**public** **static** **void** TriBulle(**int**[] tab)

{

**int** nbCase=tab.length;

**int** indiceCaseADroite=tab.length-1;

**int** tmp;

/\*

1 à nbCase pour savoir combien de fois

On doit chercher le max.

On pourrait boucler jusqu'a i<=nbCase,

Or, quand il n'y a plus qu'une case on sait

Qu'elle est forcement à la bonne place

\*/

**for**(**int** i=1;i<nbCase;i++)

{

**for**(**int** j=0;j<indiceCaseADroite;j++)

{

**if**(tab[j]>tab[j+1])

{

tmp=tab[j];

tab[j]=tab[j+1];

tab[j+1]=tmp;

}

}

//On décremente pour ne plus prendre en compte la case à droite

indiceCaseADroite--;

}

System.***out***.println(Arrays.*toString*(tab));

}

**public** **static** **void** TriInsert(**int** tab[])

{

**int** nbCase=tab.length;

**int** indiceCaseADroite=tab.length-1;

**int** max=0;

**int** position=0;

**int** tmp;

/\*

1 à nbCase pour savoir combien de fois

On doit chercher le max.

On pourrait boucler jusqu'a i<=nbCase,

Or, quand il n'y a plus qu'une case on sait

Qu'elle est forcement à la bonne place

\*/

**for**(**int** i=1;i<nbCase;i++)

{

**for**(**int** j=0;j<=indiceCaseADroite;j++)

{

//On initialise max à la 1ere valeur du tableau

**if**(j==0) {max=tab[j];}

//On conserve la valeur max et sa position

**if**(max<=tab[j])

{

max=tab[j];

position=j;

}

}

//Après avoir parcouru le tableau, on place

//tout à droite la valeur max

tmp=max;

tab[position]=tab[indiceCaseADroite];

tab[indiceCaseADroite]=tmp;

//On décremente pour ne plus prendre en compte la case à droite

indiceCaseADroite--;

}

System.***out***.println(Arrays.*toString*(tab));

}