

# Tarea: Algoritmo Set Covering

Lic. Arnoldo Del Toro Peña

15 de febrero de 2022

## Resumen

Uso de un método metahurístico para la solución a un problema del viajero utilizando programación en python.

*Palabras clave: python, viajero, metahurística.*

## SECCIÓN 1

---

### Introducción

En este documento se presentarán los resultados a las instancias obtenidas, se compararán con los resultados óptimos que se presentan a continuación:

<b>Instancias de Prueba</b>	<i>Nombre de la instancia</i>	<i>nodos</i>	<i>Óptimo</i>
76	pr76.tsp	76	108159
105	lin105.tsp	105	14379
280	a280.tsp	280	drilling problem
130	ch130.tsp	130	6110
225	tsp225.tsp	225	3919

Más adelante analizaremos las diferencias entre estos resultados y los que obtuvimos.

## SECCIÓN 2

---

### Descripción

El algoritmo que se pidió fue el del nodo más cercano el cual se menciona en el siguiente diagrama:

Una breve explicación del digrama anterior; podemos observar que una vez que marcamos el nodo se exploran sus conexiones con nodos no marcados y se selecciona el menor y esto se repite hasta marcar todos los nodos.

## SECCIÓN 3

---

### Descripción del algoritmo

El algoritmo se finaliza cuando todos los nodos estan marcados,

---

## Implementación

El algoritmo se programó en lenguaje python con referencias en [Van Rossum und Drake Jr \(1991\)](#), [Van Rossum und Drake Jr \(2017\)](#) y [Chun \(2001\)](#), y se puede verificar en el siguiente repositorio de: [git-hub](#).

---

## Resultados

Los resultados se pueden ver en el mismo enlace de git-hub, en los documentos txt, sin embargo se presentarán a continuación en una forma más ordenada:

<b>Instancias de Prueba</b>	<i>Nombre de la instancia</i>	<i>nodos</i>	<i>Óptimo</i>	<i>Aproximado</i>	<i>error %</i>
76	pr76.tsp	76	108159	137192.62	26.84 %
105	lin105.tsp	105	14379	18383.53	27.89 %
280	a280.tsp	280	drilling problem	4227.54	
130	ch130.tsp	130	6110	7318.21	19.77 %
225	tsp225.tsp	225	3919	4991.92	27.37 %
				Promedio:	25.46 %

---

## Conclusiones

Si observamos los resultados obtenidos en el cuadro 1 podemos observar tanto los resultados obtenidos como su error obtenido en porcentaje, y si nos enfocamos al final de esta misma podemos observar que el promedio es de 575 % el cual es demasiado alto; ahora si observamos los resultados obtenidos en las instancias: spca1 y scpel podemos ver que se obtuvo un porcentaje del 0 % que es muy bueno, pero en las instancias scpd1 y scpb1 los porcentajes son demasiados altos en consecuencia esto contrarresta la eficacia obtenida en las primeras dos mencionadas.

---

## Referencias

- [Chun 2001] CHUN, Wesley: *Core python programming*. Bd. 1. Prentice Hall Professional, 2001
- [Van Rossum und Drake Jr 1991] VAN ROSSUM, Guido ; DRAKE JR, Fred L.: Guía de aprendizaje de Python. In: *Release 2* (1991)
- [Van Rossum und Drake Jr 2017] VAN ROSSUM, Guido ; DRAKE JR, Fred L.: *El tutorial de Python*. Python Software Foundation, 2017