

# CS 106A Practice Diagnostic

---

*This practice diagnostic looks very similar to what the real diagnostic will look like, both in terms of material covered and structure.*

## Short Answers

Parts A & B refer to the following (buggy) program:

```
"""
This program counts the number of multiples of either 3 or 5
that are less than or equal to 30 and prints this count
"""

def count_multiples(n):
    """
    Counts the number of multiples of 3 or 5 that
    are less than or equal to n.
    """
    counter = 0
    for i in range(1, n + 1):
        if i % 3 == 0:
            counter += 1
        if i % 5 == 0:
            counter += 1
    return counter

def main():
    n = 30
    count_multiples(n)

if __name__ == "__main__":
    main()
```

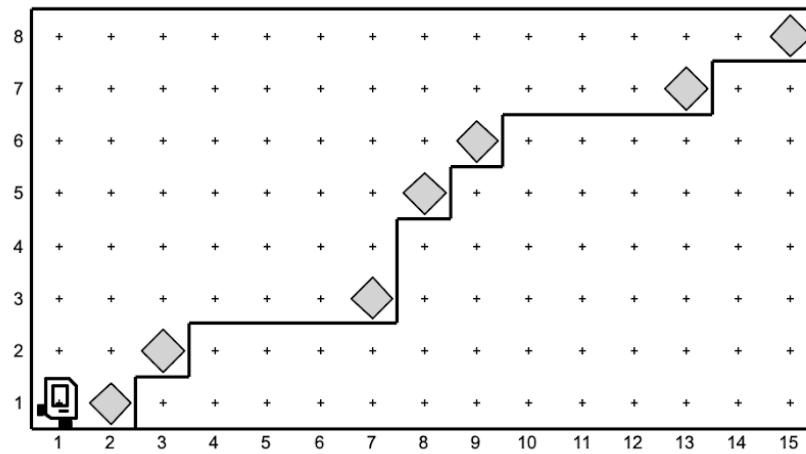
- A. When run, the program does not produce the desired output. Why not?
- B. How can this program be fixed to print the correct number? You may change either `count_multiples` or `main`, but `count_multiples` must still implement the functionality described in its comment for all possible values of its parameter.

## Short Program

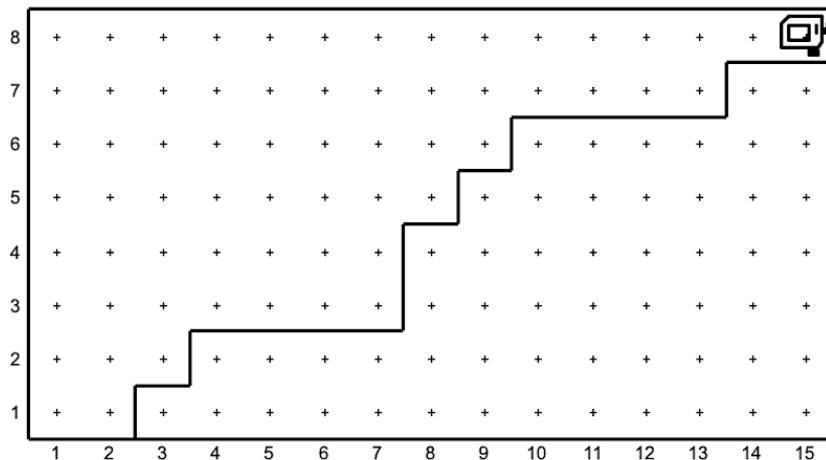
Write a program that reads in two numbers from the user and prints the larger number. You can assume the user types in two distinct real numbers.

## Flower Picking Karel

Karel is standing at the base of a mountain in the spring and wants to climb it, stopping along the way to collect any flowers it comes across. Karel's world looks something like this:



The mountain consists of a series of 'steps' each of which is at least one street high and one avenue across, and flowers are represented by beepers. A flower is represented by a single beeper, and each step has a flower at its base. Karel's job is to climb this mountain, picking up the beepers along the way, to result in a world that looks like this:



Write a program to help Karel do this. Keep in mind the following considerations:

- Your program should work for any world and any sized steps.
- The peak of the mountain is guaranteed to be at the top right corner of the world.
- Karel begins at (1,1) facing East.

Your solution should use no non-Karel features (i.e. variables, break statements, etc)

## Python

Write a program that reads in integers one by one from the user and prints out whether those integers are perfect. A perfect number is a number whose factors sum is the same as itself. For example, 28 is a perfect number, since its factors (1, 2, 4, 7 and 14) sum to 28. On the other hand, 42's factors (1, 2, 3, 6, 7, 14, 21) sum to 58, and so it isn't a perfect number. Your program should print an error message if the number entered is negative, and should end when the user types 0. A sample run of the program is below (user input is in *italic*):

```
Your number: -12
Please enter a positive number
Your number: 28
28 is perfect!
Your number: 42
42 is not perfect!
Your number: 0
```