

Proyecto Integrador
Universidad Icesi

Taller 4: Diseño de Experimentos

Algoritmos de Ordenamiento

Juan Camilo Jiménez López
Laura Hincapié Calderón
Nicolas Taborda Hoyos
Alejandro Peña

2019-1

Diseño de experimentos

1. Planeación y realización

Alcance del experimento:

El experimento se llevará a cabo en tres computadores diferentes, dos de dos de los integrantes del grupo y el tercero, un computador de una de las salas de cómputo de la universidad. Los algoritmos de ordenamiento se utilizarán para ordenar arreglos de diferentes tamaños, especificados más adelante y que se encuentran en tres tipos de orden antes de ser ordenados.

Unidad Experimental:

Unidad Experimental: la unidad donde se van a ejecutar las pruebas son los tres diferentes equipos de procesadores iguales, pero generaciones diferentes.

Variable de respuesta:

El resultado de las pruebas lo veremos reflejado en el tiempo de ejecución de los dos algoritmos de ordenamiento escogidos. *Esperamos que la medición sea confiable.*

Factores:

Los factores controlables que se van a considerar en el experimento porque se cree que pueden influir sobre el resultado de este y a los cuales tenemos acceso a modificar sus valores, serán

1. Algoritmos de ordenamiento
2. Tamaño del arreglo a ordenar
3. Orden en el arreglo

4. Computador usado para el experimento

Los factores no controlables que se deben considerar son:

1. Nivel de fragmentación del disco duro del computador
2. Procesos que ejecuta el sistema operativo mientras se lleva a cabo el experimento
3. Errores aleatorios presentados por la máquina en la que se ejecute el proceso
4. Lentitud del Ide

Los factores estudiados son:

1. Tamaño del arreglo a ordenar
2. Orden en el arreglo
3. Computador usado para el experimento

Niveles de los factores:

Algoritmos de entrada

- Nivel 1: Comb Sort
- Nivel 2: Heap Sort
- Nivel 3: Merge Sort

Tamaño del arreglo a ordenar

- Nivel 1: 160000
- Nivel 2: 800000
- Nivel 3: 4000000
- Nivel 4: 20000000
- Nivel 5: 100000000

Tipo de dato

- Nivel 1: Int

- Nivel 2: Float
- Nivel 3: Double

Orden de los arreglos

- Nivel 1: Inverso
- Nivel 2: Pseudo-Aleatorio
- Nivel 3: Pseudo-Aleatorio Inverso

Computador usado

- Nivel 1: Portátil intel Core i7 8th gen (Laura)
- Nivel 2: Portátil intel Core i7 6th gen (Camilo)
- Nivel 3: Portátil intel Core i7 4th gen (Universidad)

Tratamientos: CAMILO

Por cuestiones de espacio, los tratamientos son estos para cada algoritmo. En cada corrida del programa, lo que se cambia es el tipo de dato y el tamaño del arreglo, porque en cada corrida se ejecutan los tres métodos para los diferentes tipos de órdenes.

Por esto, al abrirse el experimento en c# no dará el resultado de todos los tratamientos, si no de los que tengan el mismo nivel en el tipo de dato y el tamaño del arreglo.

El número total de tratamientos es $3(\text{niveles de computadores}) * 3(\text{niveles tipos de datos}) * 3(\text{niveles ordenes en el arreglo}) * 3(\text{niveles de algoritmos}) * 5(\text{tamaños en el arreglo}) = 405$ tratamientos. En la tabla se muestran 135 porque el último factor es el de los computadores, y estos se repetirían igual en cada computador.

