

Laboratorio Nro. 2

Complejidad de algoritmos

Andrés Múnera
Universidad Eafit
Medellín, Colombia
ajmunerau@eafit.edu.co

Jorge Ruiz
Universidad Eafit
Medellín, Colombia
jdruizl@eafit.edu.co

3) Simulacro de preguntas de sustentación de Proyectos

3.1 Insertion sort

n	tiempo (ms)
5000	25
10000	143
15000	215
20000	233
25000	236
30000	273
35000	278
40000	706
45000	544
50000	592
55000	702
60000	887
65000	1047
70000	1138
75000	1329
80000	1643
85000	1779
90000	1862
95000	2256
100000	2891

PhD. Mauricio Toro Bermúdez
Docente | Escuela de Ingeniería | Informática y Sistemas
Correo: mtorobe@eafit.edu.co | Oficina: Bloque 19 – 627
Tel: (+57) (4) 261 95 00 Ext. 9473

ESTRUCTURA DE DATOS 1

Código ST0245

Merge sort

n	tiempo (ms)
1000000	199
2000000	276
3000000	956
4000000	708
5000000	1266
6000000	1361
7000000	1447
8000000	1411
9000000	1597
10000000	1918
11000000	1645
12000000	2110
13000000	2392
14000000	2397
15000000	2269
16000000	2414
17000000	2876
18000000	2825
19000000	3187
20000000	3355

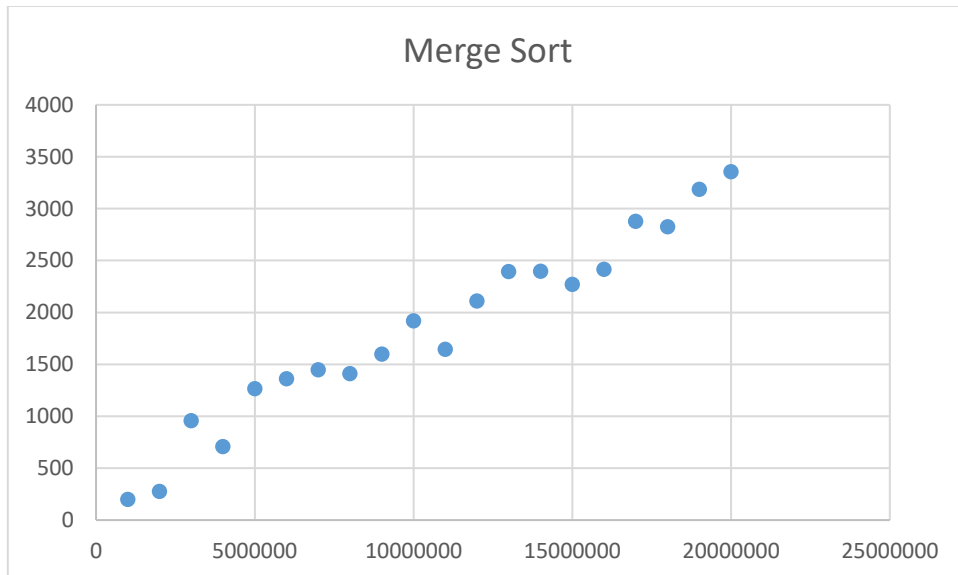
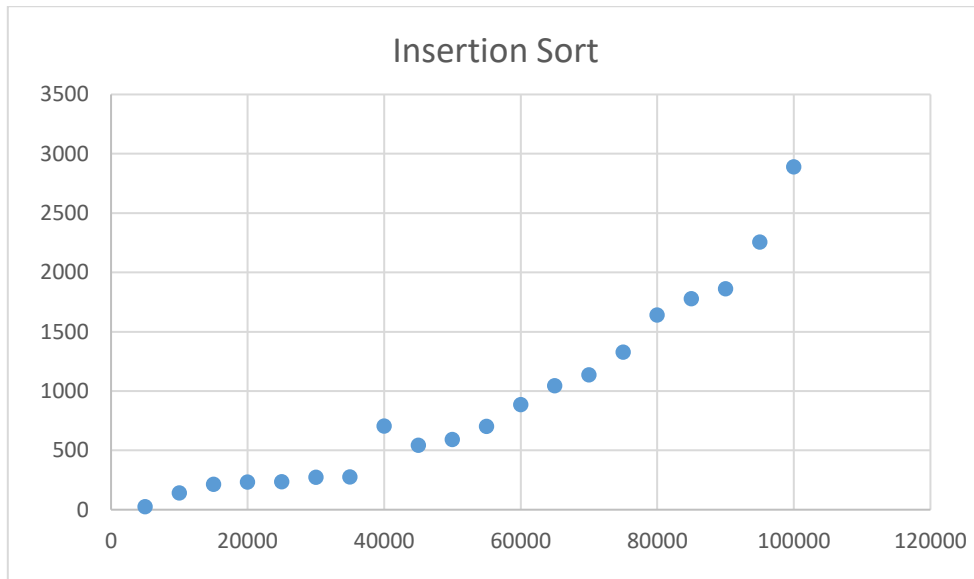
3.2

PhD. Mauricio Toro Bermúdez

Docente | Escuela de Ingeniería | Informática y Sistemas
 Correo: mtorobe@eafit.edu.co | Oficina: Bloque 19 – 627
 Tel: (+57) (4) 261 95 00 Ext. 9473

