## Computer Structure and Language

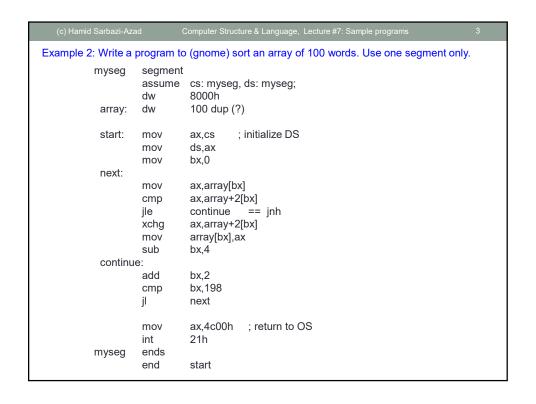
The 8086/8088 Assembly Language

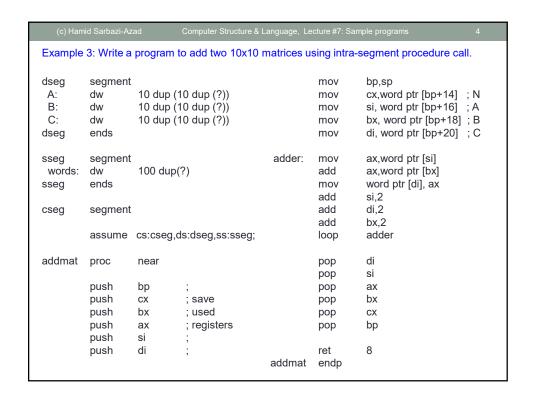
## Hamid Sarbazi-Azad

Department of Computer Engineering Sharif University of Technology (SUT) Tehran, Iran



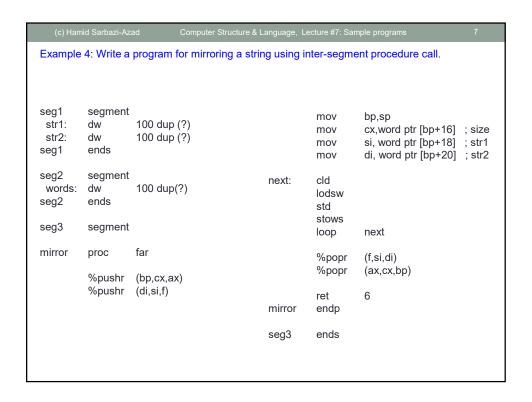
```
Example 1: Write a program to add two 10-byte integers. Translate you program to machine code
    Address
                  Machine Code
                                           Source Code
                                                 segment
    0000
              7777777777777777777
                                                 db
                                                           10 dup (?)
                                        int2:
                                                 db
                                                           10 dup (?)
    0014
                                                 db
                                                           10 dup (?)
                                        sum:-
    001E
                                       dataseg--
                                                 ends
                                                 segment
    0000
                                                          cs: codeseg, ds: dataseg;
                                                 assume
    0000
              B8???? = 10111000 ???...???start:
                                                          ax,dataseg
              8ED8-----= 10001110-11011000-
    0003
                                                          ds.ax
    0005
              BB0000 = 10111011 000 - 000
                                                          bx.0
    8000
              B90A00 = 10111001 00001010 0000 00000v
                                                          cx,10
    000B
                      - 11111000
              al,int1[bx]
    000F
              12470A= 00010010 01000111 0000 1010 adc
                                                          al,int2[bx]
              884714 = 10001000 01000111 0001 0100 mov
    0012
                                                          sum[bx],al
                    = 01000011
    0015
    0016
              E2F4 = 11101111 11110100
                                                          addnext
              B8004C = 10111000 0000 0000 0100 1100mov
                                                          ax,4c00h
    001B
              CD21 = 11001101 0010 0001
    001D
                                       codeseg
                                                 ends
                                                           start
```

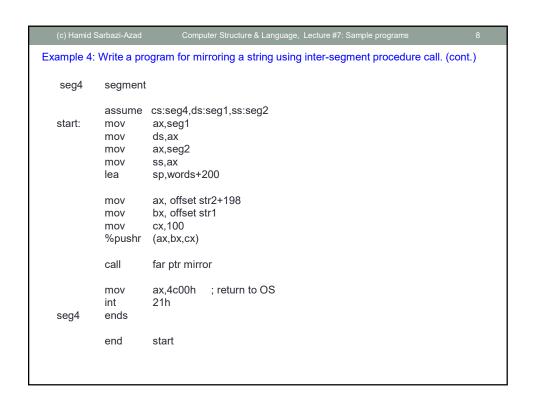




```
Example 3: Write a program to add two 10x10 matrices using procedure external. (cont.)
    start:
                     ax,dseg ; == mov ax, seg B
            mov
            mov
                     ds,ax
                     ax,sseg
            mov
            mov
                     ss,ax
                     sp,words+200 ; == mov sp, offset words+200
            lea
                     ax, offset C
            mov
            push
                     ax
                     ax, offset B
            mov
            push
                     ax
            mov
                     ax, offset A
            push
                     ax
            mov
                     ax, 100
            push
            call
                     near ptr addmat
                     ax,4c00h ; return to OS
            mov
            int
                     21h
   cseg
            ends
            end
                     start
```

```
(c) Hamid Sarbazi-Azad
                        Computer Structure & Language, Lecture #7: Sample programs
Macro Instruction definition
  %*DEFINE (Macro name (parameter list))
      prototype code
                                               Then we can use it as
 )
                                               %pushr (si,bx,cx)
 Example:
                                               %popr (bp,dx,ax)
 %*define (pushr (a,b,c))
                                               That generate the following lines
          push a
          push b
                                                       push si
          push c
                                                       push bx
                                                       push cx
 %*define (popr (a,b,c))
                                                       pop bp
          pop a
                                                       pop dx
          pop b
                                                       pop ax
          pop c
```





```
Example 5: Write a code for moving string1 to string2 using and without using string instructions.
; Without string instructions
                                                ; With string instructions
                                                                  ax,ds
                                                         mov
                  si, offset string1
         mov
                                                                  es,ds
                  di, offset string2
                                                         mov
         mov
                                                                  si, offset string1
                                                         mov
         mov
                  cx, length string1
                                                                  di, offset string2
                  al,[si]
                                                         mov
move: mov
                                                         mov
                                                                  cx, length string1
                  [di],al
         mov
                                                         cld
                  si
         inc
                                                                 string1,string2
                                                    rep movs
                  di
         inc
         loop
                  move
```

```
(c) Hamid Sarbazi-Azad
                        Computer Structure & Language, Lecture #7: Sample programs
Example 6: Write a program to add two 20-byte packed BCD numbers.
  dataseg
                   segment
   BCD1: db
                   20 dup (?)
   BCD2: db
                   20 dup (?)
  dataseg ends
  codeseg segment
          assume cs: codeseg, ds: dataseg;
                   ax,dataseg
   start:
          mov
                   ds,ax
          mov
                   bx,0
          mov
                   cx,20
          mov
          clc
          pushf
                                                       inc
                                                                bx
  nextdigit:
                                                                nextdigit
                                                       loop
          popf
                   al,BCD1[bx]
          mov
                                                       mov
                                                                ax,4c00h
          adc
                   al,BCD2[bx]
                                                       int
                                                                21h
          daa
                                              codeseg ends
          mov
                   BCD1[bx],al
          pushf
                                                       end
                                                                start
```

