



PRÁCTICA II

Divide y Vencerás

Por: Antonio Gámiz

Objetivo:

Encontrar índice i tal que $v[i] == i$.

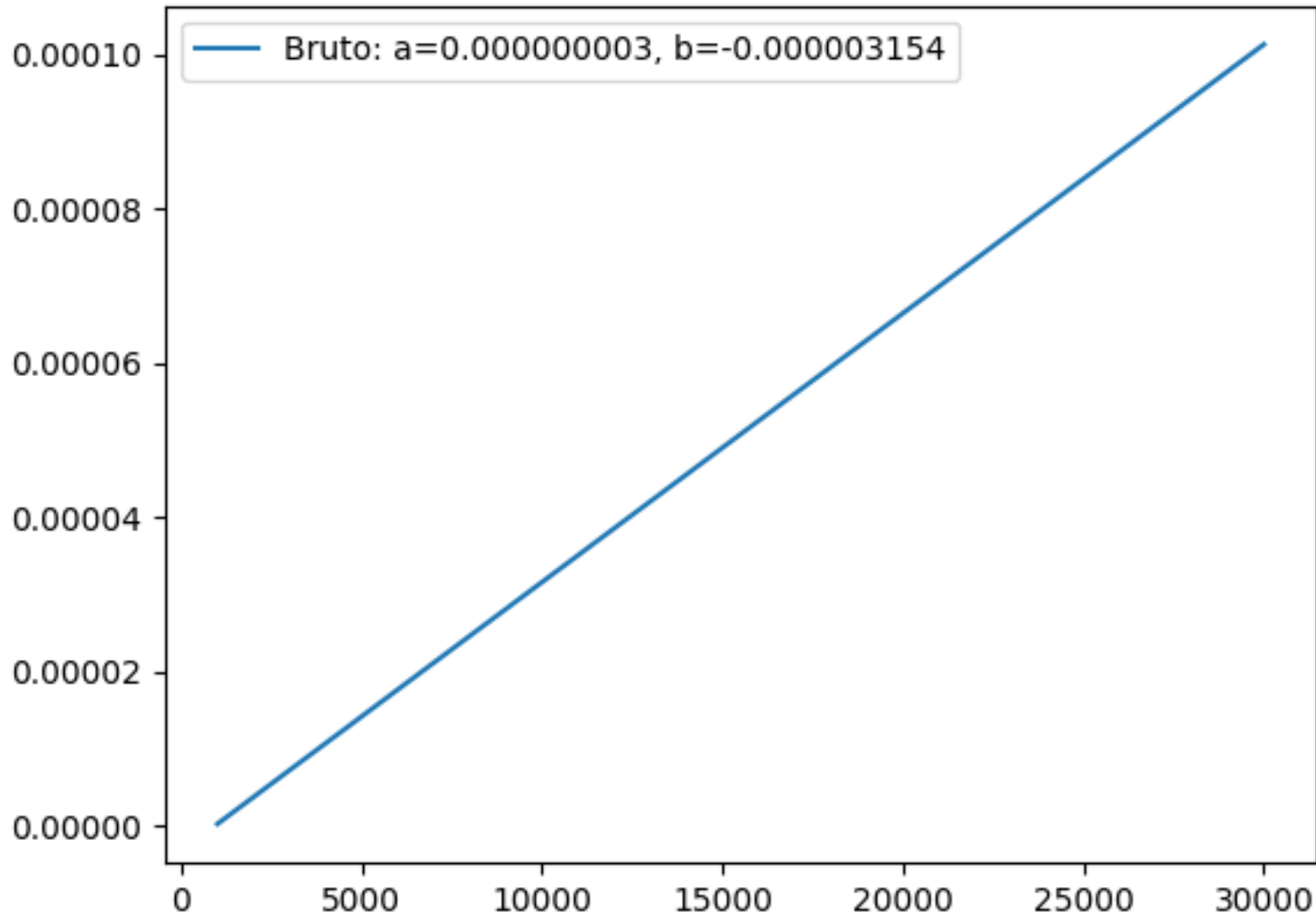
Métodos empleados:

- Fuerza bruta.
- Divide y vencerás.

