Addition:

```
.model small
.data
a dw 3629H
b dw 4738H
.code
start:
mov ax,@data
mov ds,ax
mov ax,a
mov bx,b
add al,bl
daa
mov bl,al
adc ah,bh
mov al,ah
daa
mov bh,al
mov ch,04H
mov cl,04H
l2: rol bx,cl
add bl,0fH
mov dl,bl
cmp dl,09H
jbe l4
add dl,07H
l4: add dl,30H
mov ah,02H
int 21H
jnz l2
mov ah,4cH
int 21H
end start
```

Block Transfer:

```
.model small
.code
start:mov SI,0640H
      mov DI,4000H
      mov cl,05H
   l2:mov bl,[SI]
      mov [DI],bl
      inc SI
      inc DI
      dec cl
      jnz l2
      int 03H
      end start
```

Even Odd:

```
.model small
.data
array db 12h,23h,26h,63h,25h,36h,2fh,33h,10h,35h
.code
start: mov ax,@data
       mov ds,ax
       mov cl,10h
       mov si,2000h
       mov di,2010h
       lea bp,array
 next: mov al,ds:[bp]
       mov bl,al
       and al,01h
       jz eve
       mov [di],bl
       inc di
       jmp skip
  eve:mov [si],bl
       inc si
       jmp skip
  skip:inc bp
       dec cl
       jnz next
       int 03H
       end start
```

```
.model small
.code
start:
mov si,2000H
mov di,4000H
mov cl,05H
loop1:
mov bl,[si]
mov [di],bl
inc si
inc di
dec cl
jnz loop1
int 03H
end
start
```

Pack to Unpack:

```
.model small
.data
a db 92H
.code
start:
mov ax,@data
mov ds,ax
mov al,a
and al,0f0H
rcr al,4
mov bh,al
call disp
mov al,a
and al,0fH
mov bh,al
call disp
mov ah,4cH
int 21H
proc disp near
mov ch,02H
mov cl,04H
l4: rol bh,cl
mov dl,bh
and dl,0fH
cmp dl,09
jbe l2
add dl,07
l2: add dl,30H
mov ah,02
```

```
int 21H
dec ch
jnz l4
mov ah,02H
mov dl,' '
int 21H
endp
ret
end start
```

Palindrome:

```
.model small
.data
string db 'madam','$'
string1 db 'string is palindrome','$'
string2 db 'string is not palindrome','$'
.code
start: mov ax,@data
       mov ds,ax
       call Pal
       mov ah,4cH
       int 21H
pal proc near
 mov si,offset string
 loop1: mov al,[si]
        cmp al,'$'
        je label1
        inc si
        jmp loop1

label1: mov di,offset string
        dec si

loop2: cmp si,di
       jl output1
       mov al,[si]
        mov bl,[di]
       cmp al,bl
       jne output2
       inc di

       dec si
       jmp loop2

output1: lea dx,string1
         mov ah,09H
         int 21H
         ret
output2: lea dx,string2
```

```
        mov ah,09H
         int 21H
        ret
    ENDP
    end start
```

Unpack to Pack:

```
.model small
.data
a db 09H
b db 02H
.code
start:mov ax,@data
      mov ds,ax
      mov al,a
      mov bl,b
      rcl al,4
      add al,bl
      mov bh,al
      mov ch,02H
      mov cl,04H
   l2:rol bh,cl
      mov dl,bh
      and dl,0fH
      cmp dl,09H
      jbe l4
      add dl,07H
   l4:add dl,30H
      mov ah,02H
      int 21H
      dec ch
      jnz l2
      mov ah,4cH
      int 21H
      end start
```

Zeros and Ones:

```
.model small
.data
a db 25H
.code
start:mov ax,@data
      mov ds,ax
      mov cl,08h
      mov dx,0000h
      mov al,a
   l2:rcl al,01
```

```
        jc one
        inc dl
        jmp next
  one:inc dh
next:dec cl
        jnz l2
        mov ah,02h
        int 03h
        end start
```