

Hoja de Trabajo No. 5

El programa contiguo se asemeja a un sistema operativo que simula su funcionamiento para analizar el rendimiento de dicho sistema según el tiempo de ejecución requerido para cierta cantidad de procesos (*i.e.*, las instrucciones). La aproximación para simular el sistema operativo se describe de la siguiente manera:

1. **Parámetros:** La configuración depende de aspectos como el intervalo de llegada para cada proceso, la memoria RAM, los CPU's disponibles (capacidad) y la velocidad de la CPU.
2. **Clases y Funciones:** En este caso se sugirió utilizar clases siguiendo el diseño de Programación Orientada a Objetos. Se tiene SistemaOperativo y Proceso. Cada clase tiene métodos que simulan la entrada de procesos, la ejecución de CPU y la gestión de la memoria RAM.
3. **Simulación:** El comportamiento del sistema operativo se simula a través de la ejecución de procesos. Con base en esto se registraron los tiempos para ejecutar los procesos y se estimaron los tiempo promedio y la desviación estándar de varias cantidades de procesos.

Resultados

Al correr la simulación se tienen los siguientes promedios para los intervalos de 5 con 25 procesos:

```
Cantidad de Procesos: 25, Intervalo: 10, Tiempo Medio: 2.4390111607180995, Desviación Estándar: 1.3559031680241715
Cantidad de Procesos: 25, Intervalo: 5, Tiempo Medio: 3.5907492808169827, Desviación Estándar: 2.7096580113394926
Cantidad de Procesos: 25, Intervalo: 1, Tiempo Medio: 19.692034122913473, Desviación Estándar: 10.16051966267159
```

Luego, con 50:

```
Cantidad de Procesos: 50, Intervalo: 10, Tiempo Medio: 2.654894886085513, Desviación Estándar: 1.6171457452899807
Cantidad de Procesos: 50, Intervalo: 5, Tiempo Medio: 3.330924513513096, Desviación Estándar: 2.378861484504392
Cantidad de Procesos: 50, Intervalo: 1, Tiempo Medio: 30.25769350274124, Desviación Estándar: 16.91991531201425
```

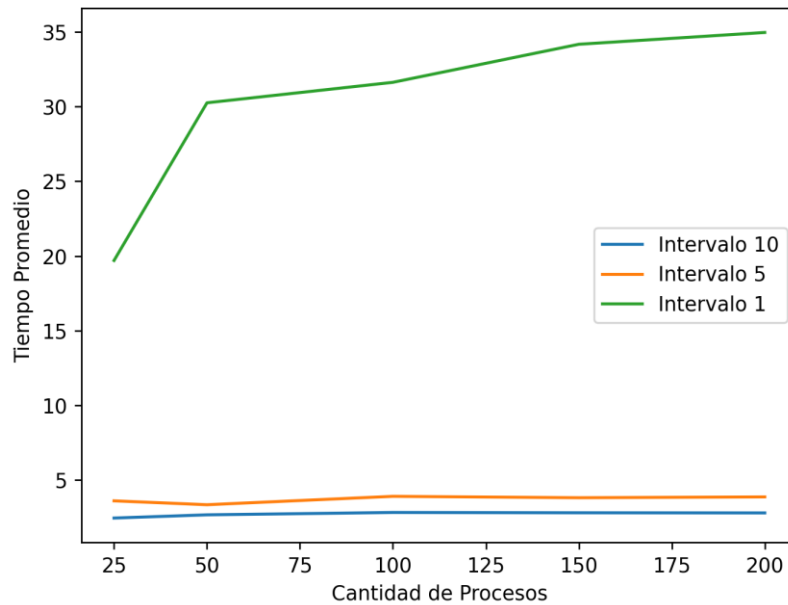
con 100:

```
Cantidad de Procesos: 100, Intervalo: 10, Tiempo Medio: 2.8125234432131925, Desviación Estándar: 1.8031082577587136
Cantidad de Procesos: 100, Intervalo: 5, Tiempo Medio: 3.8935209058756097, Desviación Estándar: 3.019813902455935
Cantidad de Procesos: 100, Intervalo: 1, Tiempo Medio: 31.62884675137062, Desviación Estándar: 16.44997958400235
```