No. Tratamiento	Computador	Tipo de dato	Orden del arreglo	Tamaño del arreglo
1	Laura	Int	Inverso	16000
2	Laura	Int	Inverso	800000
3	Laura	Int	Inverso	400000
4	Laura	Int	Inverso	20000000
5	Laura	Int	Inverso	100000000
6	Laura	Int	Pseudo-Aleatorio	16000

7	Laura	Int	Pseudo-Aleatorio	800000
8	Laura	Int	Pseudo-Aleatorio	400000
9	Laura	Int	Pseudo-Aleatorio	20000000
10	Laura	Int	Pseudo-Aleatorio	100000000
11	Laura	Int	Pseudo-Aleatorio-Inverso	16000
12	Laura	Int	Pseudo-Aleatorio-Inverso	800000
13	Laura	Int	Pseudo-Aleatorio-Inverso	400000
14	Laura	Int	Pseudo-Aleatorio-Inverso	20000000
15	Laura	Int	Pseudo-Aleatorio-Inverso	100000000
16	Laura	Double	Inverso	16000
17	Laura	Double	Inverso	800000
18	Laura	Double	Inverso	400000
19	Laura	Double	Inverso	20000000
20	Laura	Double	Inverso	100000000
21	Laura	Double	Pseudo-Aleatorio	16000
22	Laura	Double	Pseudo-Aleatorio	800000
23	Laura	Double	Pseudo-Aleatorio	400000
24	Laura	Double	Pseudo-Aleatorio	20000000
25	Laura	Double	Pseudo-Aleatorio	100000000
26	Laura	Double	Pseudo-Aleatorio-Inverso	16000
27	Laura	Double	Pseudo-Aleatorio-Inverso	800000
28	Laura	Double	Pseudo-Aleatorio-Inverso	400000
29	Laura	Double	Pseudo-Aleatorio-Inverso	20000000
30	Laura	Double	Pseudo-Aleatorio-Inverso	100000000
31	Laura	Float	Inverso	16000
32	Laura	Float	Inverso	800000
33	Laura	Float	Inverso	400000
34	Laura	Float	Inverso	20000000
35	Laura	Float	Inverso	100000000
36	Laura	Float	Pseudo-Aleatorio	16000
37	Laura	Float	Pseudo-Aleatorio	800000
38	Laura	Float	Pseudo-Aleatorio	400000
39	Laura	Float	Pseudo-Aleatorio	20000000
40	Laura	Float	Pseudo-Aleatorio	100000000

41	Laura	Float	Pseudo-Aleatorio-Inverso	16000
42	Laura	Float	Pseudo-Aleatorio-Inverso	800000
43	Laura	Float	Pseudo-Aleatorio-Inverso	400000
44	Laura	Float	Pseudo-Aleatorio-Inverso	20000000
45	Laura	Float	Pseudo-Aleatorio-Inverso	100000000
46	Camilo	Int	Inverso	16000
47	Camilo	Int	Inverso	800000
48	Camilo	Int	Inverso	400000
49	Camilo	Int	Inverso	20000000
50	Camilo	Int	Inverso	100000000
51	Camilo	Int	Pseudo-Aleatorio	16000
52	Camilo	Int	Pseudo-Aleatorio	800000
53	Camilo	Int	Pseudo-Aleatorio	400000
54	Camilo	Int	Pseudo-Aleatorio	20000000
55	Camilo	Int	Pseudo-Aleatorio	100000000
56	Camilo	Int	Pseudo-Aleatorio-Inverso	16000
57	Camilo	Int	Pseudo-Aleatorio-Inverso	800000
58	Camilo	Int	Pseudo-Aleatorio-Inverso	400000
59	Camilo	Int	Pseudo-Aleatorio-Inverso	20000000
60	Camilo	Int	Pseudo-Aleatorio-Inverso	100000000
61	Camilo	Double	Inverso	16000
62	Camilo	Double	Inverso	800000
63	Camilo	Double	Inverso	400000
64	Camilo	Double	Inverso	20000000
65	Camilo	Double	Inverso	100000000
66	Camilo	Double	Pseudo-Aleatorio	16000
67	Camilo	Double	Pseudo-Aleatorio	800000
68	Camilo	Double	Pseudo-Aleatorio	400000
69	Camilo	Double	Pseudo-Aleatorio	20000000
70	Camilo	Double	Pseudo-Aleatorio	100000000
71	Camilo	Double	Pseudo-Aleatorio-Inverso	16000
72	Camilo	Double	Pseudo-Aleatorio-Inverso	800000
73	Camilo	Double	Pseudo-Aleatorio-Inverso	400000
74	Camilo	Double	Pseudo-Aleatorio-Inverso	20000000

