Assembly

10.13.11

- get a character
  check to see if a char is = backspace

- to erase a letter, display a backspace and then a space, & then another backspace, reduce number of characters, then get another character

- if the limit has been reached, get another char
  if not, display char then increment count

- if carriage return, done with string
  compare al to 0Dh or 13d (did they hit enter?)

- no leading zeros, can only enter + or - before the number

- Allocating local storage in a method

```
int funct(int iVal, int iNum)
{
    int s, t;
    [code]
    return;
}
```

OR

```
funct proc Near32
    push ebp
    mov ebp, esp
    sub esp, 8
    s equ [ebp-4]
    t equ [ebp-8]
    iVal equ [ebp+8]
    iNum equ [ebp+12]
```

```
mov eax, iVal
add eax, iNum
mov s, eax
. . . . . .
add esp, 8
pop ebp
ret
```

```
int f(int iNum, short sVal)
{
    int t;
    short s;
    int v;
}
```

```
f proc Near32
    push ebp
    mov ebp, esp
    sub esp, 10
    t equ [ebp-4]
    s equ [ebp-6]
    v equ [ebp-10]
    ...
    add esp, 10
    pop ebp;  ret;
f endp
```

AND op1, op2

clears OF & CF

mostly used for turning off bits using a mask (binary)

sets sign flag if MSB is 1, parity is set

for odd parity - 11100101 - PF=0

expect # of bits to be odd 10101010 - PF=1

CA - PF=0
A7 - PF=0
05 - PF=1

and eax, 0FF0E000h

- OR clears OF, CF
  use OR for turning on bits

- XOR clears OF, CF
  op1 XOR op1 = 0

stc - set parity flag
clc - clear parity flag

xor ax, ax is faster than mov ax, 0

- TEST op1, op2
  op1 is not effected but has same result as AND
  set flags

- - - - - - - - -
  ↑
  is it on

jump to bit6ON ; if 1
jump to bit6OFF ; if 0

test room, 01000000b
jz bit6OFF
jnz bit6ON

jump to bit6 or bit2 on

```
test  room, 01000000b
jnz   bit6orbit2on
test  room, 00000100b
jnz   bit6 or bit2on
```