```
                        ;Lab5 CSCI 112
                        ;Bee Cha
                        ;Prompt the user to enter 4 fahrenheit temperature
                        ;Using mul and div operation
                        ;convert fahrenheit to celcius temperature
                        ;then calculate the average temperature

                        .586
                        .MODEL FLAT

                        INCLUDE io.h            ; header file for input/output
                       C ; IO.H -- header file for I/O macros (listing suppressed)
                       C .NOLIST      ; turn off listing
                       C .LIST        ; begin listing
                       C

                        .STACK 8192

 00000000               .DATA

 00000000 52 65 73 75 6C   outlbl BYTE   'Results', 0
          74 73 00
 00000008 46 61 68 72 65   label1 BYTE   'Fahrenheit Temperature:', 0      ;23 characters
          6E 68 65 69 74
          20 54 65 6D 70
          65 72 61 74 75
          72 65 3A 00
 00000020 43 65 6C 73 69   label2 BYTE   'Celsius Temperature:', 0         ;20 characters
          75 73 20 54 65
          6D 70 65 72 61
          74 75 72 65 3A
          00
 00000035 54 6F 74 61 6C   label3 BYTE   'Total Average:', 0                     ;14
characters
          20 41 76 65 72
          61 67 65 3A 00
 00000044 45 6E 74 65 72   prompt1       BYTE    "Enter first temperature", 0
          20 66 69 72 73
          74 20 74 65 6D
          70 65 72 61 74
          75 72 65 00
 0000005C 45 6E 74 65 72   prompt2       BYTE    "Enter second temperature", 0
          20 73 65 63 6F
          6E 64 20 74 65
          6D 70 65 72 61
          74 75 72 65 00
 00000075 45 6E 74 65 72   prompt3       BYTE   "Enter third temperature", 0
          20 74 68 69 72
          64 20 74 65 6D
          70 65 72 61 74
          75 72 65 00
 0000008D 45 6E 74 65 72   prompt4       BYTE   "Enter fourth temperature", 0
          20 66 6F 75 72
          74 68 20 74 65
          6D 70 65 72 61
          74 75 72 65 00
```

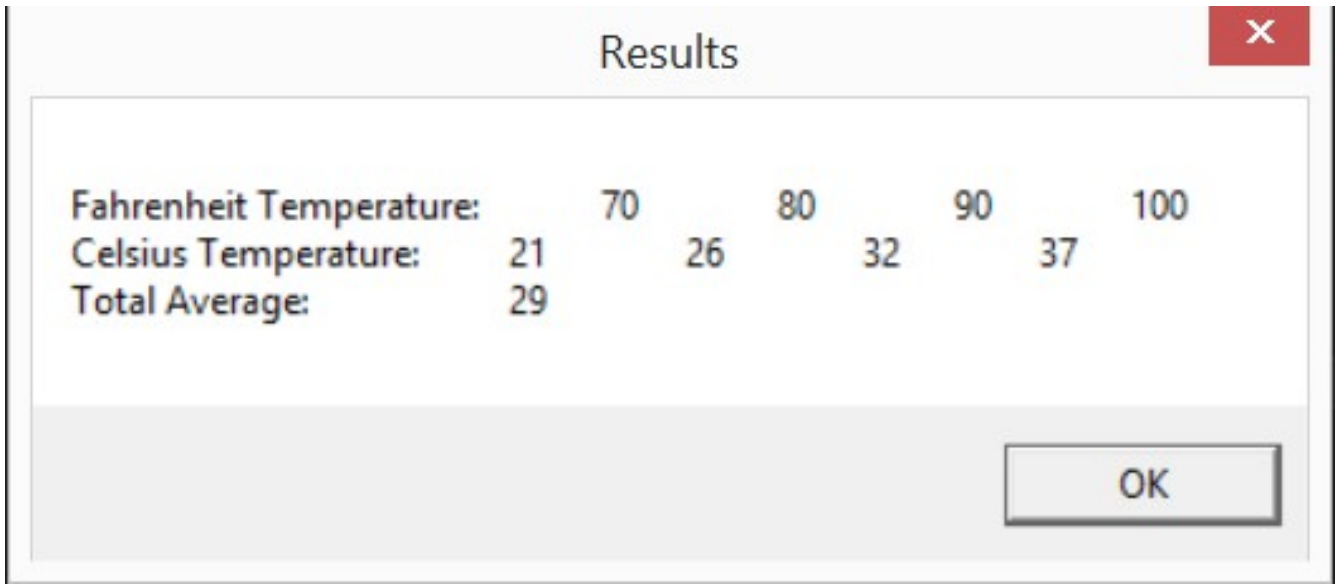
```
000000A6  00000028 [             string  BYTE    40 DUP (?)
