Algorithm TotalArea(Area, Angle, Data[0 ... n-1]).

Input: $Area$: The area of the radial diagram it's initializated in 0,
$Angle$: the angle that exist betwen two consecutive points,
$Data[0...n-1]$: the values of the n activities(0 - 100).
Output: $Area$: The total area of the radial diagram.

```
1  Angle ← sin(Angle)
2  for i ← 0 to n − 2 do
3      Area ← Area + (0.5 ∗ Data[i] ∗ Data[i + 1] ∗ Angle)
4  Area ← Area + (0.5 ∗ Data[0] ∗ Data[n − 1] ∗ Angle)
5  return Area
```

Algorithm MaxArea(Data[0 ... n-1], Ndata[0 ... n-1]).

Input: $Data[0...n-1]$: Array that contains the values of the n activities,
$Ndata[0...n-1]$: An empty array of the same length that Data[n].
Output: $Ndata[0...n-1]$: An array with the values of Data[0 ... n-1] sorted in away that allow to get a maximum area form the radial diagram.

```
1  if n >= 3
2  j ← 0
3  cont ← n − 1
4  Ndata[0] ← Data[0]
5  for i ← 0 to n − 1 do
6      if i mod 2 != 0
7          Ndata[j] ← Data[i]
8          j ← j + 1
9      else
10         Ndata[cont] ← Data[i]
11         cont ← cont − 1
12 return Ndata
```