



Electrical and Computer Engineering Department
 University of Maryland
 College Park, MD 20742-3285

Glenn L. Martin Institute of Technology ♦ A. James Clark School of Engineering

Dr. Charles B. Sileo, Jr.
 Telephone 301-405-3668
 Fax 301-314-9281
 sileo@umd.edu

ENEE 350 Homework Set 7

Programming Project 2

(Due: Class 17, Thurs., Oct. 25, 2012)

Write, assemble and run successfully on the simulator a Mac-1 subroutine **lgneg(n,x)** that returns in the AC the address of the integer possessing the algebraically largest negative value along the real line among the n integers in the array whose starting address is x. The largest negative value on the real line is the farthest right value to the left of zero. If there are no negative values among the n elements to be processed, then return -1 which is equivalent to the unsigned integer address 65535, clearly not a valid memory address. If there are two or more array entries that equally satisfy the requirements, return the address of the one with the highest address. Your subroutine should be tested with the main program shown below, which defines how the parameters are passed.

```

/main program | /continued from below halt
      EXTRN lgneg | data 57
ans1  RES 1 | | 0
ans2  RES 1 | | 129
ans3  RES 1 | | 34
n1    6 | | 8
n2    10 | | 3
n3    5 | | -29
start loco 4020 | | -15
      swap /initialize sp | | -2
      loco n1 | | -347
      push /push address n1 | | -3
      loco data | | 6
      push /push array start address | | 35
one   call lgneg | | -413
      stod ans1 | | END start
      insp 2 | |
      loco n2 /push address n2 | |
      push | |
      loco data | |
      addd (4) | |
      push /push array start address | |
two   call lgneg | |
      stod ans2 | |
      insp 2 | |
      loco n3 /push address n3 | |
      push | |
      loco data | |
      addd (9) | |
      push /push array start address | |
three call lgneg | |
      stod ans3 | |
      insp 2 | |
      halt | |
/data array continues here but | |
/ is shown in the above right hand column | |

```

Hand in a copy of the main program symbolic assembly listing, the subroutine symbolic assembly listing, the contents of (macro) memory after “load main sub” (i.e., of main.abs) before execution of the program, and the contents of memory after execution of the program. Highlight and comment upon the final answers. Specify what values are contained in the addresses specified by ans1, ans2, and ans3.