**package** exemploum;

**public** **class** BubbleSort {

**public** **static** **void** main(String[] args) {

**int**[] vetor = {8, 9, 3, 5, 1};

**int** aux = 0;

//int i = 0;

System.***out***.println("Vetor Desordenado: ");

**for**(**int** i = 0; i < 5; i++){

System.***out***.println(" " + vetor[i]);

}

System.***out***.println("===================");

**for** (**int** i = 0; i < 5; i++){

//laço repete 4x, pois na quinta iria comparar o valor com uma posição que não existe.

**for**(**int** j = 0; j < 4; j++){

**if**(vetor[j] > vetor[j + 1]){

aux = vetor[j];

vetor[j] = vetor[j + 1];

vetor[j + 1] = aux;

}

}

}

System.***out***.println("Vetor Ordenado");

**for**(**int** i = 0; i <5; i++){

System.***out***.println(vetor[i]);

}

}

}

**package** exemplodois;

**public** **class** BubbleSort {

**public** **static** **void** main(String[] args) {

BubbleSort bubble = **new** BubbleSort();

**int**[] vetor = {64, 34, 25, 12, 22, 11, 90};

bubble.bubbleSortAsc(vetor);

System.***out***.println("Vetor Ordenado");

bubble.printVetor(vetor);

}

**private** **void** bubbleSortAsc(**int**[] vetor) {

**int** tamanhoVetor = vetor.length;

**for** (**int** i = 0; i < tamanhoVetor - 1; i++) {

**for** (**int** j = 0; j < tamanhoVetor - i - 1; j++){

**if** (vetor[j] > vetor[j+1]){

//troca temporária

**int** temp = vetor[j];

vetor[j] = vetor[j+1];

vetor[j+1] = temp;

}

}

}

}

**private** **void** printVetor(**int**[] vetor){

**int** tamanhoVetor = vetor.length;

**for** (**int** i = 0; i < tamanhoVetor; i++){

System.***out***.println(vetor[i]);

}

}

}

**package** quickSort;

**import** java.util.Arrays;

**public** **class** quicksortex01 {

**public** **static** **void** main(String[] args) {

**int**[] vetor = {6, 3, 4, 5, 2, 7, 1, 9, 8, 0};

*quickSort*(vetor, 0, vetor.length - 1);

//imprimir o vetor.

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

}

**private** **static** **void** quickSort(**int**[] vetor, **int** esquerdo, **int** direito){

**int** esq = esquerdo;

**int** dir = direito;

**int** pivo = vetor[(esq + dir) / 2];

**int** troca;

**while**(esq <= dir){

**while**(vetor[esq] < pivo){

esq++;

}

**while**(vetor[dir] > pivo){

dir--;

}

**if**(esq <= dir){

troca = vetor[esq];

vetor[esq] = vetor[dir];

vetor[dir] = troca;

esq++;

dir--;

}

}

//cliassifica recursivamente as sublistas da esquerda e da direita.

**if** (dir > esquerdo){

*quickSort*(vetor, esquerdo, dir);

}

**if** (esq < direito){

*quickSort*(vetor, esq, direito);

}

}

}