

Tarea Corta 1: William Aguilar Calvo

```
[14]: import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
```

```
[15]: N = pd.DataFrame([10, 20, 40, 80, 160, 320 ], columns=["N"])
P = pd.DataFrame([1, 2, 4, 8, 16, 32, 64, 128 ], columns=["Procesadores"])
```

¿Qué sucede con los valores de speedup y eficiencia a medida que p se incrementa y n se mantiene fijo?

```
[16]: speedUpCaso1 = pd.read_csv("ResultadosSpeedUpCaso1.csv")
eficienciaCaso1 = pd.read_csv("ResultadosEficienciaCaso1.csv")

speedUpCaso2 = pd.read_csv("resultadosSpeedUpCaso2.csv")
eficienciaCaso2 = pd.read_csv("ResultadosEficienciaCaso2.csv")
```

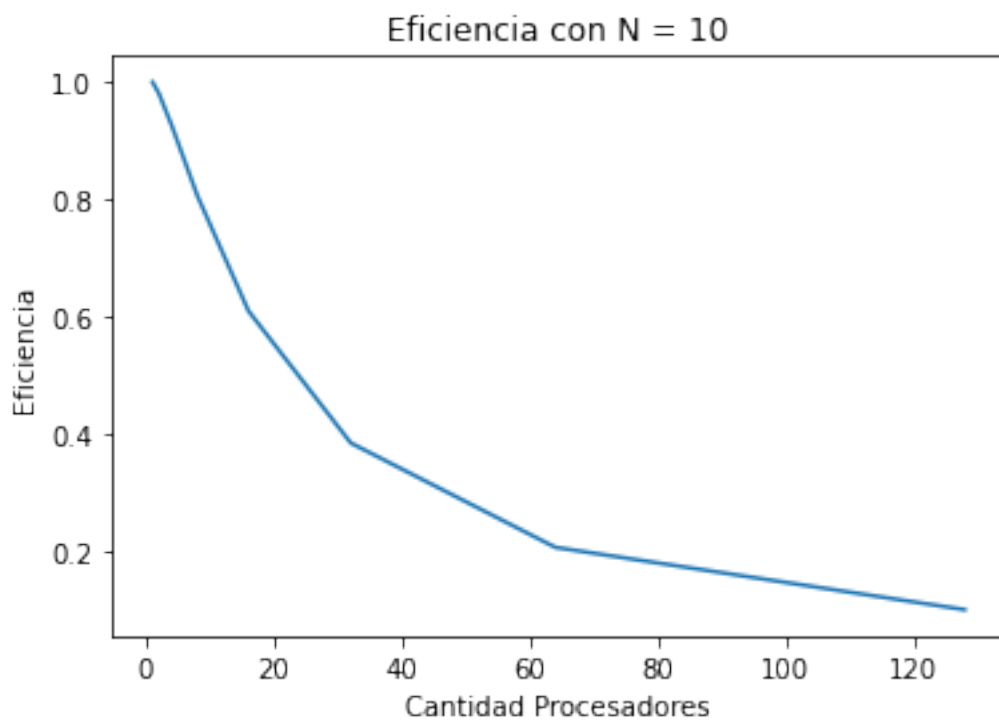
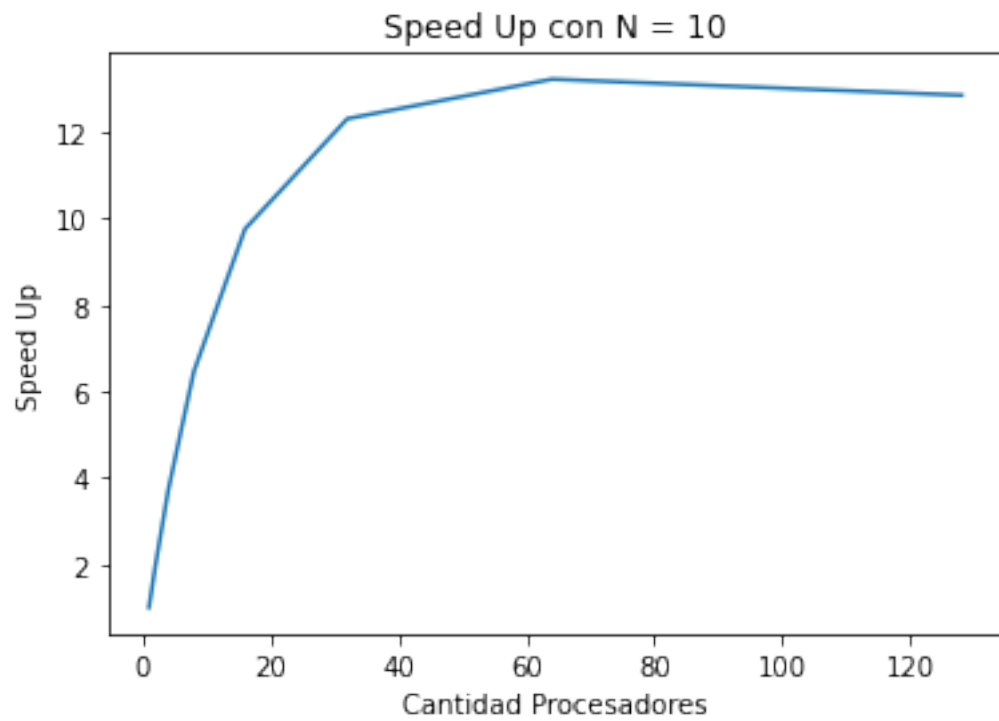
En el caso de cuando la N se mantiene constante aquí escogimos el valor de 10 para realizar los calculos, se puede notar que en el caso del Speed Up tiene un comportamiento de crecimiento logaritmico con su maximo en 60 procesadores, luego de eso tiende a bajar como podemos ver despues de los 60 procesadores, no es mucho lo que baja pero en relacion se puede decir que con 60 procesadores es cuando tiene el mejor SPEED UP

