

Ayudantía N°8

Taller de Programación 2018

Diego Caro, Francisca Maron

1. Cree una función que permita contar el número de bases nitrogenadas (a: adenina, t: timina, c: citosina, g: guanina) de una secuencia de ADN. La función debe retornar un diccionario, donde la clave sea la letra correspondiente a la base nitrogenada (a, t, c y g) y el valor corresponde a la cantidad de veces que aparece la base en la secuencia. Para probar la función copie el texto de Gen_Levadura.txt. Este representa un gen de *Saccharomyces cerevisiae*, levadura utilizada para la fabricación de pan, vino y cerveza. (GenBank: <https://www.ncbi.nlm.nih.gov/Sitemap/samplerecord.html>)

2. Cree una lista de diccionarios con los datos de usted más 4 compañeros. El diccionario debe contener nombre, edad y comuna donde vive. Luego cree un programa que imprima la cantidad de personas que tienen su misma edad y vivan en su comuna (incluyéndolo). Ejemplo:

```
Input: [{ 'nombre': 'Francisca', 'edad': 25, 'comuna': 'Providencia' },
        { 'nombre': 'Juanito', 'edad': 26, 'comuna': 'Las Condes' },
        { 'nombre': 'Anita', 'edad': 25, 'comuna': 'Providencia' },
        { 'nombre': 'Pedro', 'edad': 24, 'comuna': 'La Florida' },
        { 'nombre': 'Camila', 'edad': 27, 'comuna': 'Ñuñoa' } ]
Output: 2
```

3. Defina una función que al ingresar una lista de tuplas de tamaño 2, las ordene según el segundo valor de la tupla. Ejemplo (Tuplas de cantidad de personas viviendo con VIH en países de America Latina <http://aidsinfo.unaids.org/>):

```
Input: [ ('Chile', 67000), ('Brasil', 860000), ('Argentina', 120000),
         ('Mexico', 230000) ]
Output: [ ('Brasil', 860000), ('Mexico', 230000), ('Argentina', 120000),
          ('Chile', 67000) ]
```

Hint: Puede utilizar la función sort (<https://docs.python.org/3/howto/sorting.html>) y la función itemgetter del módulo operator (<https://docs.python.org/3/library/operator.html>).

4. Dado x (serie Fibonacci) e y (números primos), realice las siguientes operaciones:

x = set((1,1,2,3,5,8,13))

y = set((2,3,5,7,11,13))

- a. $x \cup y$
- b. $x \cap y$
- c. $x - y$
- d. $y - x$
- e. $x \oplus y$

5.

- **Objetivo:** implementar un tipo de dato para manejar números racionales (fracciones).
- **Conjunto de valores:** ... debe hacerlo usted...
- **Operaciones:**
 - suma, resta, multiplicación y división
 - devolver una representación en string

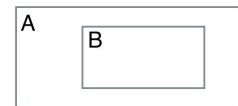
API:

class Rational

	Rational(n, m)	Crea un número racional de numerador n, y denominador m
Rational	plus(Rational b)	Suma de racional actual con b.
Rational	minus(Rational b)	Diferencia de racional actual con b.
Rational	times(Rational b)	Producto de racional actual con b.
Rational	divides(Rational b)	Cociente de racional actual con b.
str	__str__()	Representación en string del número racional

6.

- **Objetivo:** implementar un tipo de dato para manejar rectángulos en el plano. Extienda el código presentado a continuación.
- **Conjunto de valores:** el punto de la esquina inferior-izquierda y el punto superior-derecha del rectángulo.
- **Operaciones:** devolver el área, devolver True/False si dos rectángulos están solapados, devolver el área de intersección, devolver un nuevo rectángulo que sea el área de intersección.
- Proponga una **API** para las operaciones definidas. Rellene las operaciones y el retorno de cada operación en la siguiente tabla:



B está contenido en A



A y B están solapados. Área intersección.

class Rectangle

	Rectangle(...)	Crea un rectángulo..... (completar!)
str	__str__()	

```

1 class Point:
2     """A 2d point in (x, y)."""
3     def __init__(self, x, y):
4         self.x = x
5         self.y = y
6
7     def distX(self, p): return abs(self.x - p.x)
8     def distY(self, p): return abs(self.y - p.y)
9
10 class Rectangle:
11     """A rectangle with the lower-left corner in
12     p1 and upper-right corner in p2."""
13     def __init__(self, p1, p2):
14         self.p1 = p1
15         self.p2 = p2
16
17     def area(self):
18         return self.p1.distX(self.p2) * self.p1.distY(self.p2)
19
20 def main():
21     p1 = Point(-1, -1)
22     p2 = Point(1, 1)
23     r = Rectangle(p1, p2)
24     print('area de r:', r.area())
25
26 if __name__ == '__main__': main()

```

7. Implementar el modulo de matrices utilizando una clase. Diseñe una API que tenga la misma funcionalidad del modulo matriz. Recuerde tomar en cuenta las pre y post condiciones que se deben cumplir. La clase matriz de un módulo llamado algebra.