          00
         ]
000000CE  00000190 [             outstr BYTE   400 DUP (?)
          00
         ]
0000025E  0000000B [             f1            BYTE   11 DUP (?)
          00
         ]
00000269  0000000B [             f2            BYTE   11 DUP (?)
          00
         ]
00000274  0000000B [             f3            BYTE   11 DUP (?)
          00
         ]
0000027F  0000000B [             f4            BYTE   11 DUP (?)
          00
         ]
0000028A 00000005         five   DWORD  5
0000028E 00000009         nine   DWORD  9
00000292  0000000B [             avg           DWORD  11 DUP (?)
          00000000
         ]

00000000                  .CODE
00000000                  _MainProc PROC

                          ;20h = space
                          ;0dh = feedline
                          ;0ah = newline
                          ;9h  = horizontal tab

                          ;----------Formula for conversion----------
                          ;             (5 * F -  32) / 9

                                ;-----Preparing String Output-----;
00000000  8D 35 00000008 R              lea           esi, label1
      ;Points to first address of label1
00000006  8D 3D 000000CE R              lea           edi, outstr
      ;Beginning of our string storage
0000000C  FC                            cld
0000000D  B9 00000017                         mov           ecx, 23
                  ;to copy 23 characters from label1
00000012  F3/ A4                        rep           movsb
00000014  C6 05 000000E5 R              mov           outstr+23, 9h                     ;tab
         09

                                ;-----First Number-----;
                                input  prompt1, string, 40              ;Input first
temperature reading
                                atod   string
      ;Convert ascii reading to DWORD then it moves it to EAX register

00000048  8D 35 000000A6 R              lea           esi, string
0000004E  8D 3D 000000E6 R              lea           edi, outstr+24
00000054  FC                            cld
00000055  B9 00000002                         mov           ecx, 2
0000005A  F3/ A4                        rep           movsb

0000005C  83 E8 20                      sub           eax, 32
```

```
            ;(F - 32)
0000005F  F7 25 0000028A R              mul           five
      ;(5 * F - 32)
00000065  F7 35 0000028E R              div           nine
      ;(5 * F - 32) / 9
                                     ;[edx = remainder][eax = quotient]
                                        ;Ignore remainder fraction part for now
0000006B  A3 00000292 R                 mov           avg, eax
      ;to accmulate average
                                        dtoa   f1, eax
      ;don't forget to convert to ascii

                                    ;-----Second Number-----;
                                        input  prompt2, string, 40              ;Input
second temperature reading
                                        atod   string
      ;Convert to DWORD

000000B5  C6 05 000000E8 R              mov           outstr+26, 9h
          09
000000BC  8D 35 000000A6 R              lea           esi, string
000000C2  8D 3D 000000E9 R              lea           edi, outstr+27
000000C8  FC                            cld
000000C9  B9 00000002                          mov           ecx, 2
000000CE  F3/ A4                        rep           movsb

000000D0  83 E8 20                      sub           eax, 32
000000D3  F7 25 0000028A R              mul           five
000000D9  F7 35 0000028E R              div           nine
000000DF  01 05 00000292 R              add           avg, eax
                                        dtoa   f2, eax

                                    ;-----Third Number-----;
                                        input  prompt3, string, 40              ;Input third
temperature reading
                                        atod   string
      ;Convert to DWORD

0000012A  C6 05 000000EB R              mov           outstr+29, 9h