75	Camilo	Double	Pseudo-Aleatorio-Inverso	100000000
76	Camilo	Float	Inverso	16000
77	Camilo	Float	Inverso	800000
78	Camilo	Float	Inverso	400000
79	Camilo	Float	Inverso	20000000
80	Camilo	Float	Inverso	100000000
81	Camilo	Float	Pseudo-Aleatorio	16000
82	Camilo	Float	Pseudo-Aleatorio	800000
83	Camilo	Float	Pseudo-Aleatorio	400000
84	Camilo	Float	Pseudo-Aleatorio	20000000
85	Camilo	Float	Pseudo-Aleatorio	100000000
86	Camilo	Float	Pseudo-Aleatorio-Inverso	16000
87	Camilo	Float	Pseudo-Aleatorio-Inverso	800000
88	Camilo	Float	Pseudo-Aleatorio-Inverso	400000
89	Camilo	Float	Pseudo-Aleatorio-Inverso	20000000
90	Camilo	Float	Pseudo-Aleatorio-Inverso	100000000
91	LAB	Int	Inverso	16000
92	LAB	Int	Inverso	800000
93	LAB	Int	Inverso	400000
94	LAB	Int	Inverso	20000000
95	LAB	Int	Inverso	100000000
96	LAB	Int	Pseudo-Aleatorio	16000
97	LAB	Int	Pseudo-Aleatorio	800000
98	LAB	Int	Pseudo-Aleatorio	400000
99	LAB	Int	Pseudo-Aleatorio	20000000
100	LAB	Int	Pseudo-Aleatorio	100000000
101	LAB	Int	Pseudo-Aleatorio-Inverso	16000
102	LAB	Int	Pseudo-Aleatorio-Inverso	800000
103	LAB	Int	Pseudo-Aleatorio-Inverso	400000
104	LAB	Int	Pseudo-Aleatorio-Inverso	20000000
105	LAB	Int	Pseudo-Aleatorio-Inverso	100000000
106	LAB	Double	Inverso	16000
107	LAB	Double	Inverso	800000
108	LAB	Double	Inverso	400000

109	LAB	Double	Inverso	20000000
110	LAB	Double	Inverso	100000000
111	LAB	Double	Pseudo-Aleatorio	16000
112	LAB	Double	Pseudo-Aleatorio	800000
113	LAB	Double	Pseudo-Aleatorio	400000
114	LAB	Double	Pseudo-Aleatorio	20000000
115	LAB	Double	Pseudo-Aleatorio	100000000
116	LAB	Double	Pseudo-Aleatorio-Inverso	16000
117	LAB	Double	Pseudo-Aleatorio-Inverso	800000
118	LAB	Double	Pseudo-Aleatorio-Inverso	400000
119	LAB	Double	Pseudo-Aleatorio-Inverso	20000000
120	LAB	Double	Pseudo-Aleatorio-Inverso	100000000
121	LAB	Float	Inverso	16000
122	LAB	Float	Inverso	800000
123	LAB	Float	Inverso	400000
124	LAB	Float	Inverso	20000000
125	LAB	Float	Inverso	100000000
126	LAB	Float	Pseudo-Aleatorio	16000
127	LAB	Float	Pseudo-Aleatorio	800000
128	LAB	Float	Pseudo-Aleatorio	400000
129	LAB	Float	Pseudo-Aleatorio	20000000
130	LAB	Float	Pseudo-Aleatorio	100000000
131	LAB	Float	Pseudo-Aleatorio-Inverso	16000
132	LAB	Float	Pseudo-Aleatorio-Inverso	800000
133	LAB	Float	Pseudo-Aleatorio-Inverso	400000
134	LAB	Float	Pseudo-Aleatorio-Inverso	20000000
135	LAB	Float	Pseudo-Aleatorio-Inverso	100000000

