

# Práctica Eficiencia

## Participantes

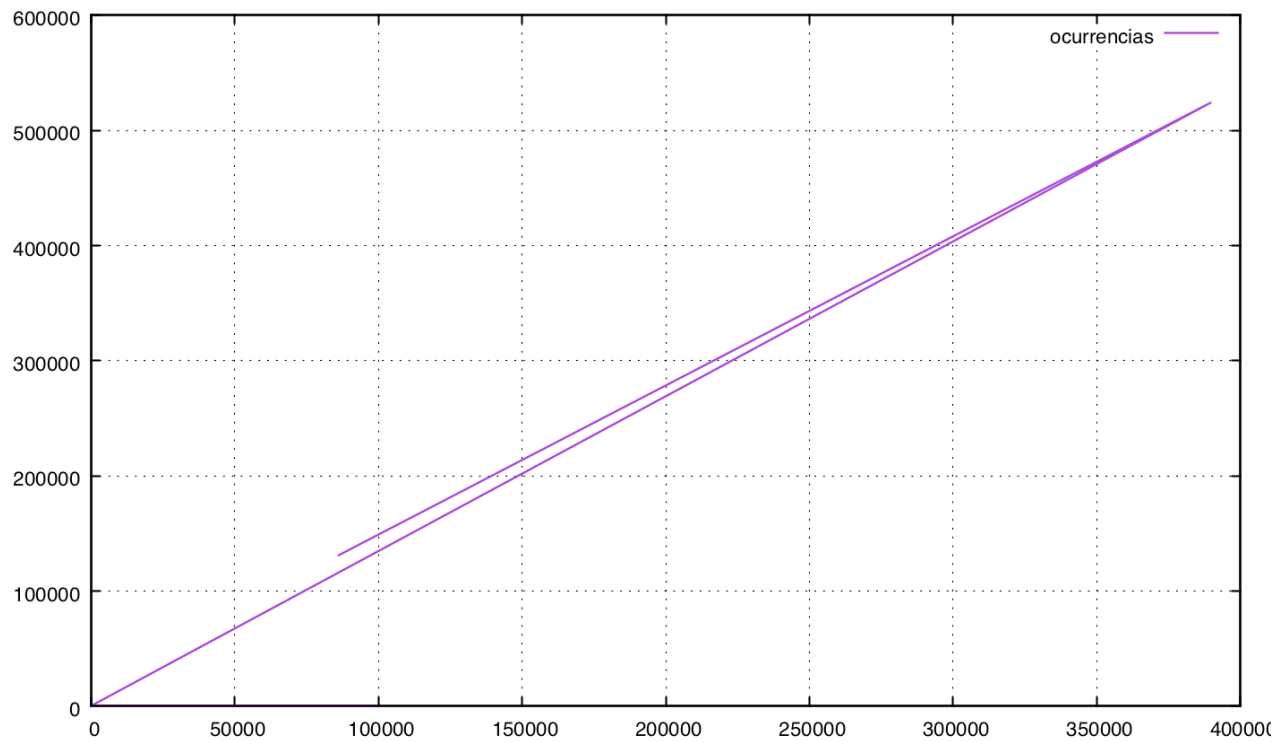
Jesús Jiménez Sánchez - 76656119V  
María Nazaret Ruiz Jaldo - 76069342Q  
Jaime Frias Funes - 50625384S

## 1. Ocurrencias

El archivo ocurrencias.dat contiene los siguientes datos

10	1.06e-06
1010	2.8793e-05
2010	5.4247e-05
3010	7.9813e-05
4010	0.000105636
5010	0.000132262
6010	0.000158158
7010	0.000184871
8010	0.000212441
9010	0.000200605
10010	0.00019807
11010	0.000217115
12010	0.000236121
13010	0.000256174
14010	0.000275425
15010	0.000295679
...	
90010	0.00228712
91010	0.00223169
92010	0.00228866
93010	0.00221506
94010	0.00237049
95010	0.00240051
96010	0.00244403
97010	0.00247374
98010	0.00252771
99010	0.00249423

La gráfica que genera es esta:



Y la regresión será la siguiente:

