

# Problemas

Conteo de instrucciones

Esquemas básicos

## Bucles simples

1

```
for (i = 10; i <= n + 5; i += 2)  
    op();
```

2

```
for (i = 1; i <= n; i *= 2)  
    op();
```

3

```
for (i = 10; i < n + 5; i *= 3)  
    op();
```

4

```
for (i = 1; i <= n * n * n; i *= 2)  
    op();
```

5

```
for (i = 1; i * i <= n; i++)  
    op();
```

## Bucles anidados independientes

1

```
for (i = 10; i < n; i++)  
  for (j = 0; j < n; j += 2)  
    op();
```

2

```
for (i = 0; i < n; i++)  
  for (j = 0; j < 100; j++)  
    op();
```

3

```
for (i = 0; i < n; i++) {  
  for (j = 0; j < n; j++)  
    op();  
  
  for (j = 1; j < n; j *= 2)  
    op();  
}
```

## Bucles anidados dependientes

1

```
for (i = 1; i <= n; i++)  
  for (j = 1; j <= i; j ++)  
    op();
```

2

```
for (i = 1; i <= n * n - 10; i++)  
  for (j = 1; j <= i; j ++)  
    op();
```

3

```
for (i = 1; i <= n; i++)  
  for (j = 1; j <= n; j += i)  
    op();
```

Problemas

## Ejercicio 1

```
encontrado = false
for i = 1 to n - 1
    if x[i] == y[1]
        if x[i + 1] == y[2]
            encontrado = true
            lugar = i
    i = i + 1
```

Calcular el **tiempo promedio** de ejecución, suponiendo que una probabilidad **p** de que dos elementos de x e y sean iguales.

## Ejercicio 2

```
p := 0
for i = 1 to n
    p := p + i*i
    for j = 1 to p
        escribir(a[p,j])
```

Cálculo el número de llamadas a **escribir()**



## Ejercicio 3

```
for i = 1 to n
  c[i,j] = a[i, 2] * a[i, i]
  for j = i + 1 to n
    if par(a[i, j])
      c[i, j]--
```

1. Número promedio de asignaciones  $c[i,j] = a[i, 2] * a[i, i]$
2. Número promedio de comparaciones  $\text{par}(a[i, j])$
3. Número promedio de ejecuciones  $c[i, j]--$

## Ejercicio 4

```
if a[1] > a[2]
    max = a[1]
else
    max = a[2]

cont = 0
for i = 3 to n
    if a[i] > max
        cont++
```

← (I)

Número promedio de ejecuciones de (I)

\* Asumimos que el rango de valores de  $n$  es amplio

## Ejercicio 5

```
if a[1] > a[2]
    max = a[1]
else
    max = a[2]

cont = 0
for i = 3 to n
    if a[i] > max
        cont++
```

← (I)

Número promedio de ejecuciones de (I)

\* Asumimos que el rango de valores es 1, 2, 3, 4 y 5

## Ejercicio 6

```
max = MAX_INT
for i = 1 to n
  for j = 1 to m
    if a[i, j] > max
      max = a[i, j] ← (I)
```

1. Cota inferior del número de veces que se ejecuta (I)
2. Cota superior del número de veces que se ejecuta (I)
3. Numero promedio de veces que se ejecuta (I)

## Ejercicio 7

```
def Una(Imagen[n, n], m)
  i = m + 1
  repeat
    for j = m + 1 to n - m
      Otra(Imagen, i, j, m)

    i = i + 1
  until i > n - m
```

```
def Otra(Imagen[n, n], x, y, tam)
  acum = 0
  for i = y - tam to y + tam
    for j = x - tam to x + tam
      acum = acum + Imagen[i, j]
```

← (I)

Número promedio de ejecuciones de (I)

## Ejercicio 8

```
x = 0
y = 0
for i=1 to n
  if par(i)
    for j = i to n
      x++
  else
    for j =1 to i - 1
      y++
```

Número promedio de ejecuciones de **x++** e **y++**

## Ejercicio simple repaso

```
count = 0
for i = 1 to n * n
  for j = 1 to i
    for k = 1 to 6
      count++
```

1. Contar el número de instrucciones **count++**
2. Contar el número de instrucciones totales