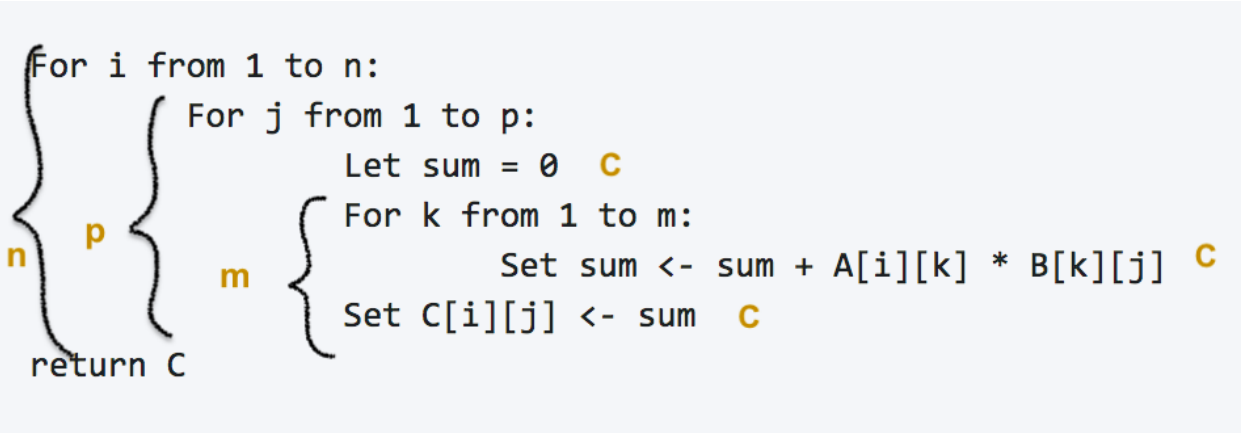
Problema 1

Algorithm 1 Busqueda Lineal

```
1: procedure BUSQUEDA(xs, v)
2:     for i ← 1 to len(xs) + 1 do
3:         if i > len(xs) then
4:             return null
5:         else if v equals xs[i] then
6:             return i
7:         i ← i + 1
```

Loop Invariant: todos los valores anteriores al indice **i** no coinciden con el valor que se busca **v**

Problema 2



O($n \times p \times m$)

Debido a que debe de correr nested loops O($n \times n \times n$)

El running time del algoritmo es O(n^3)

Problema 3

Best case Bubble Sort: Ω (n)
Ocurre cuando todos los valores del arreglo se encuentran de forma ascendente, si este es el orden que se desea. En este caso solo compara los valores pero nunca realiza el swap.

Best case Insertion Sort: Ω (n)
Igual que en Bubble Sort, ocurre cuando todos los valores del arreglo se encuentran de forma ascendente, si este es el orden que se desea. Igualmente nunca se realiza un swap, solamente compara los valores

Problema 3

Worst case Bubble Sort: O(n^2)
Ocurre cuando todo el arreglo esta de forma descendente y se quiere ordenar ascendentemente o viceversa, se hace compara por parejas y se hacen swap todas las posiciones que encuentre en el camino hasta su lugar debido, esto involucra recorrer el arreglo varias veces desde el inicio hasta el final

Worst case Insertion Sort: O(n^2)
También ocurre cuando todos los valores del arreglo se presentan de forma descendente y el objetivo es que estén ordenados ascendentemente, este algoritmo compara parejas y las hace swap con la diferencia que almacena un indice para luego de dejar el valor en su lugar indicado, regresa a este indice y así no recorrer el arreglo como en Bubble Sort