

Algoritmos de ordenación.

Ordenación rápida (quicksort)

$$P : \{v = V[0 \dots N] \wedge N \geq 0\}$$

$$Q : \{\textit{Ordenado}(V, 0, N)\}$$

Inmersión :

$$P : \{v = V[0 \dots N] \wedge N \geq 0 \wedge 0 \leq c \leq f \leq N\}$$

Quicksort

$$Q : \{\textit{Ordenado}(V, c, f)\}$$

Análisis de casos :

Casos	Algoritmo
$c \geq f$	<i>return v</i>
$c < f$	<i>particion(v, c, f, p); quicksort(V, c, p - 1); quicksort(V, p + 1, f)</i>

Ordenación rápida (quicksort)

Algoritmo:

```
if (c < f) { particion(v,c,f,p);
```

```
    quicksort(v, c, p-1);
```

Recursión múltiple.

```
    quicksort(v, p+1, f);
```

```
particion(vector v; int c; int f; int p) {
```

```
    piv = v[c]; i = c + 1; d = f;
```

```
    while (i != d + 1) {
```

```
        while ((i <= d) && (v[i] <= piv) i++;
```

```
        while ((i <= d) && (v[d] >= piv) d--;
```

```
        if (i < d) { intercambiar(v,i,d); i++; d--; }
```

```
    }
```

```
    intercambiar(v,c,d);
```

```
    p = d;
```

```
}
```