

102-A02 Computing the Lesser Common Multiple (LCM)

Work to do

- You will add the following prototype to **tools.h**
`int lcm (int a , int b);`
- You will implement the following function in **tools.c**
- You will invoke this function when menu option 2 is chosen from the **main.c** *main* function.
- You will invoke this function from **tests.c** several times with different parameters and test the return value to make sure it's working completely. Consider this as an implementation of a test-harness that will run automatically.

You need to write a function named *lcm* which computes and returns the lesser common multiple of two positive integers *a* and *b* (parameters). The LCM of two positive integers *a* and *b* is the smallest positive integer *c* such as the modulo of *c* by *a* and the modulo of *c* by *b* are both zero.

If one or both of the parameters are non valid (e.g. negative), your function will simply return -1. The code in the *main* functions in **main.c** and **tests.c** will always check if the call went ok by comparing the return value to -1, if it's equal, the parameters were invalid otherwise the function did its job. Make sure you include such scenario in your test harness.

Example(s)

Here are some examples of return values when calling *lcm*;

```
lcm ( 10 , 20 )    →    20
lcm ( 3 , 9 )      →    9
lcm ( 5 , 9 )      →   45
lcm ( -1 , 3 )     →   -1
```

Hints

- n/a

Testing

Input		Output	
a	b	Expected	Observed