Planeación y organización del experimento

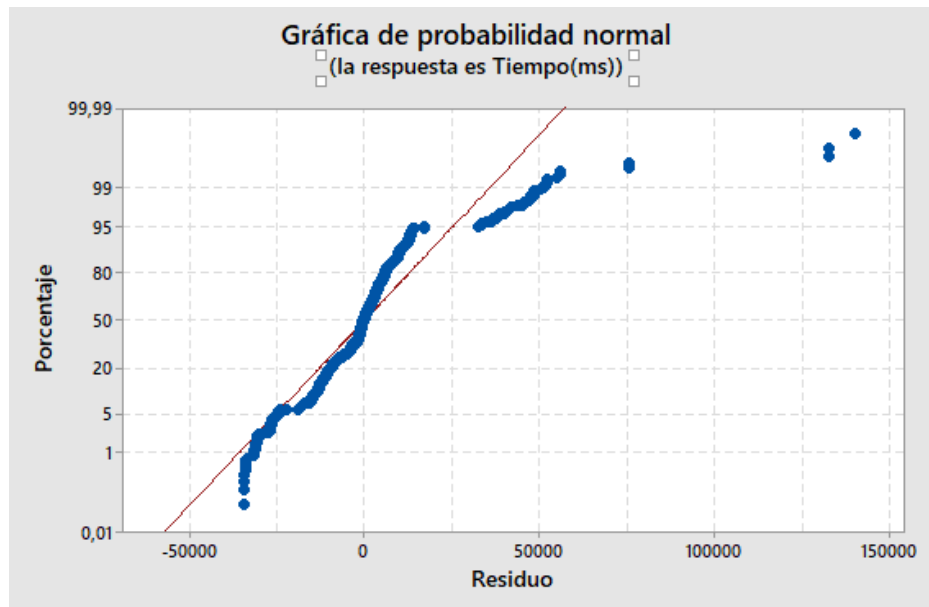
El experimento fue desarrollado por el grupo de 4 personas en donde cada uno se dividió las actividades a realizar, el programa que se desarrolló en visual studio 2017 hace la ejecución de los ordenamientos, cada vez que el programa se corre debe especificarse previamente el tamaño del arreglo, el tipo de dato, además deben de crearse las carpetas de los arreglos en donde se guardarán los datos sin arreglarse y la carpeta donde se guardarán los resultados, se debe crear en la carpeta de resultados dentro de esta se deben crear más carpetas separando por el tipo de dato como el int, float y el double, posterior a esto dentro de cada carpeta se crea otra con el tamaño del arreglo, dentro de este se generan los resultados de cada uno, se corrió tres veces cada carpeta de resultados por computador para tener una buena referencia en los datos y no depender de una sola toma de datos ya que el computador podría estar ejecutando distintas tareas.

2. Análisis

Con el fin de analizar los datos obtenidos de la mejor manera, se utilizó un diseño factorial de $3^4 \times 5$, por los niveles de cada uno de los factores utilizados (Computadores, tipos de datos, tipos de algoritmos, orden en los arreglos y tamaño del arreglo. Ya que, con este, se puede evaluar todas las combinaciones posibles de los niveles de los factores y así ver el comportamiento de la variable de interés en todas las posibles interacciones con dichos factores. Para el análisis de ellos, se hizo uso del software de análisis estadístico Minitab. El objetivo final de este análisis es determinar en que medida los factores estudiados afectan la variable de respuesta (tiempo de ejecución).

2.1. Modelo Lineal

Este modelo sirve para pronosticar el valor que toma una variable de interés que depende (en este caso), de ciertos factores. Este modelo también es útil para determinar si la media de distintos grupos de datos es igual. En esta gráfica, se muestra la prueba de normalidad con un intervalo de confianza del 95%:



Con esta gráfica se demuestra, que los residuos no siguen una distribución normal, pues se evidencia que la distribución de estos presenta unas colas largas que representa datos atípicos en los últimos valores.

