Task7

Deadline: 2.12.2015 for 1EN04C and 1.12.2015 for1EN04A

Prepare your programs as separate scripts for each sub-problem.  
**All** your scripts **must** contain description about task and author (you) in comments.  
EXAMPLE:  
# STUDENT: Vasya Pupkin  
# GROUP: 1EN04X  
# TASK: Task7 problem A

# (60pt) Problem A [Polynomial]

Create class **Polynomial** that can store and manipulate polynomial equations. We assume that length of polynomial is less than 10 (i.e. maximum powers are x9).  
*Example of polynomial equation: 4x3+3x2+x+65*Class must have following constructors, methods and attributes:

* constructor that accepts list of coefficients.   
  (hint: m = [3,6,1,7]; p = Polynomial(m); # *7x3+x2+6x+3* )
* **attribute** **coeffs** that is list of integers. (where coefficient index corresponds to x power ***ex: coeffs[3]\*x3***)
* Methods:
  + Overload operator + for two Polynomial objects.  
    **def \_\_add\_\_(self, poly):** # that sums two Polynomials and returns Polynomial
  + Overload operator - for two Polynomial objects.  
    **def \_\_sub\_\_(self, poly):** # that subtracts two Polynomials and returns Polynomial
  + Overload operator \* for two Polynomial objects.  
    **def \_\_mul\_\_ (self, poly):** # that multiplies two Polynomials and returns Polynomial
  + **def calculate(self, x):** # that finds result of this polynomial by substitution of given x
  + **def \_\_str\_\_(self):** # that converts to string representation
* Create **program** that will create multiple objects of Polynomial class and use all previous methods.

|  |
| --- |
| **Console** |
| Polynomial A  Enter size: 7  Enter coefficients: 8 4 0 3 6 2 -3 #note that coeffs are reversed  Polynomial A is:  8x^6 + 4x^5 + 3x^3 + 6x^2 + 2x - 3  Polynomial B  Enter size: 5  Enter coefficients: -1 9 0 -4 -2  Polynomial B is:  -x^4 + 9x^3 – 4x - 2  Sum of A and B is (A+B):  8x^6 + 4x^5 – x^4 + 12x^3 + 6x^2 - 2x - 5  A subtract B is (A-B):  8x^6 + 4x^5 + x^4 - 6x^3 + 6x^2 + 6x - 1  Multiplication of A and B is (A\*B):  -8x^10 + 68x^9 + 36x^8 - 35x^7 - 11x^6 + 44x^5 + 9x^4 - 57x^3 - 20x^2 + 8x + 6  Enter number: 2  Equation B with x=2:  46 |

# (60pt) Problem B [Complex numbers]

Create **class Complex** instances of which has:  
Two integer attributes: **real** and **imaginary**

*(Note: Complex numbers are such numbers that have two parts real and imaginary, whenever you subtract or add two complex numbers you subtract and add each part separately. But when you multiply them each part is multiplied by each of them, and i2is -1. EX: (1+2i)\*(3+4i) = (3+4i + 6i+8i2) = (-5+10i) )*

Write four methods:

1. \_\_init\_\_ - constructor that can accept one, two or no arguments ex: (a + 0i) (a + bi) (0 + 0i)
2. \_\_str\_\_ - that will convert object to string by use Format String   
   ex: “4 + 3i” or “-5 - i” or “- 3i” or “-8” or “0”
3. \_\_add\_\_ / \_\_radd\_\_ - that adds two Complex numbers / int and Complex number then return Complex number
4. \_\_sub\_\_ / \_\_rsub\_\_ - that subtracts two Complex numbers / int and Complex number then return Complex number
5. \_\_mul\_\_ / \_\_rmul\_\_ - that multiplies two Complex numbers / int and Complex number then return Complex number

Write Program that creates objects of Complex numbers and outputs some intermediate results:

|  |
| --- |
| **Console** |
| Number A  Enter real part: **7**  Enter imaginary part: **8**  Complex A number is:  7 + 8i  Number B  Enter real part: **-4**  Enter imaginary part: **9**  Complex B number is:  -4 + 9i  Number C  Enter real part: **-7**  Enter imaginary part:  Complex C number is:  -7  Sum of A and B is:  3 + 17i  Sum of A and C is:  8i  A subtract B is:  11 – i  Multiplication of A and B is:  -100 + 31i |

# (bonus 50pt) Problem C [TurtleRaces]

Your task is to create your own class Turtle, instances of which will hold following attributes:  
strength – the amount of pixels turtle can crawl in one attempt, but it also define randomness of angle  
agility – define speed of turtle when it moves  
stamina – number of attempts turtle will make in one round  
color – string that define color of turtle  
winner – Boolean value that will change if this turtle wins.

You are given following function YOU MUST put it in your program and call it to execute crawling for each round.

|  |
| --- |
| def crawl(avatar,bob):  """ avatar - object of Turtle from turtle module  bob - object of Turtle just created by you. """  def dist(a,b):  return ((a[0]-b[0])\*\*2 + (a[1]-b[1])\*\*2)\*\*0.5  import random  avatar.color(bob.color)  avatar.pensize(3)  avatar.shape("turtle")  avatar.speed(bob.agility)  for i in range(bob.stamina):  avatar.right(random.randint(1,bob.strength\*10))  avatar.forward(bob.strength)  if dist(avatar.pos(),(0,0))>250:  bob.winner = True  return |

Your program should create two objects of Turtle class and two objects of turtle.Turtle class from turtle module. Draw circle of arena with radius of 250 pixels. And keep executing **crawl** function for each of them until one of turtles crawls out of circle.

*(note: Notice names of classes are same, so that you understand how to distinguish classes from different modules)*

|  |  |
| --- | --- |
| **Console Sample** | **Racing Arena** |
| **Welcome to Turtle Races!!!**  **Please select attributes of First hero!**  **Enter STRENGTH(1..36):**25  **Enter AGILITY(1..10):**9  **Enter STAMINA(1..10):**5  **Enter COLOR:**red  **Now select parameters of Second opponent!**  **Enter STRENGTH(1..36):**32  **Enter AGILITY(1..10):**5  **Enter STAMINA(1..10):**10  **Enter COLOR:**green  **Now watch the challenge!!!**  **green turtle is WINNER!!!** |  |