

### Question: [60 Marks]

We provide an implementation of a **Vector class** in the attached file **Vector.py**, representing the coordinates of a vector in a multidimensional space. For example, in a three-dimensional space, we might wish to represent a vector with coordinates 5, -2, 3. Although it might be tempting to directly use a Python list to represent those coordinates, a list does not provide an appropriate abstraction for a geometric vector. In particular, if using lists, the expression  $[5, -2, 3] + [1, 4, 2]$  results in the list  $[5, -2, 3, 1, 4, 2]$ . When working with vectors, if  $u = 5, -2, 3$  and  $v = 1, 4, 2$ , one would expect the expression,  $u + v$ , to return a three-dimensional vector with coordinates 6, 2, 5.

We therefore try to define a Vector class, in Vector.py, that provides a better abstraction for the notion of a geometric vector. Internally, our vector should rely upon an instance of a list, named `_coords`, as its storage mechanism. By keeping the internal list encapsulated, we should be able to enforce the desired public interface for instances of our class.

In Vector.py, you need to define the following magic functions which should work as specified below:

#### Question 1.

- a) [1 Mark] `__len__(self)` : This function should be able to return the dimension of the vector
- b) [1 Mark] `__getitem__(self,j)` : Function should be able to return the jth coordinate of the vector
- c) [1 Mark] `__setitem__(self,j,val)` : Function should be able to set the jth coordinate of vector to val
- d) [2 Marks] `__add__(self,other)` : Function should be able to return sum of two vectors  
Example : If two vectors are  $u = [5, -2, 3]$  and  $v = [1, 4, 2]$  then, output of  $u + v = [6, 2, 5]$
- e) [2 Marks] `__eq__(self,other)` : Function should be able to return True if vector has same coordinates as other
- f) [2 Marks] `__ne__(self, other)` : Function should be able to return True if vector differs from other
- g) [3 Marks] `__str__(self)` : Function should be able to return the string representation of a vector. For example, the string representation of a vector with coordinates 4, 7 and 5 is "<4, 7, 5>"

#### Question

2.

[6 Marks] Implement the sub method for the Vector class in code fragment, so that the expression  $u - v$  returns a new vector instance representing the difference between two vectors.

#### Question

3.

[6 Marks] Implement the neg method for the Vector class in code fragment, so that the

expression  $-v$  returns a new vector instance whose coordinates are all the negated values of the respective coordinates of  $v$ .

**Question**

**4.**

[8 Marks] Implement the `mul` method for the `Vector` class of in code fragment, so that the expression  $v * 3$  returns a new vector with coordinates that are 3 times the respective coordinates of  $v$ .

**Question**

**5.**

[8 Marks] In Question 4. you try to implement `mul` , for the `Vector` class in code fragment, to provide support for the syntax  $v * 3$ . Implement the `rmul` method, to provide additional support for syntax  $3 * v$ .

**Question**

**6.**

[10 Marks] Implement the `mul` method for the `Vector` class in code fragment, so that the expression  $u * v$  returns a scalar that represents the dot product of the vectors, that is,  $\sum_{i=1}^d u_i \cdot v_i$

**Question**

**7.**

[10 Marks] The `Vector` class in code fragment provides a constructor that takes an integer  $d$ , and produces a  $d$ -dimensional vector with all coordinates equal to 0. Another convenient form for creating a new vector would be to send the constructor a parameter that is some iterable type representing a sequence of numbers, and to create a vector with dimension equal to the length of that sequence and coordinates equal to the sequence values. For example, `Vector([4, 7, 5])` would produce a three-dimensional vector with coordinates  $\langle 4, 7, 5 \rangle$ . Modify the constructor so that either of these forms is acceptable. that is, if a single integer is sent, it produces a vector of that dimension with all zeros, but if a sequence of numbers is provided, it produces a vector with coordinates based on that sequence.