Assembly
Assign #6
Hamer
Due: 11/29/23

The greatest common divisor (GCD) of two integers is the largest positive integer that divides both evenly. Thus GCD(100, 240) is 20, GCD(34, 27) is 1 and GCD(0, 14) is 14. A standard elementary programming exercise computes the GCD of M and N (both > 0) without using division (which is slow) by the following method:

```
while (M<>N) do
      if (N > M) then N = N - M else M = M - N;
GCD = N;
```

It is easy to see that this method works. Since M and N start out nonzero, they remain nonzero, and every time thru the loop the larger gets smaller. Therefore the loop either terminates or they both reach 1, in which case the loop terminates. It should be easy to see that the GCD of M and N after the loop is the same as before the loop, and that GCD(N,N)=N;

Write a separately assembled procedure GCD to compute the GCD of the integers in AX and BX using the algorithm and return the value in AX. Treat the cases M or N = 0 or negative as special cases. (GCD(-M,N) = GCD(M,N); GCD(-32768,-32768)= -32768, GCD(-32768,N) = GCD(16384,N) if N <> -32768. GCD(0,0) should be infinity, but return 0.) Be sure to test your procedure as completely as possible.

You will need to write and submit two programs. The first is your GCD procedure in a separate file and the second will be the program that tests your GCD procedure. Your test program must loop and allow the user to run the GCD procedure on two inputs. You will stop the program by asking the user if they want to continue and checking their response to an input of Y/N.

Be sure to document your code and use procedures when ever possible in this program. Remember that the main procedure answers the question "What does the program do?" and the procedures tell how the task is performed.

As always, submit your code to the assign 6 dropbox on D2L. You will submit two files for this assignment lnameA6.asm and XYZGCD.asm where XYZ are your initials (use two letters if you do not have a middle initial).