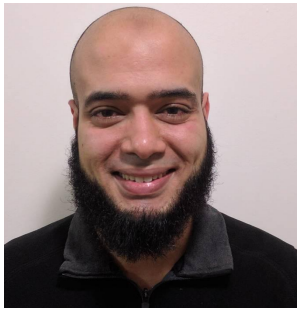
Python Programming

Special Methods

Homework 1

Mostafa S. Ibrahim
Teaching, Training and Coaching since more than a decade!

Artificial Intelligence & Computer Vision Researcher
PhD from Simon Fraser University - Canada
Bachelor / Msc from Cairo University - Egypt
Ex-(Software Engineer / ICPC World Finalist)



Problem #1: Fraction

```
f1 = Fraction(4, 8)
print(f1)       # 1/2
print(f1 * f1)  # 1/4

f1 = Fraction(3, 8)
f2 = 2 * f1
f3 = f1 * Fraction(5, 4) * 16
print(f1, f2, f3)
# 3/8 3/4 15/2

print(f1 == f2) # False
print(f2 > f1)  # True
print(f1.value) # 0.375
```

- Implement fraction class that support the requested operations.
  - Note: init should receive integer only
  - Feel free to go deeper
- Math hint: To simplify a fraction, you need greatest common divisor (gcd)
  - Import math ⇒ math.gcd
- You don’t need to handle a lot of intentional mistakes from the user
  - Eg. change numerator to a string

Problem #2: Vector

```
if __name__ == '__main__':
    v1 = Vector(1, 2, 3, 4)
    v2 = Vector(4, 5, 1, -2)

    print(v1 + v2)  # [5, 7, 4, 2]
    print(v1 <= v2)  # True
```

- Implement a simple vector class to add vectors or compare them
  - Feel free to go deeper
- Raise an error if added vectors of unequal length
- You don’t need to handle intentional mistakes from the user

Problem #3: MyFloat!

```
f1 = 1 + 3/7 -1
f2 = 3/7

# False 0.4285714285714286 0.42857142857142855
print(f1 == f2, f1, f2)

f1 = MyFloat(f1)
f2 = MyFloat(f2)
print(f1 == f2) # True
print(f1 < f2)  # False
print(f1 >= f2) # True
```

- Have u ever failed to compare floats directly?
  - This is due to double representation
  - More worse: dct won't work
- Never compare floats directly
- Create float class that supports comparisons
  - One way to check if 2 float numbers are equal is to check if their abs is a very small value
- *Note: you still can't use with dict*

"Acquire knowledge and impart it to the people."

"Seek knowledge from the Cradle to the Grave."