2.2. Prueba de igualdad de varianzas

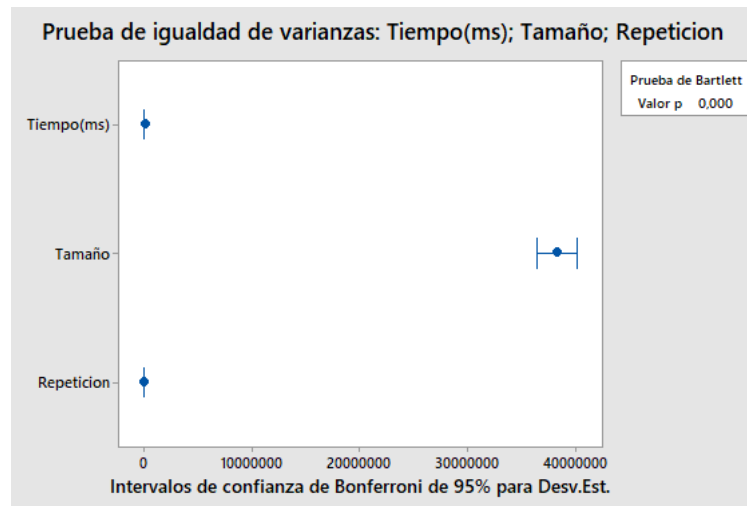
Con esta prueba se busca probar la igualdad en las varianzas, basándonos en la siguiente hipótesis.

H₀: Todas las varianzas son iguales.

H_a: Por lo menos una varianza es diferente.

Nivel de significancia: 0,05

Según los resultados obtenidos que se muestran a continuación, la hipótesis nula se rechaza, pues el valor p que se obtiene es menor a α , concluyendo que por lo menos una de las varianzas es diferente. Esto se ve en la diferencia de varianza del factor tamaño respecto a los factores tiempo y repetición, cuyo valor es mínimo en comparación con el del tamaño.



2.3. Pruebas ANOVA

Tras realizar las dos primeras pruebas, se continuó con la prueba ANOVA, para entrar a analizar las interacciones entre los factores. Para esto, se tomaron los 5 factores estudiados y la variable de respuesta (tiempo de ejecución) y sus resultados se muestran en la siguiente tabla.

Análisis de Varianza

Fuente	GL	SC Ajust.	MC Ajust.	Valor F	Valor p
PC	2	162538218	81269109	0,34	0,711
Tamaño	4	9,10192E+11	2,27548E+11	953,35	0,000
Tipo de dato	2	4993415277	2496707639	10,46	0,000
Metodo	2	48328479959	24164239980	101,24	0,000
Orden de arreglo	2	19144253967	9572126983	40,10	0,000
Error	1202	2,86897E+11	238683401		
Falta de ajuste	392	2,60093E+11	663502766	20,05	0,000
Error puro	810	26804363640	33091807		
Total	1214	1,26972E+12			

Con respecto a este análisis, se observa que el valor p para los factores de Tamaño del arreglo, Tipo de dato, Método (o Algoritmo) y Orden del arreglo, en valor p es menor al nivel de significancia (0,05), significando que estos factores son significativos para la variable de respuesta.

Por esto mismo, no se acepta ni se rechaza ninguna hipótesis, por lo que se “asume” que el tiempo de ejecución para los tres algoritmos varía con respecto a los diferentes factores. Debido a esto, a continuación, se implementa la prueba F de Fisher.

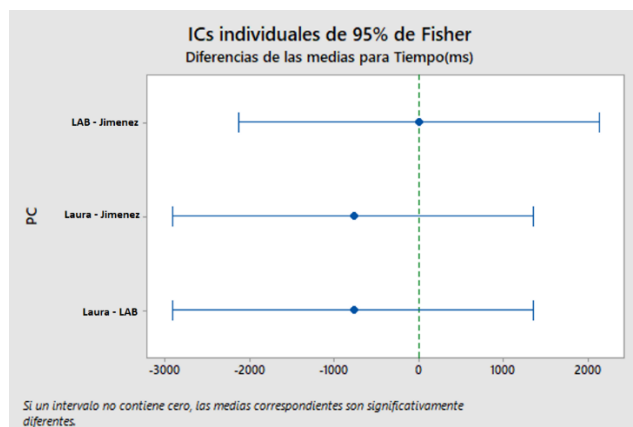
2.4. Prueba F de Fisher

Comparación por parejas de Fisher: PC

Agrupar información utilizando el método LSD de Fisher y una confianza de 95%

PC	N	Media	Agrupación
Jimenez	405	17232,4	A
LAB	405	17232,2	A
Laura	405	16456,4	A

Las medias que no comparten una letra son significativamente diferentes.



