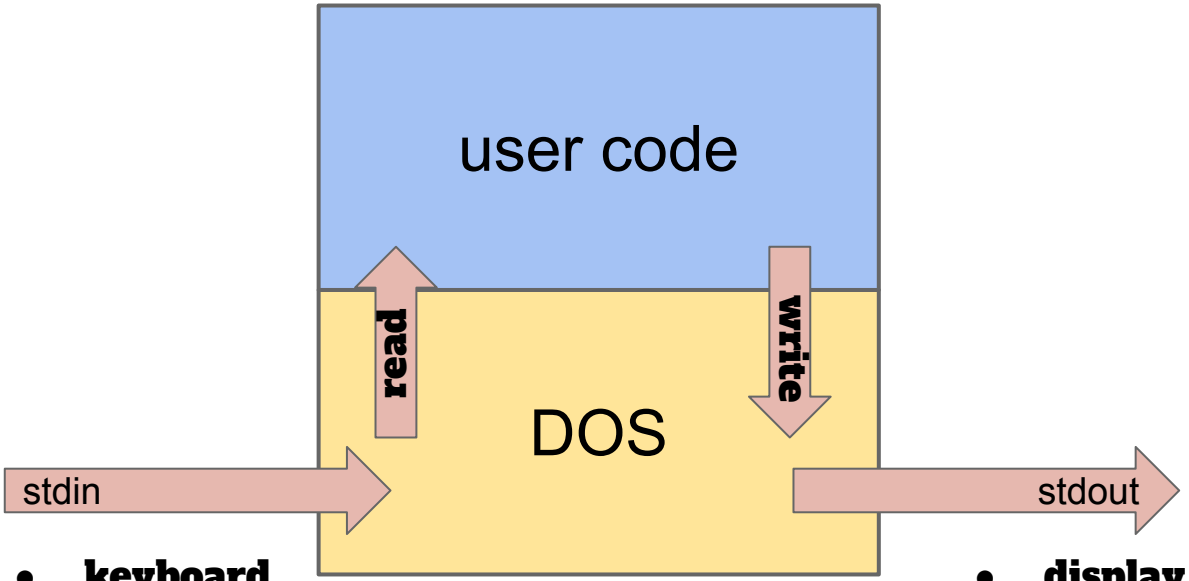
# Read an integer from the terminal

- When you type
  - 165
- Terminal receives
  - 3 ASCII characters (bytes)
  - 31h 36h 35h
  - $49_{10}$ $54_{10}$ $53_{10}$
- Desired value in register
  - $A5_{16}$

- Must convert to integer
  - How?
- You can use Horner's rule to handle digits left-to-right.

```
n = 0;
while c = read() {
  n *= 10;
  n += c - 48;
}
```

# IO in DOS


user code
read
write
DOS
stdin
stdout
• keyboard
• file
• display
• file

# Read/write one ASCII character

| Operation | Syntax | Semantics | Example |
|---|---|---|---|
| **Read - no echo** | ***ah*=8** | **reads char into *al*** | `mov ah,8`<br>`int 21h` |
| **Write - char must be in dl** | ***ah*=2**<br>***dl*=char** | **char in *dl* is written to standard output** | `mov ah,2`<br>`mov dl,[char]`<br>`int 21h` |

```
;---------------------------------------
        .model    small
        .8086
        .stack    256
;---------------------------------------

        .data
;---------------------------------------
end_char db       0Dh
;---------------------------------------

        .code
;---------------------------------------
start:
        mov       ax,@data
        mov       ds,ax
getloop:
        mov       ah,8
        int       21h
        mov       dl,al
        mov       ah,2
        int       21h

        cmp       dl,[end_char]
        jne       getloop
;---------------------------------------
exit:
        mov       ax,4c00h
        int       21h
        end       start
;---------------------------------------
```

# copyfile

- program in samples directory

```
;---------------------------------------
        .model    small
        .8086
        .stack    256
;---------------------------------------

        .data
;---------------------------------------
end_char db       0Dh
;---------------------------------------

        .code
;---------------------------------------
start:
        mov       ax,@data
        mov       ds,ax
getloop:
        mov       ah,8
        int       21h
        mov       dl,al
        mov       ah,2
        int       21h

        cmp       dl,[end_char]
        jne       getloop
;---------------------------------------
exit:
        mov       ax,4c00h
        int       21h
        end       start
;---------------------------------------
```

