

Auxiliar Pre Exámen

Profesor: Patricio Nellef Inostroza Fajardin
Auxiliares: Alonso Almendras T., Felipe Avendaño A.,
Vicente Bórquez Z., Matías Godoy C.

P1.- Clases y Objetos

Una relación es un conjunto de pares ordenados, por ejemplo, $R=(1,2),(2,3),(1,4),(3,3),\dots$. Al respecto, se dispone de la clase Relacion con los siguientes métodos:

class Relacion:

```
def __init__(self):  
    self.lista = []  
  
def agregarPar(self, x, y):  
    # Implementación del método agregarPar  
    # ...  
  
def ordenadas(self, x):  
    # Implementación del método ordenadas  
    # ...  
  
def abscisas(self, y):  
    # Implementación del método abscisas  
    # ...
```

En donde self.lista, serían todas las tuplas del conjunto de la relación. Implemente los siguientes métodos:

- agregarPar, el cual agrega un par (x,y) a la lista de tuplas, únicamente si es que esta no estaba previamente en la lista. E imprime si se agregó o si estaba previamente en la lista
- ordenadas, el cual recibe un entero y devuelve una lista de todas las tuplas que tengan a x como primer elemento de la tupla
- abscisas, el cual recibe un entero y devuelve una lista de todas las tuplas que tengan a y como segundo elemento de la tupla.

P2.- Archivos y Strings

Escriba un programa que reciba un archivo llamado P2.txt que contiene una lista de palabras (es decir, una palabra por línea) ordenadas alfabéticamente y que las regrave, en otro archivo llamado respuesta.txt, pero ordenadas por tamaño, es decir, primero la más corta y al final la más larga.

P3.- Listas Recursivas

Cree la función `merge(a,b)` que reciba dos listas enlazadas (recursivas) `a` y `b` que se encuentran ordenadas de manera decreciente. La función debe retornar una nueva lista enlazada ordenada que contenga los elementos de `a` y `b` de forma decreciente.

P4.- Listas indexadas

Cree la función `selectionSort(a)` que recibe una lista indexada y la ordena de la siguiente manera con el algoritmo de selection Sort, el cual sigue los siguientes pasos:

- Partimos buscando el menor valor de la lista, luego, este menor valor, se intercambia con el primer valor de la lista
- Repetimos este proceso, pero ahora ordenando un intervalo más pequeño, dado que sabemos que el primer valor, ahora es el mínimo, por lo que ahora ordenaremos la lista en el intervalo $[1:n[$, luego, $[2:n[$, hasta terminar la lista.