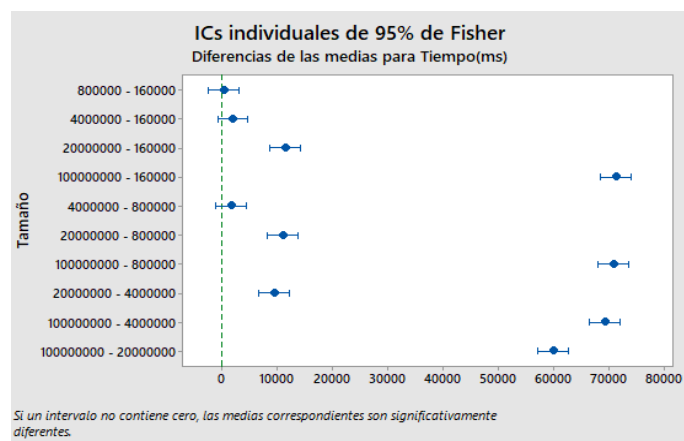
En la primera tabla, se muestra que los tres niveles de este factor pertenecen al mismo grupo A, lo que señala que sus medias no varían de manera significativa y una de las razones de esto podría ser que a pesar de que los tres computadores sean de diferente generación y con distinta RAM, el procesador es el mismo.

Comparación por parejas: Tamaño

Agrupar información utilizando el método LSD de Fisher y una confianza de 95%

Tamaño	N	Media	Agrupación
100000000	243	71082,5	A
200000000	243	11372,2	B
400000000	243	1999,8	C
800000000	243	350,2	C
1600000000	243	63,5	C

Las medias que no comparten una letra son significativamente diferentes.



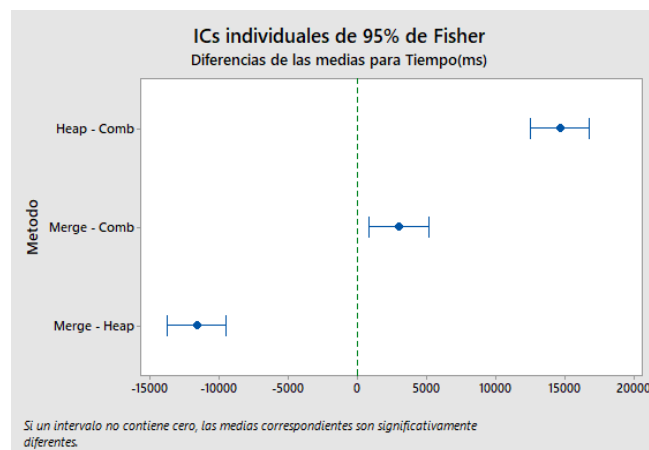
En esta prueba se evidencia que el tamaño de los arreglos más pequeños, se agrupan en el mismo grupo, siendo sus medias similares. Mientras que los últimos dos valores que se alejan bastante de estos primeros se agrupan cada uno en diferentes grupos, pues arrojan tiempos mucho mayores.

Comparación por parejas de Fisher: Algoritmos

Agrupar información utilizando el método LSD de Fisher y una confianza de 95%

Metodo	N	Media	Agrupación
Heap	405	25723,8	A
Merge	405	14095,5	B
Comb	405	11101,6	C

Las medias que no comparten una letra son significativamente diferentes.



Gráfica 5. Prueba F de Fisher por pareja para los tipos de algoritmos.

