

Práctica 1

Análisis de eficiencia de algoritmos

Maria Jesús López Salmerón

Nazaret Román Guerrero

Laura Hernández Muñoz

José Baena Cobos

Carlos Sánchez Páez

14 de marzo de 2018

Índice

Índice

Script individual

```
#!/bin/bash
if [ $# -eq 3 ]
then
    i="0"
    output="out"
    tam=$2
    #Primer argumento: programa a ejecutar
    #Segundo argumento: tamaño inicial
    #Tercer argumento : incremento
    while [ $i -lt 25 ]
    do
        ./$1 $tam >> $1.out
        i=$((i+1))
        tam=$((tam+$3))
    done
else
    echo "Error de argumentos"
fi
```

Script conjunto

```
#!/bin/bash
echo "Ejecutando burbuja..."
./individual.sh burbuja 1000 1000
echo "Ejecutando insercion..."
./individual.sh insercion 1000 1000
echo "Ejecutando seleccion..."
./individual.sh seleccion 1000 1000
echo "Ejecutando mergesort..."
./individual.sh mergesort 1000000 500000
echo "Ejecutando quicksort..."
./individual.sh quicksort 1000000 500000
echo "Ejecutando heapsort..."
./individual.sh heapsort 1000000 500000
echo "Ejecutando hanoi..."
./individual.sh hanoi 10 1
echo "Ejecutando floyd..."
./individual.sh floyd 100 100
```

Makefile

```
DOC=doc
SRC=src
OUT=out
BIN=src

all : todos
todos : burbuja floyd hanoi heapsort insercion mergesort quicksort seleccion
        cd $(SRC) ; ./todos.sh
burbuja :
        g++ -o ./$(BIN)/burbuja ./$(SRC)/burbuja.cpp
floyd :
        g++ -o ./$(BIN)/floyd ./$(SRC)/floyd.cpp
hanoi :
        g++ -o ./$(BIN)/hanoi ./$(SRC)/hanoi.cpp
heapsort :
        g++ -o ./$(BIN)/heapsort ./$(SRC)/heapsort.cpp
insercion :
        g++ -o ./$(BIN)/insercion ./$(SRC)/insercion.cpp
mergesort :
        g++ -o ./$(BIN)/mergesort ./$(SRC)/mergesort.cpp
quicksort :
        g++ -o ./$(BIN)/quicksort ./$(SRC)/quicksort.cpp
seleccion :
        g++ -o ./$(BIN)/seleccion ./$(SRC)/seleccion.cpp
```

Índice

Modificación de código fuente

```
clock_t tantes;  
clock_t tdespues;  
tantes = clock();  
algoritmo_en_cuestion(T, n);  
tdespues = clock();  
cout << ((double)(tdespues - tantes))  
/ CLOCKS_PER_SEC << endl;
```


Índice

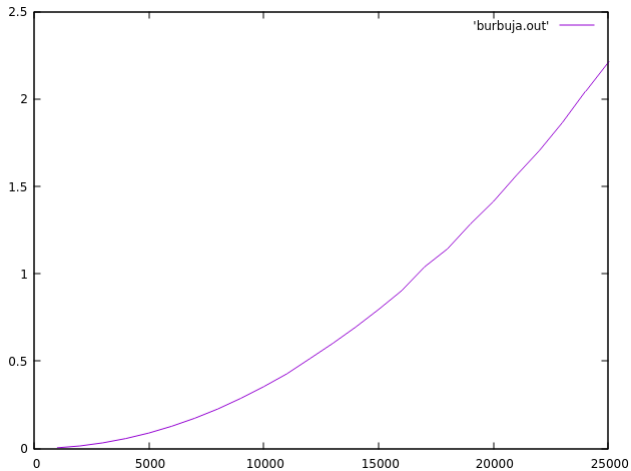
Índice

Tamaños de problema

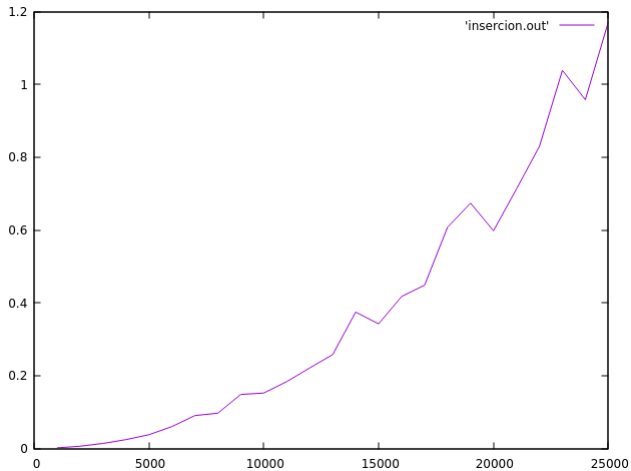
Algoritmo	Eficiencia	Tamaño inicial	Incremento
Burbuja	$O(n^2)$	1000	1000
Inserción	$O(n^2)$	1000	1000
Selección	$O(n^2)$	1000	1000
Mergesort	$O(n \cdot \log(n))$	1.000.000	500.000
Quicksort	$O(n \cdot \log(n))$	1.000.000	500.000
Heapsort	$O(n \cdot \log(n))$	1.000.000	500.000
Floyd	$O(n^3)$	100	100
Hanoi	$O(2^n)$	10	1

