Selection - Pior caso: $O\left(n^2\right)$ - elementos ordenados em ordem inversa

Código	Custo
`long int min, tmp`	2
`for(long int i = 0; i < size - 1; i++)`	$\sum_{i=0}^{n-2}$
`    min = i`	1
`    for(long int j = i + 1; j < size; j++)`	$\sum_{j=i+1}^{n-1}$
`        if(vector[j] < vector[min])`	1
`            min = j`	1
`    tmp = vector[min]`	1
`    vector[min] = vector[i]`	1
`    vector[i] = tmp`	1

Cálculo da função $T(n)$

$$
\begin{aligned}
T(n) &= 2 + \sum_{i=0}^{n-2}\left(1 + \sum_{j=i+1}^{n-1} 2 + 1 + 1 + 1\right) \\
&= 2 + \sum_{i=0}^{n-2}\left(4 + \sum_{j=i+1}^{n-1} 2\right) \\
&= 2 + \sum_{i=0}^{n-2} 4 + \sum_{i=0}^{n-2}\sum_{j=i+1}^{n-1} 2 \\
&= 2 + 4n - 4 + \sum_{i=0}^{n-2} 2(n-2) \\
&= 2 + 4n - 4 + 2n^2 - 4n \\
&= 2n^2 - 2 \Rightarrow O\left(n^2\right)
\end{aligned}
$$

Selection - Melhor caso: $O\left(n^2\right)$ - elementos ordenados

Código	Custo
`long int min, tmp`	2
`for(long int i = 0; i < size - 1; i++)`	$\sum_{i=0}^{n-2}$
`    min = i`	1
`    for(long int j = i + 1; j < size; j++)`	$\sum_{j=i+1}^{n-1}$
`        if(vector[j] < vector[min])`	1
`            min = j`	0
`    tmp = vector[min]`	1
`    vector[min] = vector[i]`	1
`    vector[i] = tmp`	1

Cálculo da função $T(n)$
$T(n) = 2 + \sum_{i=0}^{n-2}\left(1 + \sum_{j=i+1}^{n-1} 1 + 1 + 1 + 1\right)$
$= 2 + \sum_{i=0}^{n-2}\left(4 + \sum_{j=i+1}^{n-1} 1\right)$
$= 2 + \sum_{i=0}^{n-2} 4 + \sum_{i=0}^{n-2} \sum_{j=i+1}^{n-1} 1$
$= 2 + 4n - 4 + \sum_{i=0}^{n-2} n - 2$
$= 2 + 4n - 4 + n^2 - 2n$
$= n^2 + 2n - 2 \Rightarrow O\left(n^2\right)$

Insertion - Pior caso: $O\left(n^2\right)$ - elementos ordenados em ordem inversa

Código	Custo
`long int tmp, i`	2
`for(long int j = 1; j < size; j++){`	$\sum_{j=1}^{n-1}$
`    tmp = vector[j]`	1
`    for(i = j - 1; (i >= 0) && (vector[i] > tmp); i--)`	$\sum_{i=0}^{j-1}$
`        vector[i + 1] = vector[i]`	1
`    vector[i + 1] = tmp`	1

Cálculo da função $T(n)$

$$
\begin{aligned}
T(n) &= 2 + \sum_{j=1}^{n-1}\left(1 + \sum_{i=0}^{j-1} 1 + 1\right) \\
&= 2 + \sum_{j=1}^{n-1}\left(2 + \sum_{i=0}^{j-1} 1\right) \\
&= 2 + \sum_{j=1}^{n-1} 2 + \sum_{j=1}^{n-1}\sum_{i=0}^{j-1} 1 \\
&= 2 + 2(n-1) + \sum_{j=1}^{n-1} j \\
&= 2 + 2n - 2 + \frac{n(n-1)}{2} \\
&= \frac{n^2 + 5n}{2} \Rightarrow O\left(n^2\right)
\end{aligned}
$$

Insertion - Melhor caso: $O(n)$ - elementos ordenados

Código	Custo
`long int tmp, i`	2
`for(long int j = 1; j < size; j++){`	$\sum_{j=1}^{n-1}$
`    tmp = vector[j]`	1
`    for(i = j - 1; (i >= 0) && (vector[i] > tmp); i--)`	1
`        vector[i + 1] = vector[i]`	0
`    vector[i + 1] = tmp`	1

Cálculo da função $T(n)$

$$
\begin{aligned}
T(n) &= 2 + \sum_{j=1}^{n-1} (1 + 1 + 1) \\
&= 2 + \sum_{j=1}^{n-1} 3 \\
&= 2 + 3(n-1) \\
&= 2 + 3n - 3 \\
&= 3n - 1 \Rightarrow O(n)
\end{aligned}
$$