          09
00000131  8D 35 000000A6 R              lea           esi, string
00000137  8D 3D 000000EC R              lea           edi, outstr+30
0000013D  FC                            cld
0000013E  B9 00000002                          mov           ecx, 2
00000143  F3/ A4                        rep           movsb

00000145  83 E8 20                      sub           eax, 32
00000148  F7 25 0000028A R              mul           five
0000014E  F7 35 0000028E R              div           nine
00000154  01 05 00000292 R              add           avg, eax
                                        dtoa   f3, eax
                                        ;output       outlbl, f3

                                    ;-----Fourth Number-----;
                                        input  prompt4, string, 40              ;Input
fourth temperature reading
                                        atod   string
      ;Convert to DWORD

0000019F  C6 05 000000EE R              mov           outstr+32, 9h
```

```
         09
000001A6  8D 35 000000A6 R              lea           esi, string
000001AC  8D 3D 000000EF R              lea           edi, outstr+33
000001B2  FC                            cld
000001B3  B9 00000003                          mov           ecx, 3
000001B8  F3/ A4                        rep           movsb
000001BA  C6 05 000000F2 R              mov           outstr+36, 0dh
         0D

000001C1  83 E8 20                      sub           eax, 32
000001C4  F7 25 0000028A R              mul           five
000001CA  F7 35 0000028E R              div           nine
000001D0  01 05 00000292 R              add           avg, eax
                                        dtoa   f4, eax
                                        ;output       outlbl, f4


                                ;-----Append Celcius String-----;

000001EE  8D 35 00000020 R              lea           esi, label2
      ;Points to first address of label2
000001F4  8D 3D 000000F3 R              lea           edi, outstr+37
000001FA  FC                            cld
000001FB  B9 00000014                          mov           ecx, 20
                   ;to copy 23 characters from label1
00000200  F3/ A4                        rep           movsb
00000202  C6 05 00000107 R              mov           outstr+57, 9h
         09

                                ;-----Put all four temperature into output string-----;

00000209  8D 35 0000025E R              lea           esi, f1
0000020F  8D 3D 00000108 R              lea           edi, outstr+58
00000215  FC                            cld
00000216  B9 0000000B                          mov           ecx, 11
0000021B  F3/ A4                        rep           movsb
0000021D  C6 05 00000113 R              mov           outstr+69, 9h
         09

00000224  8D 35 00000269 R              lea           esi, f2
0000022A  8D 3D 00000114 R              lea           edi, outstr+70
00000230  FC                            cld
00000231  B9 0000000B                          mov           ecx, 11
00000236  F3/ A4                        rep           movsb
00000238  C6 05 0000011F R              mov           outstr+81, 9h
         09

0000023F  8D 35 00000274 R              lea           esi, f3
00000245  8D 3D 00000120 R              lea           edi, outstr+82
0000024B  FC                            cld
0000024C  B9 0000000B                          mov           ecx, 11
00000251  F3/ A4                        rep           movsb
00000253  C6 05 0000012B R              mov           outstr+93, 9h
         09

0000025A  8D 35 0000027F R              lea           esi, f4
00000260  8D 3D 0000012C R              lea           edi, outstr+94
00000266  FC                            cld
00000267  B9 0000000B                          mov           ecx, 11
0000026C  F3/ A4                        rep           movsb
```

```
 0000026E  C6 05 00000137 R              mov           outstr+105, 0dh
          0D

                                        ;output       outlbl, outstr

