

CSE 20
Fall 2021
Quiz 5

Solutions

1. (30 Points) The Vector class below represents vectors in the x-y plane. The initialization function `__init__(self)`, which is included, defines a Vector as having two attributes: `xcomp` and `ycomp`. Fill in definitions for the two built-in functions: `__str__(self)`, and `__eq__(self, other)` and the three instance methods: `add(self, other)`, `sub(self, other)`, and `dot(self, other)`. Your definitions must satisfy the conditions stated in the accompanying doc strings. If the Vector class is implemented in a module called `vector.py`, then the following print statements should produce the output given in the corresponding comments. (Each function is worth 6 points.)

```
from vector import *
u = Vector(2, 5)
v = Vector(-1, 7)
print(u==v)          # False
print(u.add(v))      # (1, 12)
print(u.sub(v))      # (3, -2)
print(v.sub(u))      # (-3, 2)
print(u.dot(v))      # 33
```

Implementation file: `vector.py`

```
class Vector:
    def __init__(self, a, b):
        self.xcomp = a
        self.ycomp = b

    def __str__(self):
        """str(Vector(a, b)) returns the string '(a, b)'."""

        return f'({self.xcomp}, {self.ycomp})'

    def __eq__(self, other):
        """Returns True if and only if u and v have same components."""

        a = self.xcomp
        b = self.ycomp
        c = other.xcomp
        d = other.ycomp
        return a==c and b==d

    def add(self, other):
        """Returns the vector sum: (a,b)+(c,d)=(a+c,b+d)."""

        a = self.xcomp
        b = self.ycomp
        c = other.xcomp
        d = other.ycomp
        return Vector(a+c, b+d)
```

```

def sub(self, other):
    """Returns the vector difference: (a,b)+(c,d)=(a-c,b-d)."""

    a = self.xcomp
    b = self.ycomp
    c = other.xcomp
    d = other.ycomp
    return Vector(a-c, b-d)

def dot(self, other):
    """Returns the vector dot product: (a,b)·(c,d)=ac+bd."""

    a = self.xcomp
    b = self.ycomp
    c = other.xcomp
    d = other.ycomp
    return a*c+b*d

```

2. (20 Points) Continuing the Vector class above, write a class method called `from_string(s)` that takes as input a string `s`, and returns the Vector object. The format of `s` will be as returned by function `__str__()` in the preceding problem. Thus `Vector.from_string('(1.2, 3.4)')` will return a Vector object with `xcomp` equal to 1.2 and `ycomp` equal to 3.4. Thus the expression

```
u==Vector.from_string(str(u))
```

will evaluate to `True` for any Vector `u`.

```

def from_string(s):
    """
    Returns the Vector object represented by string s, i.e. the
    expression u==Vector.from_string(str(u)) is always True.
    """

    # your code begins here

    t = s[1:-1].split(',')      # delete parentheses and comma
    a = float(t[0])              # turn string into number
    b = float(t[1])              # turn string into number
    return Vector(a, b)          # create and return Vector object

    # your code ends here

```