En el caso del calculo de la eficiencia, podemos ver que con menos procesadores la eficiencia es mejor, mientras que a medida que se van aumentando los procesadores la eficiencia va en caída

```
[17]: fig, ax = plt.subplots()
ax.set(xlabel='Cantidad Procesadores', ylabel='Speed Up',
       title='Speed Up con N = 10')
ax.plot(P, speedUpCaso1.iloc[:,1:])

fig, ax = plt.subplots()
ax.set(xlabel='Cantidad Procesadores', ylabel='Eficiencia',
       title='Eficiencia con N = 10')
ax.plot( P, eficienciaCaso1.iloc[:,1:],)
```

```
[17]: [<matplotlib.lines.Line2D at 0x7fe0dc59b880>]
```



¿Qué sucede cuando p se mantiene fijo y n se incrementa?

En el primer grafico podemos ver igualmente un comportamiento logaritmico, pero a diferencia del anterior primer grafico, podemos ver que el speed up con 16 procesadores tiende a aumentar en relacion a los valores de N siempre, aunque cabe destacar que cuando sobrepasa los 150, podemos ver que el grafico se va quedando estable pero muy rapido

En el caso de la eficiencia, podemos observar que igualmente tiene una forma como codo, o sea que va exponencialmente en caida, pero podemos ver que la caida mas importante sucede masomenos entre los 30 y los 50, en general es bastante similar al anterior

```
[18]: fig, ax = plt.subplots()
      ax.set(xlabel='Valores de N', ylabel='Speed Up',
            title='Speed Up con 16 procesadores')
      ax.plot(N, speedUpCaso2.iloc[:,1:])

      fig, ax = plt.subplots()
      ax.set(xlabel='Valores de N', ylabel='Eficiencia',
            title='Eficiencia con 16 procesadores')
      ax.plot(N, eficienciaCaso2.iloc[:,1:])
```

```
[18]: [<matplotlib.lines.Line2D at 0x7fe0dc4f97f0>]
```

