Para el siguiente algoritmo:

```
algoritmo(n)

    i = 0
    res = 0

    while(i <= 2n+1)

        j = 1
        s = 3

        while(j<n)
            s+=4
            j++

        i++
        res = res + 4s
    print res
```

1

1

Valido

$0, 1, 2, \ldots 2n+1, 2n+2 \leftarrow 2n+3$

$2n+2$

$2n+2$

$1 \ldots n-1, n$ $n(2n+2)$

$(n-1)(2n+2)$

$2n+2$

$O(n^2)$

Ciclo interno

(j, s)

Est int $(1, 3)$

$(j, S) \rightarrow (j+1, S+4)$

Inv

$(1, 3) \rightarrow (2, 3+4) \rightarrow (3, 3+4+4)$

$(k, 3+4(k-1))$

Dem $k=1$

$(1, 3+4(0))$ ✓

$k=n$

$(n, 3+4(n-1))$

EsF

$(n, 3+4(n-1))$

$(k, 3+4(k-1))$

$(k+1, 3+4(k-1)+4)$

$(k+1, 3+4k)$ ✓

(i, res)

$(0, 0)$

$$\Big(2n+2,\ \underbrace{4(2n+2)\big(3+4(n-1)\big)}_{\text{Sol, do total}}\Big)$$

$(i, res) \to (i+1,\ res + 4(3+4(n-1)))$

$(0,0) \to (1,\ 0+4(3+4(n-1)))$

$\to 2,\ 0+4(2)(3+4(n-1))$

$(k,\ 4\cdot k(3+4(n-1)))$

$k=0$

$(0,\ 0)$ ✓

$(k+1,\ \underbrace{4k(3+4(n-1))}_{res}+4(3+4(n-1)))$

$k+1,\ 4(k+1)(3+4(n-1))$

$$T(n) = 3T\left(\frac{n}{4}\right) + \frac{n^2}{2} \qquad T(1) = O(1)$$

$$T(n) = 3\left(3T\left(\frac{n}{4^2}\right) + \left(\frac{n}{4}\right)^2 \times \frac{1}{2}\right) + \frac{n^2}{2} \qquad \left(\frac{3}{4}\right)^i \qquad \left(\frac{3}{4}\right)^0 \quad \left(\frac{3}{4}\right)^1$$

$$T(n) = \frac{3^i}{2} T\left(\frac{n}{4^i}\right) + \left(3^{i-1}\frac{n^2}{4^{i-1}} + 3^{i-2}\frac{n^2}{4^{i-2}} + \dots + \frac{n^2}{4^2}\right) \qquad \left(\frac{3}{4}\right)^i$$

$$\frac{n}{4^i} = 1 \qquad i = \log_4(n) \qquad < 1$$

$$3^{\log_4(n)} \times O(1) + \sum_{i=1}^{\log_4(n)-1} \frac{n^2}{4^i} \rightarrow n^{\log_4(3)} + n^2\left(\frac{\left(\frac{1}{4}\right)^{\log_4(n)} - 1}{\frac{1}{4} - 1}\right)$$

$$n^{\log_4(3)} + n^2\left(1 - n^{\log_4\left(\frac{1}{4}\right)}\right)$$

$$n^{0.7} \qquad O(n^2) \quad n^{0.8}$$

$$F(n) \in O(g(n))$$

$$F(n) \leq C \times g(n)$$

$$n^2 \, 0.^{-0.8} = n^{1.2}$$

3. Diseño de soluciones [35 puntos]

Se tienen n puntos bidimensionales $((x_1, y_1), (x_2, y_2), ...(x_n, y_n))$ y se desea ordenarlos de tal forma se cumple que $(x'_i + y'_i) <= (x'_{i+1} + y'_{i+1}), 1 \leq i \leq n-1$. Es decir, los ordenamos ascendentemente tomando como criterio la suma de los puntos. Ejemplo: $((1,2)(2,3)(1,1))$ da como resultado $((1,1),(1,2),(2,3))$

1. **(5 puntos)** Explique cómo es la solución ingenua e indique su complejidad en términos de $O(f(n))$ con una cota apropiada
2. **(10 puntos)** Explique cómo sería una solución en tiempo $O(n^2)$. Indique con pseudocódigo su implementación
3. **(20 puntos)** Diseñe una solución usando divide y vencerás. Explique cual es la estrategia de dividir y combinar. Así mismo explique la solución trivial. Escriba un pseudocódigo de su implementación. Calcule la complejidad de la solución.

$X = (X_1, X_2, X_3, X_n)$

1) Solucion ingenua

¿Cuanto me cuesta listar TODAS las ordenaciones?

$(X_1, X_2) \neq (X_2, X_1)$

$P(n,n) = n!$

$\underline{n}\ \underline{n-1}\ \underline{n-2} \cdots 1 = n!$

Para i = 1 hasta n
    Para j = i+1 hasta n
        Si Ax[i]+Ay[i]>= Ax[j]+Ay[j]
            A[i] <-> A[j]

Dividir : Partir el array en n/2 cada vez
Vencer: Un array de tamaño 1 esta ordenado
Combinar: Dados dos array A y B, que vienen ordenas, entregar uno ordenado combinado

```
Dividir(A,p,r)
    Si p < r
        q = p + r
            -------
               2
         Dividir(A,p,q)
         Dividir(A,q+1,r)
         Combinar(A,p,q,r)
```

n

$O(1)$

```
Combinar(A,p,q,r)
     X = A[p...q]
     Y = A[q+1...r]
      i, j = 1

      Para k = p hasta r
           if X[i]x + X[i]y > Y[j]x+Y[j]y
                A[k] = Y[j]
                j = j+1
            Si no
                A[k] = X[i]
                i = i+1
```

n

$O(n)$

$T(n) = 4T\left(\frac{n}{b}\right) + F(n)$

$T(n) = 2T\left(\frac{n}{2}\right) + \underbrace{O(1)}_{\text{Dividir}} + \underbrace{O(n)}_{\text{Combinar}}$

$O(n)$

$\log_2(2) = n$ n^1

2) $F(n)$ es $\Theta(n)$ $\Theta(\log_b(n)(n^{\log_b a}))$

$\Theta(\log(n) \times n)$