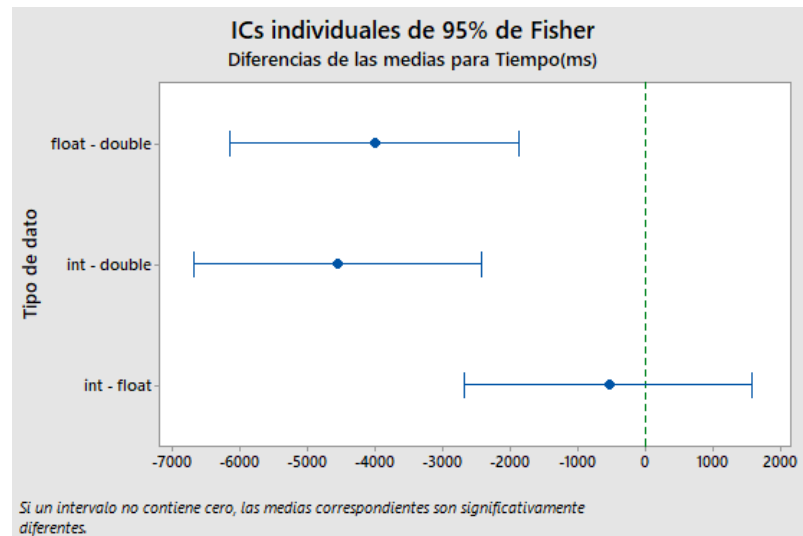
Esta prueba nos muestra la diferencia de los tiempos de cada algoritmo, significando que sus medias varían con respecto a la de cada uno y esto probablemente se debe a que cada algoritmo a pesar de tener la misma complejidad temporal acomoda los datos de diferente manera en su método sort.

Comparación por parejas: Tipo de dato

Agrupar información utilizando el método LSD de Fisher y una confianza de 95%

Tipo de dato	N	Media	Agrupación
double	405	19823,6	A
float	405	15819,3	B
int	405	15278,1	B

Las medias que no comparten una letra son significativamente diferentes.



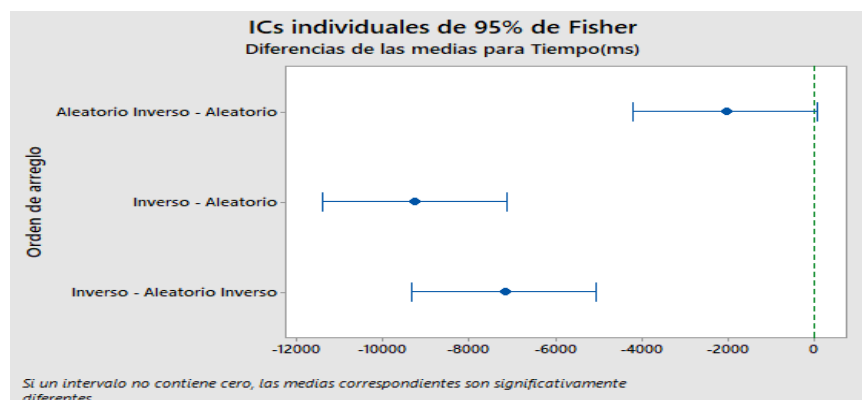
Para este caso del tipo de dato, los tipos int y float quedaron en el mismo grupo, y sus medias fueron muy cercanas, lo que no sucedió con el double, el cual arrojaba mayores tiempos de ejecución ya que ocupa el doble de espacio y por eso su media fue mucho mayor, lo que ocasiona que en la gráfica los intervalos de confianza de las parejas que lo contienen se alejan del cero.

Comparación por parejas: Orden del arreglo

Agrupar información utilizando el método LSD de Fisher y una confianza de 95%

Orden de arreglo	N	Media	Agrupación
Aleatorio	405	20750,0	A
Aleatorio Inverso	405	18682,6	A
Inverso	405	11488,3	B

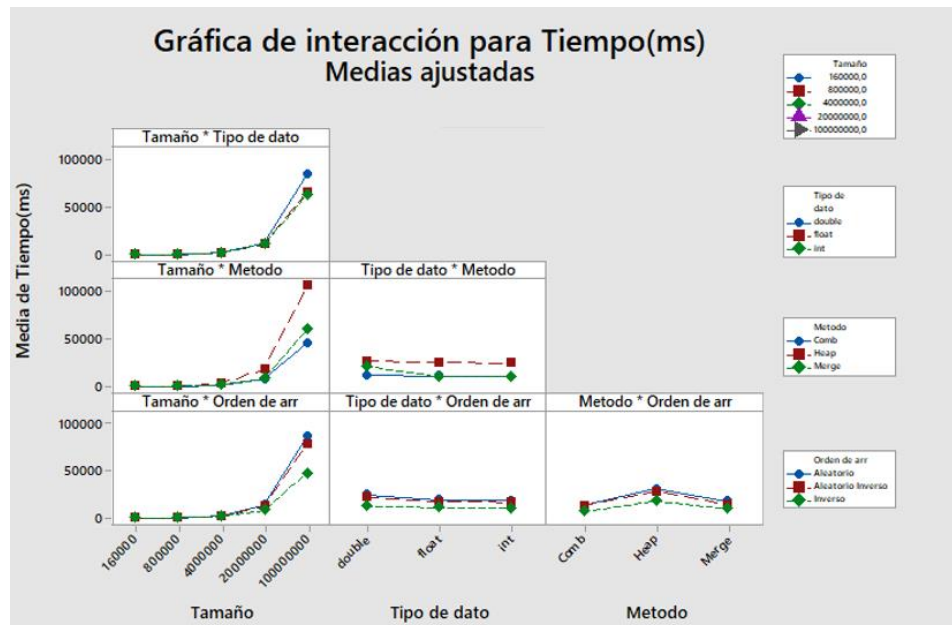
Las medias que no comparten una letra son significativamente diferentes.



