Ordenar

swap

```
1 void swap(reunion &r, int i, int j) {
2     pair<senial, hablante> aux = r[i];
3     r[i] = r[j];
4     r[j] = aux;
5 }
```

c_1	1
c_2	1
c_3	1

- $m = |r|$
- $T_{swap}(m) = c_1 + c_2 + c_3$
- $T_{swap}(m) \in O(1)$

valorAbsoluto

```
1 int valorAbsoluto(int &n){
2     if(n<0){
3         n = n * (-1);
4     }
5     return n;
6 }
```

c'_1	1
c'_2	1

- $T_{valorAbsoluto}(n) = c'_1 + c'_2$
- $T_{valorAbsoluto}(n) \in O(1)$

tono

```
1 float tono(senial s){
2     float sumatoria = 0;
3     for(int i=0; i < s.size(); i++){
4         sumatoria = sumatoria + valorAbsoluto(s[i]);
5     }
6     return sumatoria / s.size();
7 }
```

c''_1	1
c''_2	$n+1$
c''_3	n
c''_4	1

- $n = |s|$
- $T_{tono}(n) = c''_1 + c''_2 \cdot (n+1) + c''_3 \cdot n + c''_4$
- $T_{tono}(n) \in O(n)$

insert

```
1 void insert(reunion &r, int i) {
2     int j = i;
3     while (j > 0 && tono(r[j].first) > tono(r[j - 1].first)) {
4         swap(r, j, j - 1);
5         j--;
6     }
7 }
```

c'''_1	1
$c'''_2 \cdot (n+n)$	$m+1$
c'''_3	m
c'''_4	m

- $m = |r|$
- $n = |r[0].first|$
- $T_{insert}(m,n) = c'''_1 + c'''_2 \cdot 2n \cdot (m+1) + c'''_3 \cdot m + c'''_4 \cdot m$
- $T_{insert}(m,n) \in O(n \cdot m)$

insertionSort

```
1 void insertionSort(reunion &r) {
2     int i = 0;
3     while (i < r.size()) {
4         insert(r, i);
5         i++;
6     }
7 }
```

c_1''''	1
c_2''''	$m+1$
$c_3'''' \cdot n \cdot m$	m
c_4''''	m

- $m = |r|$
- $n = |r[0].first|$
- $T_{insertSort}(m,n) = c_1'''' + c_2'''' \cdot (m+1) + c_3'''' \cdot n \cdot m^2 + c_4'''' \cdot m$
- $T_{insertSort}(m,n) \in O(n \cdot m^2)$

ordenar

```
1 void ordenar(reunion &r, int freq, int prof) {
2     insertionSort(r);
3 }
```

$c_1''''' \cdot n \cdot m^2$	1

- $m = |r|$
- $n = |r[0].first|$
- $T_{ordenar}(m,n) = c_1''''' \cdot n \cdot m^2$
- $T_{ordenar}(m,n) \in O(n \cdot m^2)$