Write, assemble and run successfully on the simulator a Mac-1 subroutine \( \text{maxm}(n,x) \) that returns in the AC the address of the integer possessing the largest absolute value (i.e., magnitude) among the \( n \) integers in the array whose starting address is \( x \). Your subroutine should be tested with the main program shown below, which defines how the parameters are passed.

```assembly
EXTRN maxm

ans1 RES 1
ans2 RES 1
ans3 RES 1
n1 7
n2 10
n3 6

start loco 4020

/initialize sp
loco n1
push /push address n1
loco data
push /push array start address

one call maxm
stod ans1
insp 2
loco n2 /push address n2
push
loco data
add (4)
push /push array start address

two call maxm
stod ans2
insp 2
loco n3 /push address n3
push
loco data
add (6)
push /push array start address

three call maxm
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.