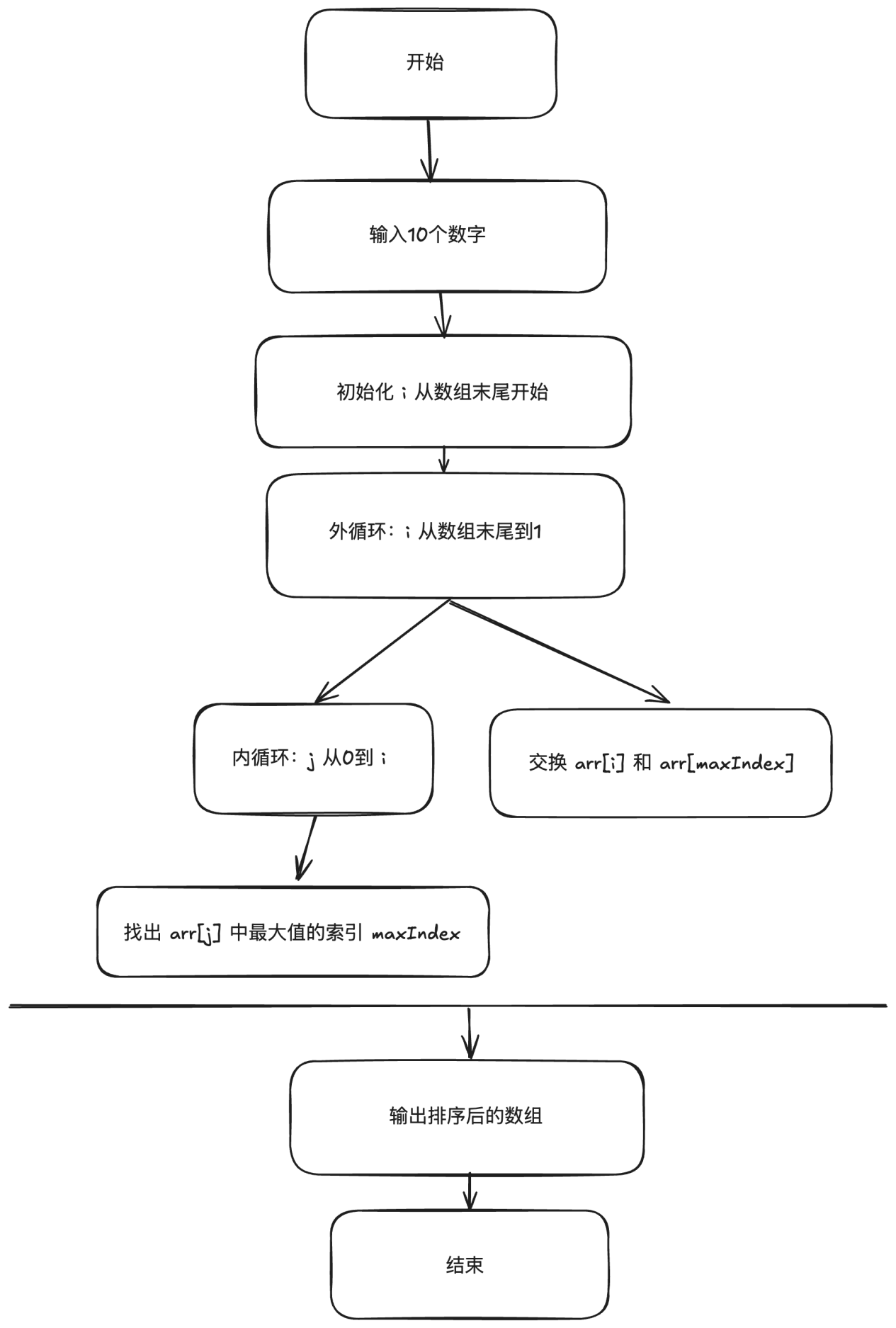
**实验二**

**1.题目：**

|  |
| --- |
| 改写选择排序算法：  在当前数组中找到最大值，然后将其与数组最后一个数进行交换。如此循环操作，直到数组最前端。编写测试程序，读取10个double型数字并排序后输出结果。 |