                                    ;-----Calculate Average-----;
 00000275  A1 00000292 R                mov           eax, avg
 0000027A  99                           cdq
 0000027B  BB 00000004                         mov           ebx, 4
 00000280  F7 F3                        div           ebx
                                        dtoa   avg, eax

 0000029A  8D 35 00000035 R             lea           esi, label3
 000002A0  8D 3D 00000138 R             lea           edi, outstr+106
 000002A6  FC                           cld
 000002A7  B9 0000000E                         mov           ecx, 14
 000002AC  F3/ A4                       rep           movsb

 000002AE  C6 05 00000146 R             mov           outstr+120, 9h
          09

 000002B5  8D 35 00000292 R             lea           esi, avg
 000002BB  8D 3D 00000147 R             lea           edi, outstr+121
 000002C1  FC                           cld
 000002C2  B9 0000000B                         mov           ecx, 11
 000002C7  F3/ A4                       rep           movsb

                                        output outlbl, outstr

                                        ;output       outlbl, avg

 000002E2  B8 00000000                   mov     eax, 0  ; exit with return code 0
 000002E7  C3                        ret
 000002E8                    _MainProc ENDP
                             END                            ; end of source code

Microsoft (R) Macro Assembler Version 12.00.30501.0        03/03/15 03:01:38
lab5.asm                                                 Symbols 2 - 1



Macros:

                N a m e                 Type

atod . . . . . . . . . . . . . .  Proc
atow . . . . . . . . . . . . . .  Proc
dtoa . . . . . . . . . . . . . .  Proc
input  . . . . . . . . . . . . .  Proc
output . . . . . . . . . . . . .  Proc
wtoa . . . . . . . . . . . . . .  Proc


Segments and Groups:

                N a m e                 Size     Length   Align   Combine Class

FLAT . . . . . . . . . . . . . .  GROUP
```

```
STACK  . . . . . .  32 Bit  00002000 Para         Stack         'STACK'
_DATA  . . . . . .  32 Bit  000002BE Para         Public  'DATA'
_TEXT  . . . . . .  32 Bit  000002E8 Para         Public  'CODE'


Procedures, parameters, and locals:

                N a m e                 Type     Value    Attr

_MainProc  . . . . . .  P Near  00000000 _TEXT      Length= 000002E8 Public


Symbols:

                N a m e                 Type     Value    Attr

@CodeSize  . . . . . .  Number  00000000h
@DataSize  . . . . . .  Number  00000000h
@Interface . . . . . .  Number  00000000h
@Model . . . . . .  Number  00000007h
@code  . . . . . .  Text           _TEXT
@data  . . . . . .  Text           FLAT
@fardata?  . . . . . .  Text           FLAT
@fardata . . . . . .  Text           FLAT
@stack . . . . . .  Text           FLAT
_getInput  . . . . . .  L Near  00000000 FLAT       External
_showOutput  . . . . . .  L Near  00000000 FLAT       External
atodproc . . . . . .  L Near  00000000 FLAT       External
atowproc . . . . . .  L Near  00000000 FLAT       External
avg  . . . . . .  DWord   00000292 _DATA
dtoaproc . . . . . .  L Near  00000000 FLAT       External
f1 . . . . . .  Byte    0000025E _DATA
f2 . . . . . .  Byte    00000269 _DATA
f3 . . . . . .  Byte    00000274 _DATA
f4 . . . . . .  Byte    0000027F _DATA
five . . . . . .  DWord   0000028A _DATA
label1 . . . . . .  Byte    00000008 _DATA
label2 . . . . . .  Byte    00000020 _DATA
label3 . . . . . .  Byte    00000035 _DATA
nine . . . . . .  DWord   0000028E _DATA
outlbl . . . . . .  Byte    00000000 _DATA
outstr . . . . . .  Byte    000000CE _DATA
prompt1  . . . . . .  Byte    00000044 _DATA
prompt2  . . . . . .  Byte    0000005C _DATA
prompt3  . . . . . .  Byte    00000075 _DATA
prompt4  . . . . . .  Byte    0000008D _DATA
string . . . . . .  Byte    000000A6 _DATA
wtoaproc . . . . . .  L Near  00000000 FLAT       External

          0 Warnings
          0 Errors
```


Results
Fahrenheit Temperature: 70 80 90 100
Celsius Temperature: 21 26 32 37
Total Average: 29
OK