

TAREA I: Análisis de Algoritmos

Prof. Andrea Rodríguez, Ayudante: Erick Elejalde

GRUPO 3

Los grupos deberán resolver 2 problemas que han sido asignados en forma aleatoria. Para los problemas que deban escribir e implementar un algoritmo, especificar claramente lo que signifiquen las variables del algoritmo y documentar. La implementación debe ser sometida a una evaluación experimental donde el tamaño del problema sea creciente y donde la entrada sea creada en forma aleatoria. Al definir un algoritmo se pide no solo la complejidad sino demostrar que es correcto. Se debe entregar un informe y el código fuente que sea compilable en linux (C o C++) . El trabajo se debe entregar el día viernes 7 de abril a las 12:00 en forma digital por correo a la profesora y las presentaciones de los trabajos se harán a partir de la semana del 18 de abril (luego del primer control).

Los problemas asignados a este grupo son los siguientes:

1. Considere el siguiente algoritmo donde la entrada es una matriz de $n \times n + 1$ números reales.

ENIGMA(var $A[0 \dots n - 1, 0 \dots m]$)

```
1  for  $i = 0$  to  $n - 1$  do
2    for  $j = i + 1$  to  $n - 1$  do
3      for  $k = i$  to  $n$  do
4         $A[j, k] \leftarrow A[j, k] - A[i, k] * A[j, i] / A[i, i]$ 
```

- Indique que hace el algoritmo y demuestre que hace lo que Usted dice.
 - Derive la complejidad del algoritmo.
 - ¿Existe alguna mejora posible al algoritmo? Implemente y haga una comparación experimental de ambos algoritmos.
2. Suponga que se desea mantener un elemento en memoria con probabilidad uniforme sobre todos los elementos que vayan llegando de uno en uno. Esto se quiere hacer sin saber el número de elementos que llegará por adelantado. Se le pide escribir un algoritmo que realice esto,

probando que la probabilidad de que un elemento cualquier que haya pasado pueda quedar en memoria sea efectivamente uniforme. Indicar su complejidad.