Índice

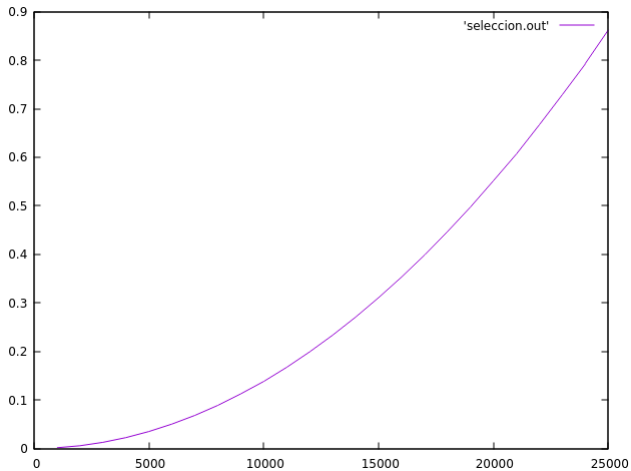
Algoritmo burbuja



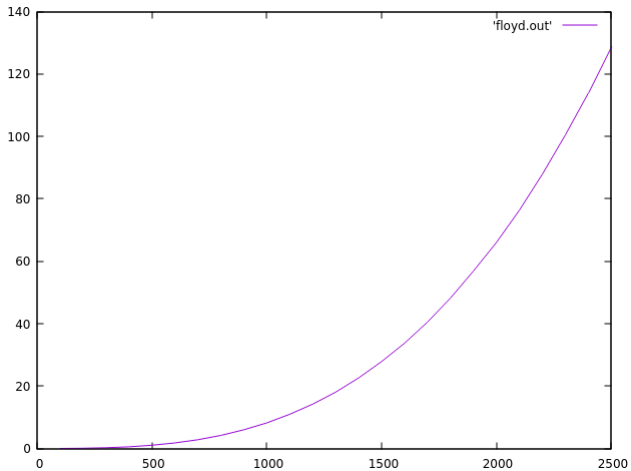
Algoritmo de inserción



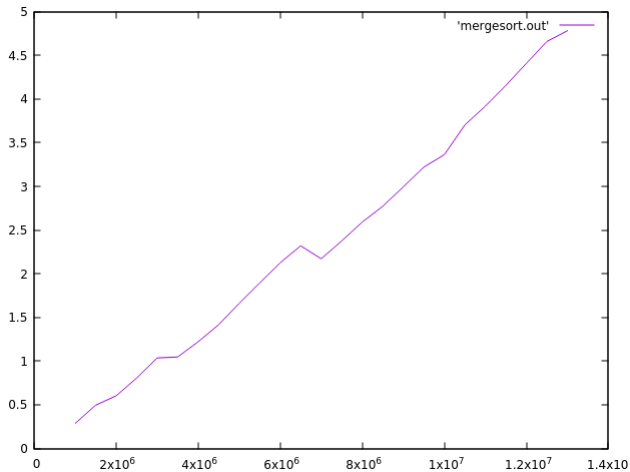
Algoritmo de selección



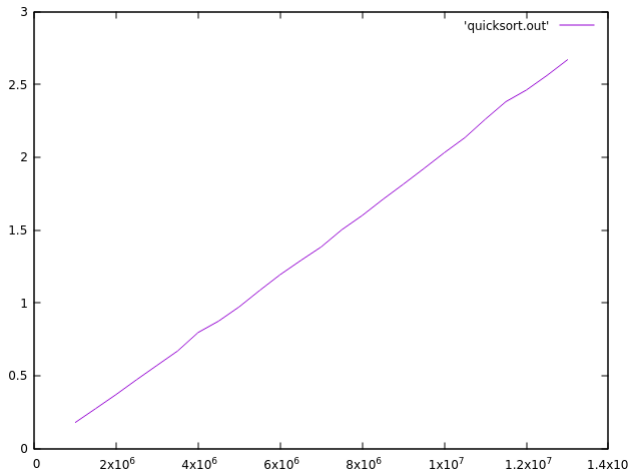
Algoritmo de Floyd



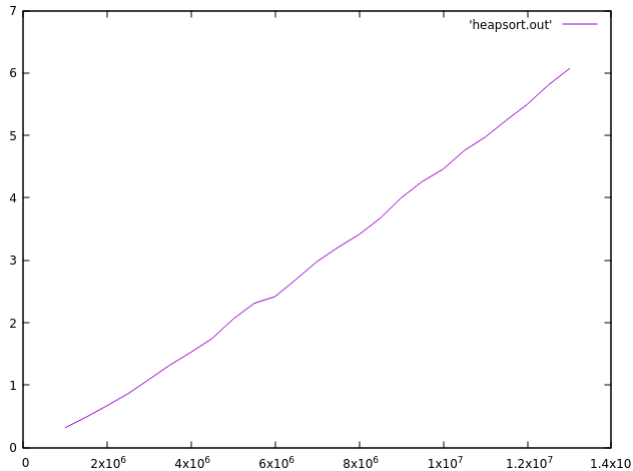
Algoritmo mergesort



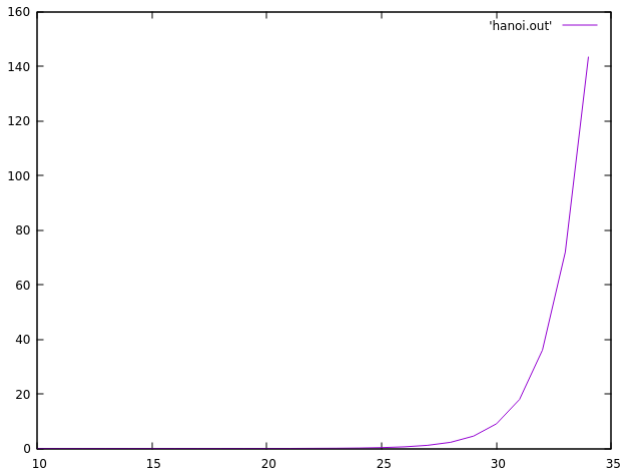
Algoritmo quicksort



Algoritmo heapsort



Algoritmo Hanoi



Índice

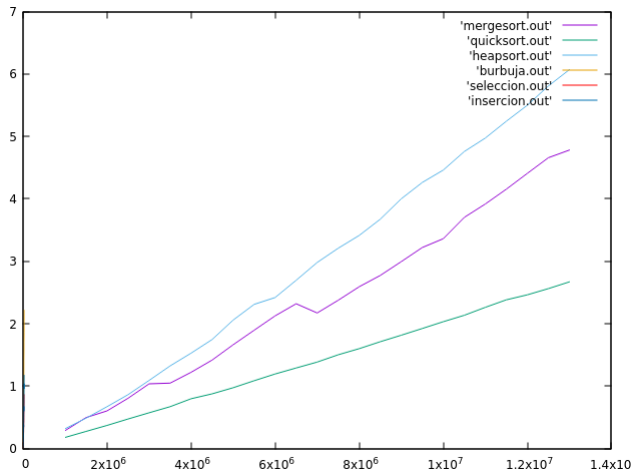
Variación de la eficiencia empírica

