**Conclusiones:**

Se destaca de las pruebas realizadas que los algoritmos de complejidad O(n) registran tiempos de ejecución cercanos a 0 segundos. Y de estos los más rápidos son los que utilizan programación dinámica. Exceptuando a evaluarPow que debido a la llamada a la función Math.Pow registra un tiempo de ejecución levemente mayor a otros que utilizan cálculos simples.

evaluarHorner y obtenerCoeficienteK(Binomio de Newton) además de tener complejidad O(n) destacan en que no necesitan de memoria auxiliar a diferencia de otros métodos con Programación Dinámica.

Como contraparte los métodos con complejidad O(n2) registran los peores tiempos de ejecución siendo hasta 60 veces mayor en algunos casos. Los métodos de complejidad recursivos se encuentran entre los peores, debido a lo costoso de apilar en el stack. Pero el peor tiempo lo tiene obtenerCoeficientesYEvaluar(Binomio de Newton), esto es debido a que necesita a evaluarHorner para evaluar el polinomio y por lo tanto tiene que instanciar una clase Polinomio, hecho que consume mucho tiempo.