Para el orden de los arreglos, el orden inverso se ubica en un grupo diferente a los otros dos, ya que posiblemente a la hora de ordenar en los valores aleatorios, muchos pudieron no haber quedado consecutivos, o muy distantes entre valores por lo que los algoritmos tendrían que

hacer mayores iteraciones para organizarlos, pero en el caso de orden inverso tenían que hacer menor cantidad de accesos.

Gráficas de los factores relevantes



3. Interpretación

Como se planteó en el análisis de cada gráfica, los factores mas influyentes en el tiempo de ejecución son:

- El tamaño del arreglo
- El orden inicial del arreglo
- El algoritmo ejecutado
- El tipo de datos

Esto se ve reflejado en el valor p de la prueba de ANOVA, donde el valor p de la fuente PC, es el único mayor al nivel de significancia. Por otro lado, en cada prueba que se realizó, se determinaron los factores que más afectaban los resultados. En el modelo lineal, se obtuvo que los datos no seguían una distribución normal, ya que después de haberse realizado la grafica con los datos se evidencian que hay datos muy atípicos a la línea de tendencia normal. En la evaluación de la igualdad de las varianzas en los tres aspectos de tamaño, tiempo y repetición, se puede ver la varianza del factor tamaño es diferente a la varianza del factor tiempo y repetición, significando que este factor es significativo para la variable respuesta.

Después de realizar la prueba ANOVA se concluyó que los diferentes factores afectan el tiempo de ejecución de los tres algoritmos.

4. Conclusiones

En conclusión, después de realizar las diferentes pruebas de inferencia a los datos arrojados durante el experimento, se llegó a concluir que los tiempos de ejecución de los algoritmos se veían afectados por los diferentes factores evaluados durante el experimento. En cada prueba de inferencia se evidenció como cada factor se va evaluando y de esta manera comprobando lo que se vio en la experimentación. A partir de la gráfica de los residuos se realizó una gráfica de normalidad y se concluyó que estos residuos no seguían una distribución normal ya que había datos muy atípicos, además en la prueba de diferencia de varianza se quería comprobar que había al menos una varianza diferente de las tres evaluadas y se pudo concluir que la varianza del factor tamaño era diferente y que las demás evaluadas tenían un valor de comparación mínimo con respecto a la de tamaño. También se concluyó con la prueba de ANOVA que los factores evaluados afectaban significativamente el tiempo de ejecución ya que, con un valor p del 5 %, en la prueba el valor p de los factores fue menor que 5% y no se rechazó ninguna de la hipótesis planteada al inicio de la ANOVA.

En cada prueba de F Fisher se compararon parejas y se finalizó en cada comparación que los computadores no alteraban mucho el tiempo de respuesta, ya que en todos los computadores los componentes eran diferentes menos el procesador, en otra instancia se llegó a la conclusión de que los arreglos de tamaño similar estaban en la misma agrupación ya que el tamaño de el arreglo es el que determina el tiempo que se demora el algoritmo realizando la tarea asignada.

Cuando se comparó y se realizó la prueba Fisher por tipo de dato, el tipo de dato double quedó en una agrupación diferente al tipo int y float, esto se puede argumentar porque el tipo de dato double ocupa un espacio de memoria mayor a los demás.