$$f(x) = a \cdot x^2 + b \cdot x + c$$

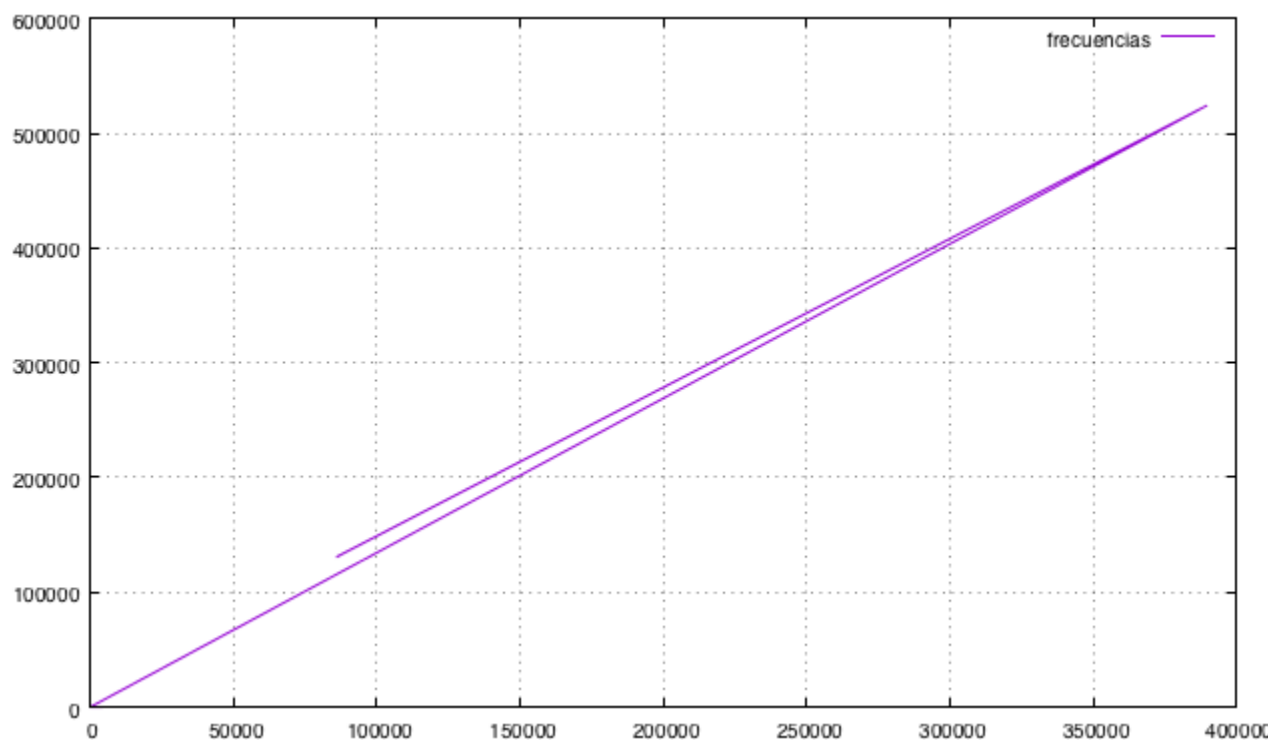
$$f(x) = 4.41659e-06 \cdot x^2 - 0.401867 \cdot x + 6043.03$$

## 2. Frecuencias

En el archivo frecuenciasF.dat aparece lo siguiente:

10	3.1832e-05
110	0.000569125
210	0.00196911
310	0.00431948
410	0.00579837
510	0.00827145
...	
2500	0.00353221
2600	0.0030029
2700	0.00363125
2800	0.00317023
2900	0.00409952
3000	0.0039395
3100	0.00405172
3200	0.00411942
3300	0.00539271
3400	0.0072112
3500	0.00790045
3600	0.0121086
3700	0.0105186
3800	0.0117592
3900	0.0100488

La gráfica que genera es esta, que es muy parecida a la anterior:



Y la regresión será la siguiente:

$$f(x) = a \cdot x^2 + b \cdot x + c$$

$$f(x) = -4.77234e-07 \cdot x^2 + 1.53916 \cdot x - 2973.1$$

### 3. Algoritmos ordenación

#### Ordenación por inserción

```
void Ordenacion_Insercion ( vector<string> & v){
    string tmp;
    int i, j;
    for (i=1; i<v.size();i++){
        tmp=v[i];
        for(j=i;(j>0)&&( tmp<v[j-1] );j--){
            v[j]=v[j-1];
        }
        v[j]=tmp;
    }
}
```

Considerado el peor caso, el bucle interno se ejecuta una vez en la primera iteración, dos veces en la segunda, y así sucesivamente hasta  $n - 1$ :

$$\sum_{i=1}^{n-1} i = \frac{n(n-1)}{2} \quad \text{iteraciones en el bucle interno, por tanto:}$$

$$\sum_{i=1}^n i = \frac{n(n+1)}{2}$$
$$T(n) = \frac{n(n-1)}{2} c_1 + (n-1)c_2 + c_3$$

$$T(n) = O(n^2)$$

#### Ordenación por selección

```
void Ordenacion_Selección(string v[] ){
    string temp;
    int i,j,pos_min;
    for(i=0;i<v.size();i++){
        pos_min=i;
        for(j=i+1;j<v.size();j++){
            if(v[j]<v[pos_min])
                pos_min=j;
        }
        //Coloca el mínimo en v[i]
        tmp=v[i];
        v[i]=v[pos_min];
        v[pos_min]=tmp;
    }
}
```

$$\sum_{i=1}^{n-1} (n - i) = \sum_{i=1}^{n-1} n - \sum_{i=1}^{n-1} i = (n - 1)n - \frac{n(n-1)}{2}$$

$$T(n) = O(n^2)$$

Nos ha salido un orden de eficiencia **cuadrático**.

## Ordenación por burbuja

```
void burbuja(vector<string> & T, int inicial, int final) {
    int i, j;
    string aux;
    for (i = inicial; i < final - 1; i++)
        for (j = final - 1; j > i; j--)
            if (T[j] < T[j-1]) {
                aux = T[j];
                T[j] = T[j-1];
                T[j-1] = aux;
            }
}
```

$$\sum_{i=1}^{n-2} a(n - i - 1)$$

$$T(n) = O(n^2)$$