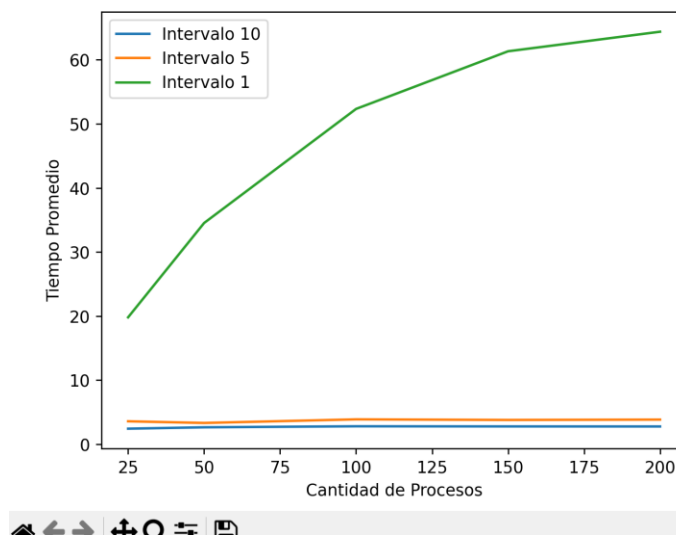
Finalmente, para 150 y 200 procesos de ejecución.

```
Cantidad de Procesos: 150, Intervalo: 10, Tiempo Medio: 2.7919277833475955, Desviación Estándar: 1.7175964122655571
Cantidad de Procesos: 150, Intervalo: 5, Tiempo Medio: 3.7997953837635046, Desviación Estándar: 2.7519036085787736
Cantidad de Procesos: 150, Intervalo: 1, Tiempo Medio: 34.17923116758041, Desviación Estándar: 17.189842161878257
Cantidad de Procesos: 200, Intervalo: 10, Tiempo Medio: 2.7826137742901706, Desviación Estándar: 1.5965811465702633
Cantidad de Procesos: 200, Intervalo: 5, Tiempo Medio: 3.855132608660723, Desviación Estándar: 2.6551216949406053
Cantidad de Procesos: 200, Intervalo: 1, Tiempo Medio: 34.96942337568531, Desviación Estándar: 16.450146392660603
```

Como forma de resumen se realizó un gráfico con el módulo de matplotlib que muestra el “comportamiento” del CPU.

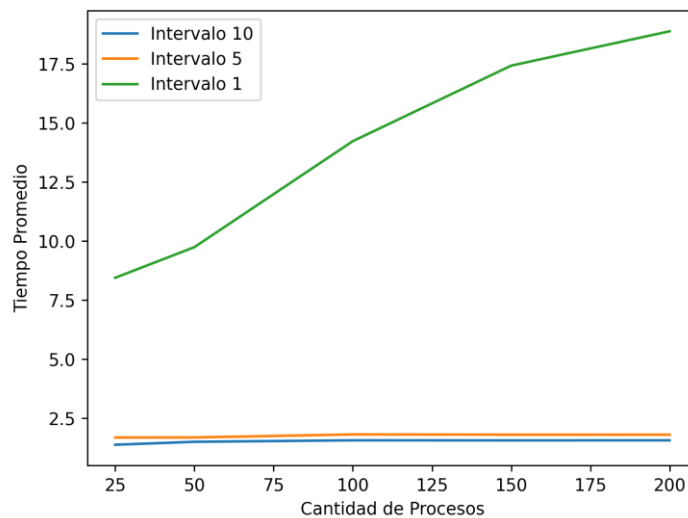


- i) Cuando se trata de reducir el tiempo promedio al incrementar la memoria a 200 se obtuvo el siguiente resultado:



Se obtuvieron resultados similares, pero como se evidencia en el gráfico, el tiempo requerido para más procesos aumentó considerablemente.

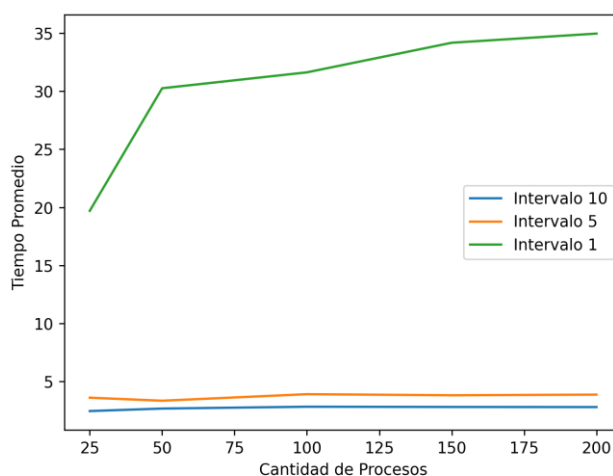
- ii) Por otro lado, al restaurar la memoria a 100, pero con un mejor procesador que efectua



6 instrucciones por unidad de tiempo se obtienen resultados considerablemente mejores. Es conspicuo que el tiempo promedio para cada caso se acortó significativamente, lo cual es un fenómeno lógico pues a mayor rapidez de procesamiento, más instrucciones se operarán en un mismo periodo a comparación de un procesador normal.

- iii) Ahora que se ha regresado a la velocidad normal del procesador y se emplean dos procesadores

No se tienen resultados sorprendentes para este caso, si no mas bien se trata de un rendimiento



normal si se compara con los parámetros iniciales.

En vista de lo anterior, se sugiere que para reducir el tiempo promedio de ejecución de esta simulación se aumente la velocidad del procesador o se agreguen muchas más unidades de procesamiento. Dado que se trata de un modelo se puede prescindir del coste o facilidad que tiene la implementación de las unidades de procesamiento. Es evidente que estos dos parámetros permiten aumentar la rapidez de ejecución de procesos y, por ende, reducir los tiempos de espera.