Fuerza Bruta

```
1  int algoritmo_bruto(vector<int> v)
2  {
3      for(int i=0; i<v.size(); i++){
4          if(v[i]==i) return i;
5      }
6      return -1;
7  }
```

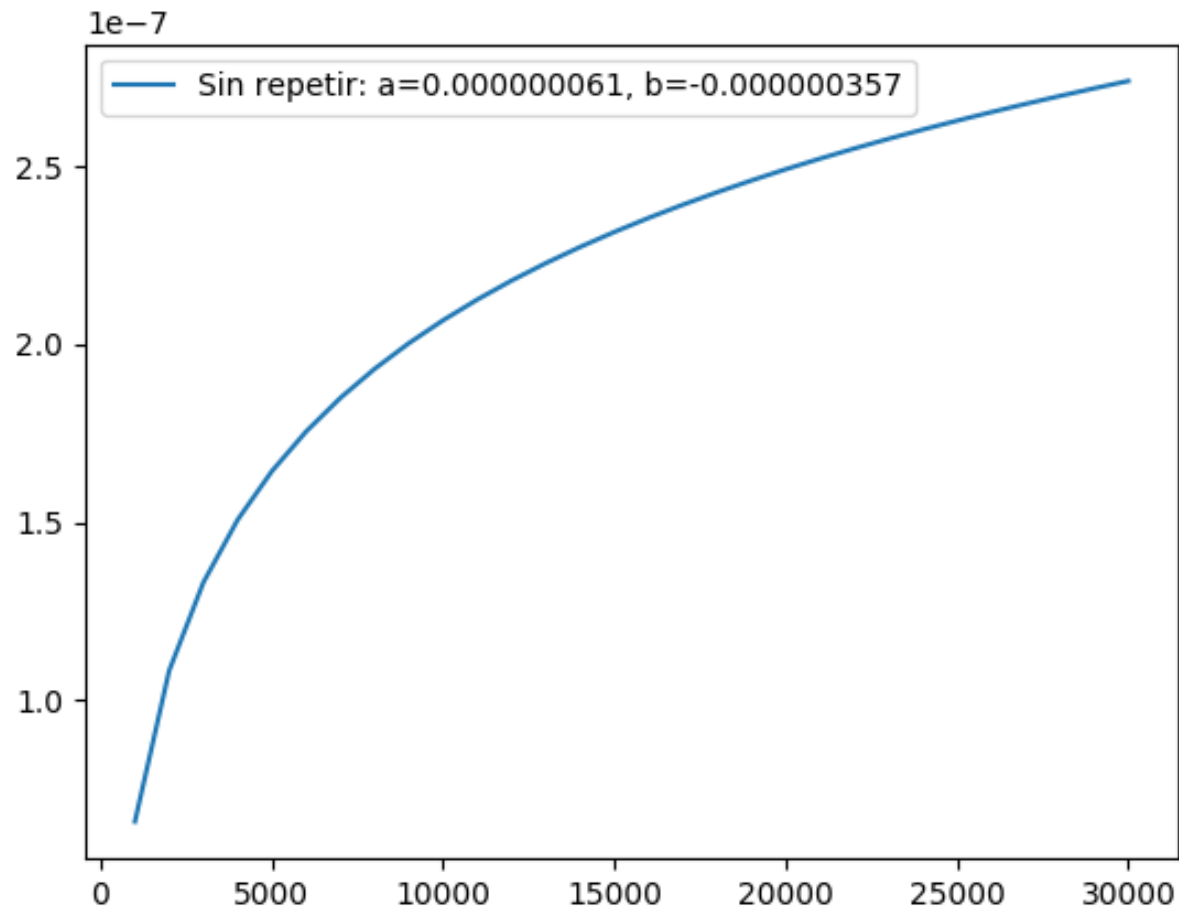
Eficiencia Fuerza bruta

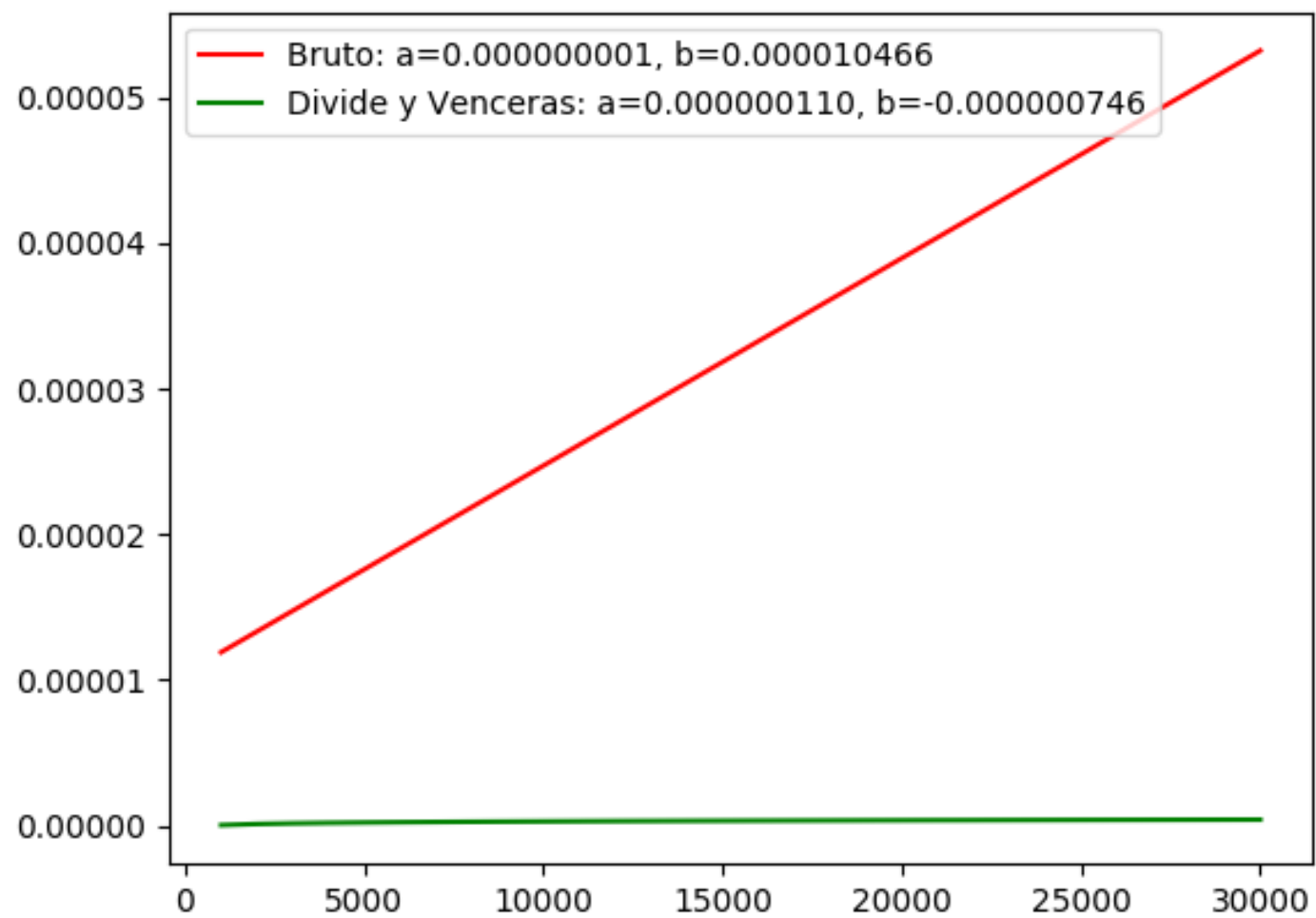


Divide y vencerás (sin repetición)

```
1 int divide_nr(vector<int>& v, int n1, int n2)
2 {
3     if( n1==n2 )
4         if( v[n1]==n1 ) return n1;
5     else return -1;
6     else {
7         int middle=(n1+n2)/2;
8         if( v[middle]==middle ) return middle;
9         else if ( v[middle] > middle ) return divide_nr(v, n1, middle-1);
10        else return divide_nr(v, middle+1, n2);
11    }
12 }
```

Eficiencia Divide y vencerás





Divide y Vencerás (repetición)

```
1 int divide_r(vector<int>& v, int n1, int n2){
2     if( n1==n2 ){
3         if( v[n1]==n1 )
4             return n1;
5         else
6             return -1;
7     }else{
8         int middle=(n1+n2)/2;
9         if( v[middle]==middle )
10            return middle;
11        else if( v[middle] > middle ){
12
13            int left=divide_r(v, n1, middle-1);
14            int right;
15
16            if( v[middle] <= n2 )
17                right=divide_r(v, v[middle], n2);
18            else right=0;
19
20            if( left!=0 ) return left;
21            if( right!=0 ) return right;
22
23            return -1;
24        }else{
25            int right=divide_r(v, middle+1, n2);
26            int left;
27
28            if( n1<=v[middle] ) left=divide_r(v, n1, v[middle]);
29            else left=0;
30
31            if( right!=0 ) return right;
32            if( left!=0 ) return left;
33
34            return -1;
35        }
36    }
37 }
```




FIN