Grid

Write a Python program called `grid.py` that will create a square grid from the argument given on the command line. The program will expect one positional argument; if the number of arguments is not exactly one, print a "usage" statement and exit *with an error code*. The test suite will only provide integer values, so you can assume it is safe to use `int` to convert the input from a string to an integer. The number provided must be in the range of 2 to 9 (inclusive). If it is not, print "NUM () must be between 1 and 9" and exit *with an error code*. You will square the given number to create a grid (so think of the number as how many rows and columns).

Expected Behavior

```
$ ./grid.py
Usage: grid.py NUM
$ ./grid.py 1
NUM (1) must be between 1 and 9
$ ./grid.py -1
NUM (-1) must be between 1 and 9
$ ./grid.py 10
NUM (10) must be between 1 and 9
$ ./grid.py 2
  1  2
  3  4
$ ./grid.py 5
  1  2  3  4  5
  6  7  8  9 10
 11 12 13 14 15
 16 17 18 19 20
 21 22 23 24 25
$ ./grid.py 8
  1  2  3  4  5  6  7  8
  9 10 11 12 13 14 15 16
 17 18 19 20 21 22 23 24
 25 26 27 28 29 30 31 32
 33 34 35 36 37 38 39 40
 41 42 43 44 45 46 47 48
 49 50 51 52 53 54 55 56
 57 58 59 60 61 62 63 64
```

Tests

A passing test suite looks like this:

```
$ make test
python3 -m pytest -v test.py
============================= test session starts ==============================
platform darwin -- Python 3.7.0, pytest-3.8.0, py-1.6.0, pluggy-0.7.1 -- /anaconda3/bin/pyth
cachedir: .pytest_cache
rootdir: /Users/kyclark/work/worked_examples/03-python-grad, inifile:
plugins: remotedata-0.3.0, openfiles-0.3.0, doctestplus-0.1.3, arraydiff-0.2
collected 4 items

test.py::test_exists PASSED                                              [ 25%]
test.py::test_usage PASSED                                               [ 50%]
test.py::test_bad_input PASSED                                           [ 75%]
test.py::test_grid PASSED                                                [100%]

=========================== 4 passed in 0.29 seconds ===========================
```