Memoria Práctica 1

# Introducción

# Código impreso

# Resultados

# Cuestiones

1. Justifica tu implementación de aleat num ¿en qué ideas se basa? ¿de qué libro/artículo, si alguno, has tomado la idea? Propón un método alternativo de generación de números aleatorios y justifica sus ventajas/desventajas respecto a tu elección.

El método elegido para crear números aleatorios ha sido el algoritmo de Park y Miller para crear números aleatorios, junto al shuffle/barajeo de Bays-Durham. Explicándolo rápidamente, este método se basa en tras un calentamiento, llenar una tabla con “semillas” que serán números aleatorios creados con el algoritmo de Park y Miller, y posteriormente se seleccionará el último número generado y se convertirá en un índice dentro del rango de la tabla, este índice será el elemento seleccionado de la tabla para ser convertido a un float en [0,1) que será devuelto como nuestro número aleatorio. Nótese que ese elemento será reemplazado en la tabla por otro generado a partir de este con el algoritmo de Park y Miller.

En estas imágenes se comentan línea por línea lo que sucede

2. Justifica lo más formalmente que puedas la corrección (o dicho de otra manera, por qué ordena bien) del algoritmo InsertSort.

3. ¿Por qué el bucle exterior de InsertSort no actúa sobre el último elemento de la tabla?

4. ¿Cu ́al es la operación básica de InsertSort?

5. Dar tiempos de ejecución en función del tamaño de entrada n para el caso peor WBS(n) y el caso mejor BBS(n) de InsertSort y BubbleSort. Utilizad la notación asintótica (O, Θ, o, Ω, etc) siempre que se pueda.

6. Compara los tiempos medios de reloj, así como el caso medio, peor y mejor obtenidos para InsertSort y BubbleSort, justifica las similitudes o diferencias entre ambos (es decir, indicad si las gráficas son iguales o distintas y por qué).