Principio de Invarianza

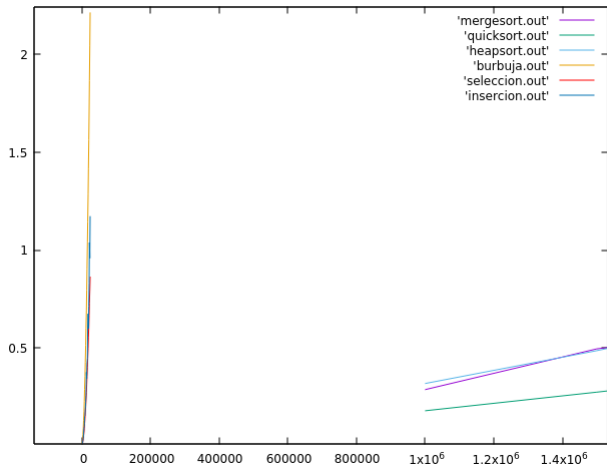
La eficiencia empírica varía al cambiar de plataforma, lenguaje, etc. como mucho en una constante.

Índice

Comparación entre algoritmos de ordenación



Comparación entre algoritmos de ordenación (zoom)



Índice

Índice

Índice

Blocks

Block Title

You can also highlight sections of your presentation in a block, with it's own title

Theorem

There are separate environments for theorems, examples, definitions and proofs.

Example

Here is an example of an example block.