ESTRUCTURA DE DATOS 1

Código ST0245



3.3

3.4 No sería lo mejor ya que es un poco lento, el juego se vería *lagueado* y nadie lo jugaría.

3.5 En el único caso que sería más rápido es en el que ya este organizado

3.6

3.7

Array 2

```
public int countEvens(int[] nums) {
    int res = 0;
    for(int n: nums){
        if(n % 2 == 0){
```

PhD. Mauricio Toro Bermúdez

Docente | Escuela de Ingeniería | Informática y Sistemas
 Correo: mtorobe@eafit.edu.co | Oficina: Bloque 19 – 627
 Tel: (+57) (4) 261 95 00 Ext. 9473

ESTRUCTURA DE DATOS 1

Código ST0245

```

        ++res;
    }
}
return res;
}

```

= O(n)

```

public int bigDiff(int[] nums) {
    int min = Integer.MAX_VALUE;
    int max = Integer.MIN_VALUE;
    for(int n: nums){
        max = Math.max(max,n);
        min = Math.min(min,n);
    }
    return max - min;
}
= O(n)

```

```

public int centeredAverage(int[] nums) {
    int min = Integer.MAX_VALUE;
    int max = Integer.MIN_VALUE;
    int suma = 0;
    for(int n: nums){
        max = Math.max(max,n);
        min = Math.min(min,n);
        suma = suma + n;
    }
    return (suma-max-min)/(nums.length - 2);
}

```

= O(n)

```

public int sum13(int[] nums) {
    int suma = 0;
    boolean immediate = false;
    for(int n: nums){
        if(n != 13){
            if(!immediate){
                suma = suma + n;
            }
            else{
                immediate = false;
            }
        }
        else{
            immediate = true;
        }
    }
}

```

PhD. Mauricio Toro Bermúdez

Docente | Escuela de Ingeniería | Informática y Sistemas
 Correo: mtorobe@eafit.edu.co | Oficina: Bloque 19 – 627
 Tel: (+57) (4) 261 95 00 Ext. 9473

ESTRUCTURA DE DATOS 1

Código ST0245

```

    }
    }
    return suma;
}

```

= O(n)

```

public boolean has22(int[] nums) {
    for(int i = 0; i < nums.length - 1; ++i){
        if((nums[i] == 2) && (nums[i] == nums[i+1])){
            return true;
        }
    }
    return false;
}

```

= O(n)

Array 3

```

public int maxSpan(int[] nums) { //Tomado de Arthur Manning
    int dist = 0;

    for(int i = 0; i < nums.length; i++){
        int j = nums.length - 1;
        while(nums[i] != nums[j])
            j--;

        if(dist < j - i + 1)
            dist = j - i + 1;
    }
    return dist;
}

public int countClumps(int[] nums) {
    int a = 0;
    int b = -1;
    for(int i = 0; i < nums.length - 1; i++) {
        if(nums[i] == nums[i + 1] && nums[i] != b) {
            b = nums[i];
            a++;
        }
        for(int j = 0; j < nums.length - 1; j++) {
            if(nums[j] != b) {
                b = -1;
            }
        }
    }
}

```

PhD. Mauricio Toro Bermúdez

Docente | Escuela de Ingeniería | Informática y Sistemas
 Correo: mtorobe@eafit.edu.co | Oficina: Bloque 19 – 627
 Tel: (+57) (4) 261 95 00 Ext. 9473

ESTRUCTURA DE DATOS 1

Código ST0245

```

    }
    return a;
}
}

public boolean canBalance(int[] nums) {
    for(int i = 1; i < nums.length; i++) {
        int sum1 = 0, sum2 = 0;
        for(int a = 0; a < i; a++) sum1 += nums[a];
        for(int b = i; b < nums.length; b++) sum2 += nums[b];
        if(sum1 == sum2) return true;
    }
    return false;
}

public int[] seriesUp(int n) {
    int [] a = new int [n*(n + 1)/2];
    int b = 0;
    for(int i = 1; i<=n; i++){
        for(int j = 1; j<=i; j++){
            a[b++]=j;
        }
    }
    return a;
}

public int[] squareUp(int n) {
    int [] a = new int [n*n];
    int b = 0;
    for(int i =1; i<=n; i++){
        for(int j = n; j>=1; j--){
            a[b++]=j;
            if(i<j)
                a[b-1] = 0;
        }
    }
}

```

3.8

n: longitud del arreglo

4) Simulacro de Parcial

4.1 C

4.2 B

4.3 D

4.4 B

4.5.1D

PhD. Mauricio Toro Bermúdez

Docente | Escuela de Ingeniería | Informática y Sistemas
 Correo: mtorobe@eafit.edu.co | Oficina: Bloque 19 – 627
 Tel: (+57) (4) 261 95 00 Ext. 9473

ESTRUCTURA DE DATOS 1
Código ST0245

4.5.2A
4.6 10 segundos
4.7
4.8 A
4.9 A
4.11 C
4.12 B
4.13 C
4.14 A

PhD. Mauricio Toro Bermúdez

Docente | Escuela de Ingeniería | Informática y Sistemas
Correo: mtorobe@eafit.edu.co | Oficina: Bloque 19 – 627
Tel: (+57) (4) 261 95 00 Ext. 9473