# copyfile

- program in samples directory
  - copies from standard input to standard output
  - stops after it sees the start of a line terminator
    - 0Dh — enter key
    - Normally, 0Ah follows this.

```
;---------------------------------------
        .model    small
        .8086
        .stack    256
;---------------------------------------

        .data
;---------------------------------------
end_char db       0Dh
;---------------------------------------

        .code
;---------------------------------------
start:
        mov       ax,@data
        mov       ds,ax
getloop:
        mov       ah,8
        int       21h
        mov       dl,al
        mov       ah,2
        int       21h

        cmp       dl,[end_char]
        jne       getloop
;---------------------------------------
exit:
        mov       ax,4c00h
        int       21h
        end       start
;---------------------------------------
```

# copyfile

- Initialize the ds register

```
;---------------------------------------
        .model    small
        .8086
        .stack    256
;---------------------------------------

        .data
;---------------------------------------
end_char db       0Dh
;---------------------------------------

        .code
;---------------------------------------
start:
        mov       ax,@data
        mov       ds,ax
getloop:
        mov       ah,8
        int       21h
        mov       dl,al
        mov       ah,2
        int       21h

        cmp       dl,[end_char]
        jne       getloop
;---------------------------------------
exit:
        mov       ax,4c00h
        int       21h
        end       start
;---------------------------------------
```

# copyfile

- Read character
  - Use `int 21h`
  - Code for ‘input char’ is 8 (in ah)
  - Result in al

```
;---------------------------------------
        .model    small
        .8086
        .stack    256
;---------------------------------------

        .data
;---------------------------------------
end_char db       0Dh
;---------------------------------------

        .code
;---------------------------------------
start:
        mov       ax,@data
        mov       ds,ax
getloop:
        mov       ah,8
        int       21h
        mov       dl,al
        mov       ah,2
        int       21h

        cmp       dl,[end_char]
        jne       getloop
;---------------------------------------
exit:
        mov       ax,4c00h
        int       21h
        end       start
;---------------------------------------
```

# copyfile

- Output character just read
  - Code for ‘output char’ is 2
  - Outputs char in dl
  - Character just read in al

```
;---------------------------------------
        .model    small
        .8086
        .stack    256
;---------------------------------------

        .data
;---------------------------------------
end_char db       0Dh
;---------------------------------------

        .code
;---------------------------------------
start:
        mov       ax,@data
        mov       ds,ax
getloop:
        mov       ah,8
        int       21h
        mov       dl,al
        mov       ah,2
        int       21h

        cmp       dl,[end_char]
        jne       getloop
;---------------------------------------
exit:
        mov       ax,4c00h
        int       21h
        end       start
;---------------------------------------
```

# copyfile

- Termination check
  - Compare char read to end_char
  - Read char is in
    - dl
    - al

```
;---------------------------------------
        .model    small
        .8086
        .stack    256
;---------------------------------------

        .data
;---------------------------------------
end_char db       0Dh
;---------------------------------------

        .code
;---------------------------------------
start:
        mov       ax,@data
        mov       ds,ax
getloop:
        mov       ah,8
        int       21h
        mov       dl,al
        mov       ah,2
        int       21h

        cmp       dl,[end_char]
        jne       getloop
;---------------------------------------
exit:
        mov       ax,4c00h
        int       21h
        end       start
;---------------------------------------
```

# copyfile

- Termination check
  - Compare char read to end_char
  - Read char is in
    - dl
    - ~~al~~
  - False
- int 21h
  - May change value in ax
  - Which includes al

```
;---------------------------------------
        .model    small
        .8086
        .stack    256
;---------------------------------------

        .data
;---------------------------------------
end_char db       0Dh
;---------------------------------------

        .code
;---------------------------------------
start:
        mov       ax,@data
        mov       ds,ax
getloop:
        mov       ah,8
        int       21h
        mov       dl,al
        mov       ah,2
        int       21h

        cmp       dl,[end_char]
        jne       getloop
;---------------------------------------
exit:
        mov       ax,4c00h
        int       21h
        end       start
;---------------------------------------
```

# copyfile

- Exit program
  - Code ‘exit’ is 4C (ah)
  - Return value is 0 -- success (al)