

Práctica N^o 3

Teoría de colas

Nombre: José Adrián García Fuentes
Fecha: 02/Marzo/2021

Profesor: Satu Elisa Schaeffer

1. Introducción

La teoría de colas es un área de las matemáticas que estudia el comportamiento de líneas de espera. Los trabajos que están esperando ejecución en un cluster esencialmente forman una línea de espera [1].

2. Objetivo

- Examinar cómo las diferencias en los tiempos de ejecución de los diferentes ordenamientos cambian cuando se varía el número de núcleos asignados al cluster, utilizando como datos de entrada un vector que contiene primos grandes, descargados de <https://primes.utm.edu/lists/small/millions/> y no-primos (creados a partir de ellos) con por lo menos nueve dígitos, aplicando pruebas estadísticas adecuadas y visualización científica clara e informativa [1].

3. Metodología

La metodología empleada se realizó a través de RStudio[2] llevando a cabo los pasos señalados en la *Práctica 3: teoría de colas* [1], se examinaron las diferencias en tiempos de ejecución de los diferentes ordenamientos cuando se varia el numero de nucleos asignados al cluster, utilizando como datos de entrada del vector utilizando primos grandes, el código completo de la metodología empleada se encuentra en el repositorio [3] del autor.

4. Resultados

Se obtuvo el código secuencia para determinar si un número es primo o no primo, a partir de este experimento se determino el tiempo que tardaba en detectar si era primo o no primo. A continuación se muestra parte del código del experimento [3] en el cual se señala el número de repeticiones y parte de la función dada solicitando números de manera pseudoaleatoria que determinaran la posición final de nuestro punto, dicho código fue modificado de el código del docente [4].

```
1 library(parallel)
2 detectCores(-1)
3 primo <- function(n) {
4   if (n < 4) {
5     return(TRUE)
6   }
7   if (n %% 2 == 0) {
8     return(FALSE)
```

Cuadro 1: Registro de tiempos

min	max	promedio	media
0.20	0.72	0.50	0.48
0.1	0.21	0.19	0.16
0.03	0.20	0.10	0.12

```

9  }
10 for (i in seq(3, max(3, ceiling(sqrt(n))), 2)) {
11     if (n %% i == 0) {
12         return(FALSE)
13     }
14 }
15 return(TRUE)
16 for (noprimo in (valoresprimos[1] + 1):(valoresprimos[2]-1))
17 {print(noprimo)}
18 }

```

A continuación en la Fig 1 se muestra un diagrama caja bigote de el tiempo de interacción al correr el experimento.

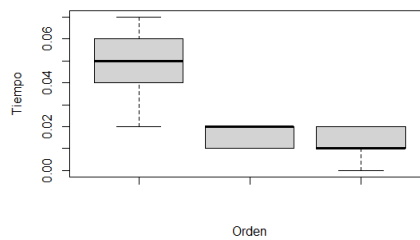


Figura 1:
Diagrama caja bigote

5. Reto 1

Modificar la tarea por realizar a que se encuentren todos los divisores del número (es decir, todos los enteros mayores a uno y menores al número mismo que lo dividen exactamente y examinar si las conclusiones cambian.

```

1 divisor<- function (n) {
2     div<-numeric()
3     for (i in 1:ceiling(sqrt(n))) {
4         if ((n %% i) == 0) {
5             div<-c(div,i)
6             div<-c(div,n/i)
7         }
8     }
9     return(sort(unique(div)))
10 }

```

6. Reto 2

Modificar el primer reto a que encuentre solamente los factores y sus multiplicidades, es decir, que encuentre para n aquellos números primos $1 < p < n$ y sus potencias para que el producto de los factores con esas potencias de n . Nuevamente hay que examinar si este cambio afectó las conclusiones del experimento de la tarea.

```
1 fact_primos<-function(n){
2   div<-numeric()
3   if(n==1){
4     return(1)
5   }
6   while(n%%2 == 0){
7     div<-c(div,2)
8     n = n/2
9   }
10  for(i in seq(3, max(3, ceiling(sqrt(n))), 2)){
11    while(n%i==0){
12      div<-c(div,i)
13      n=n/i
14    }
15  }
16  if(n>2){
17    div<-c(div,n)
18  }
19  return(table(div))
20 }
21
```

7. Conclusión

En conclusión el tiempo en que tarda un experimento en correr una secuencia variando el numero de nucleos de un cpu puede ser mas corto de pendiendo de que tan largo sea el experimento

Referencias

- [1] E. Schaeffer, “Práctica 3: teoría de colas,” Marzo 2021. <https://elisa.dyndns-web.com/teaching/comp/par/p3.html>.
- [2] J. J. Allaire, “Rstudio,” Marzo 2021. <https://rstudio.com>.
- [3] J. A. Garcia Marzo 2021. <https://github.com/fuentesadrian/SIMULACION-DE-NANOMATERIALES/tree/main/Tarea%202>.
- [4] E. Schaeffer, “Práctica 3: teoría de colas,” Marzo 2021. <https://github.com/fuentesadrian/Simulation/blob/master/QueuingTheory/ordering.R>.