ESTRUCTURA DE DATOS 1
Código ST0245

Laboratory practice No. 1

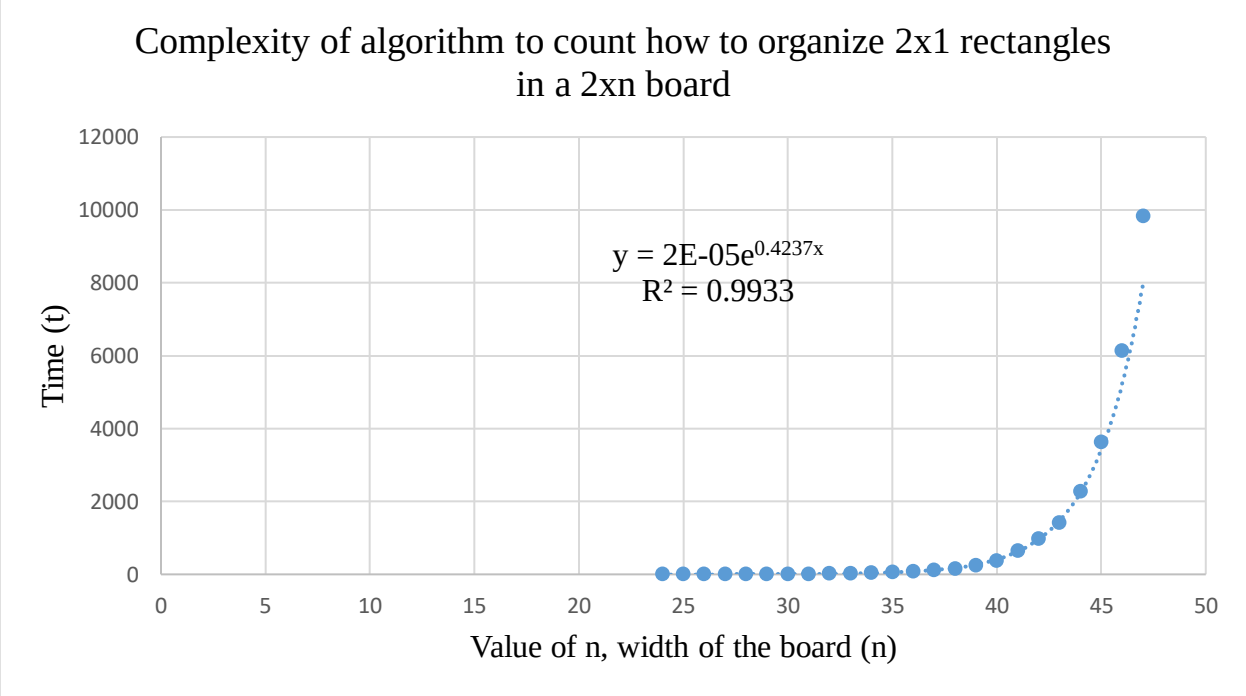
Gregorio Pérez Bernal
Universidad Eafit
Medellín, Colombia
gperezb1@eafit.edu.co

Luisa Toro Villegas
Universidad Eafit
Medellín, Colombia
ltorov@eafit.edu.co

3) Practice for final project defense presentation

3.1 T(n)=T(n-1)+T(n-2)+c
And when you solve it, it throws a complexity of 2^n.

3.2



For n=50, it would be approximately 39474.

PhD. Mauricio Toro Bermúdez
Professor | School of Engineering | Informatics and Systems
Email: mtorobe@eafit.edu.co | Office: Building 19 – 627
Phone: (+57) (4) 261 95 00 Ext. 9473

3.3

No, because it is an exponential complexity, which means that with big numbers, it would take forever to throw an answer. For example, with n=64 it would take the time that earth has existed.

3.5

The majority of the complexities of the exercises in coding bat in recursion 1 are n, due to the fact that it has a recursion in each method.

For recursion 2, the complexities are 2^n, because it has two recursive calls per exercise.

3.6

N refers to the term we will find, the number of recursions that will be tried. For example, n=50 in Fibonacci refers to the 50th term of the sequence.

4) Practice for midterms

4.1
```
static boolean isPal(String s) {
if(s.length() == 0 || s.length() == 1)
return true;
if(s.chartAt(0)== s.charAt(s.length() -1)) return isPal(s.substring(1,
s.length()-1));
return false;
}
```

4.2 d) T(n)=T(n−1)+C

4.3

```
int solucionar (int n, int a, int b, int c) {
if(n==0||(n<a&&n<b&&n<c)) return 0;
int res = solucionar(n-a,a.b,c) + 1;
res = Math.max(res,solucionar(n-b,a,b,c)+1);
res = Math.max(res,solucionar(n-c,a,b,c)+1);
return res;
}
```

4.4 e) La suma de los elementos del arreglo a y es O(n)

4.5
```
int ways(int T){

if (T == 0)return 1;
```

PhD. Mauricio Toro Bermúdez
Professor | School of Engineering | Informatics and Systems
Email: mtorobe@eafit.edu.co | Office: Building 19 – 627
Phone: (+57) (4) 261 95 00 Ext. 9473

```
if (T<0) return 0;

int f1 = ways(T - 3);

int f2 = ways(T - 5);

int f3 = ways(T - 7);

return f1+f2+f3;

}
```

4.6

```
public int suma(String n) {

return sumaAux(n, 0); 3

}
private int sumaAux(String n, int i){

if (i >= n.length()) return 0;
if(i + 1 < n.length() &&

n.charAt(i) == n.charAt(i + 1)){

return sumaAux(n, i+2);

 }

return (n.charAt(i) - '0') + sumaAux (n,i+1);

}
```

4.8

```
public int cuantas(int K, int[] v, int n){

        if(K == 0) return 1;

        boolean imposible;

        imposible = n <= 0 && K >= 1;

        imposible = imposible || K < 0;
```

PhD. Mauricio Toro Bermúdez
Professor | School of Engineering | Informatics and Systems
Email: mtorobe@eafit.edu.co | Office: Building 19 – 627
Phone: (+57) (4) 261 95 00 Ext. 9473

```
        if(imposible)return 0;

        int ni = cuantas(K, v, n - 1);

        int nj = cuantas(K - v[n-1], v,  n);

        int suma =  ni+nj ;

        return suma;}
```

4.9 c. 22

4.10 b. 6

4.11

```
int lucas(int n){
     if(n == 0) return 2;
     if(n == 1) return 1;
     return lucas(n-1) + lucas (n-2);
     }
```

c. T(n)=T(n−1)+T(n−2)+c, que es O(2^n)

4.12

```
int conejo(char[][] A, int n, int m, int i, int j,int d, int k){

           if (i>= n|| j>= m) return 0;

           int sat = 0;

           if (A[i][j] == ´x´) sat = d;

           if (A[i][j]== ´#´) sat = k;

           if (i == n-1 && j == m-1) return sat;

           int fi = conejo (A,n,m,i+1,j,d,k);

           int fj = conejo (A,n,m,i,j+1,d,k);

           sat += Math.max(fi+fj);

           return sat;

}
```

PhD. Mauricio Toro Bermúdez
Professor | School of Engineering | Informatics and Systems
Email: mtorobe@eafit.edu.co | Office: Building 19 – 627
Phone: (+57) (4) 261 95 00 Ext. 9473