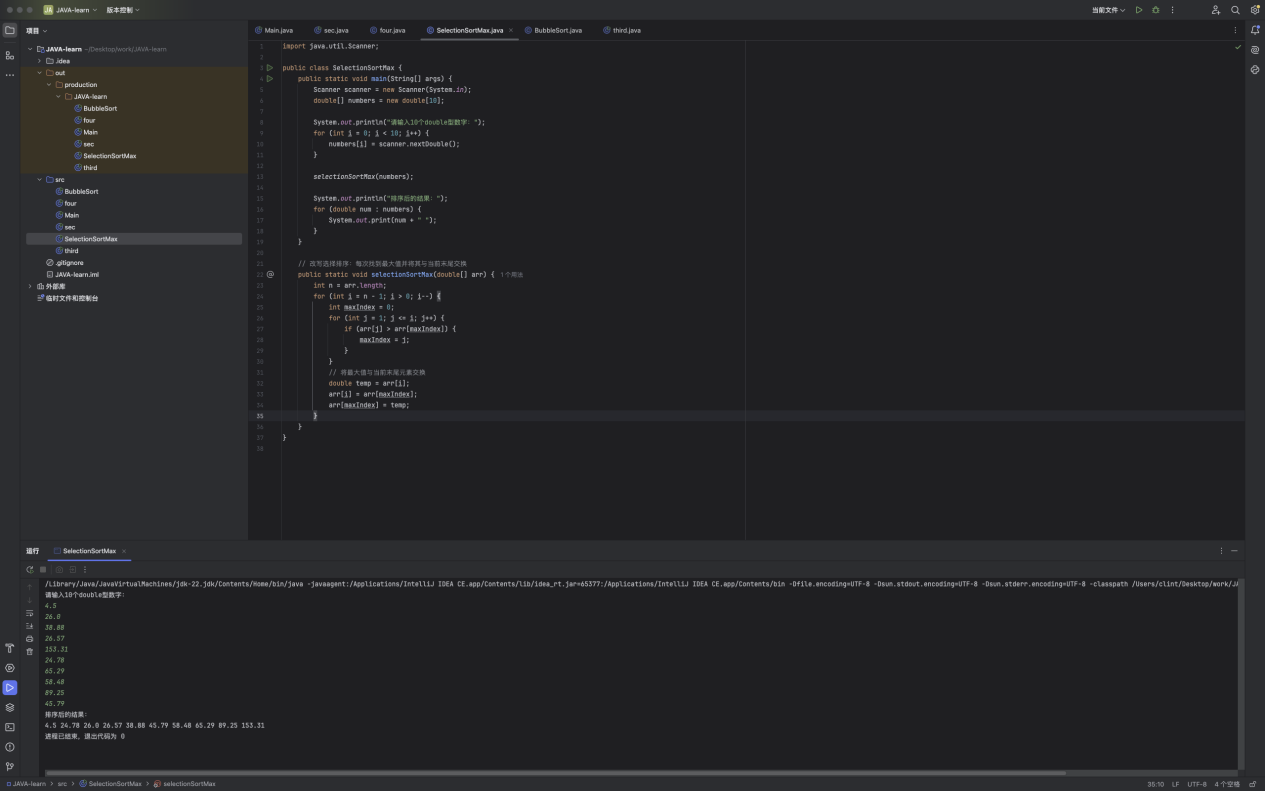
**思维导图：**



**完整代码：**

|  |
| --- |
| JavaScript import java.util.Scanner;  public class SelectionSortMax {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.*in*);  double[] numbers = new double[10];   System.*out*.println("请输入10个double型数字：");  for (int i = 0; i < 10; i++) {  numbers[i] = scanner.nextDouble();  }   *selectionSortMax*(numbers);   System.*out*.println("排序后的结果：");  for (double num : numbers) {  System.*out*.print(num + " ");  }  }   // 改写选择排序：每次找到最大值并将其与当前末尾交换  public static void selectionSortMax(double[] arr) {  int n = arr.length;  for (int i = n - 1; i > 0; i--) {  int maxIndex = 0;  for (int j = 1; j <= i; j++) {  if (arr[j] > arr[maxIndex]) {  maxIndex = j;  }  }  // 将最大值与当前末尾元素交换  double temp = arr[i];  arr[i] = arr[maxIndex];  arr[maxIndex] = temp;  }  } } |

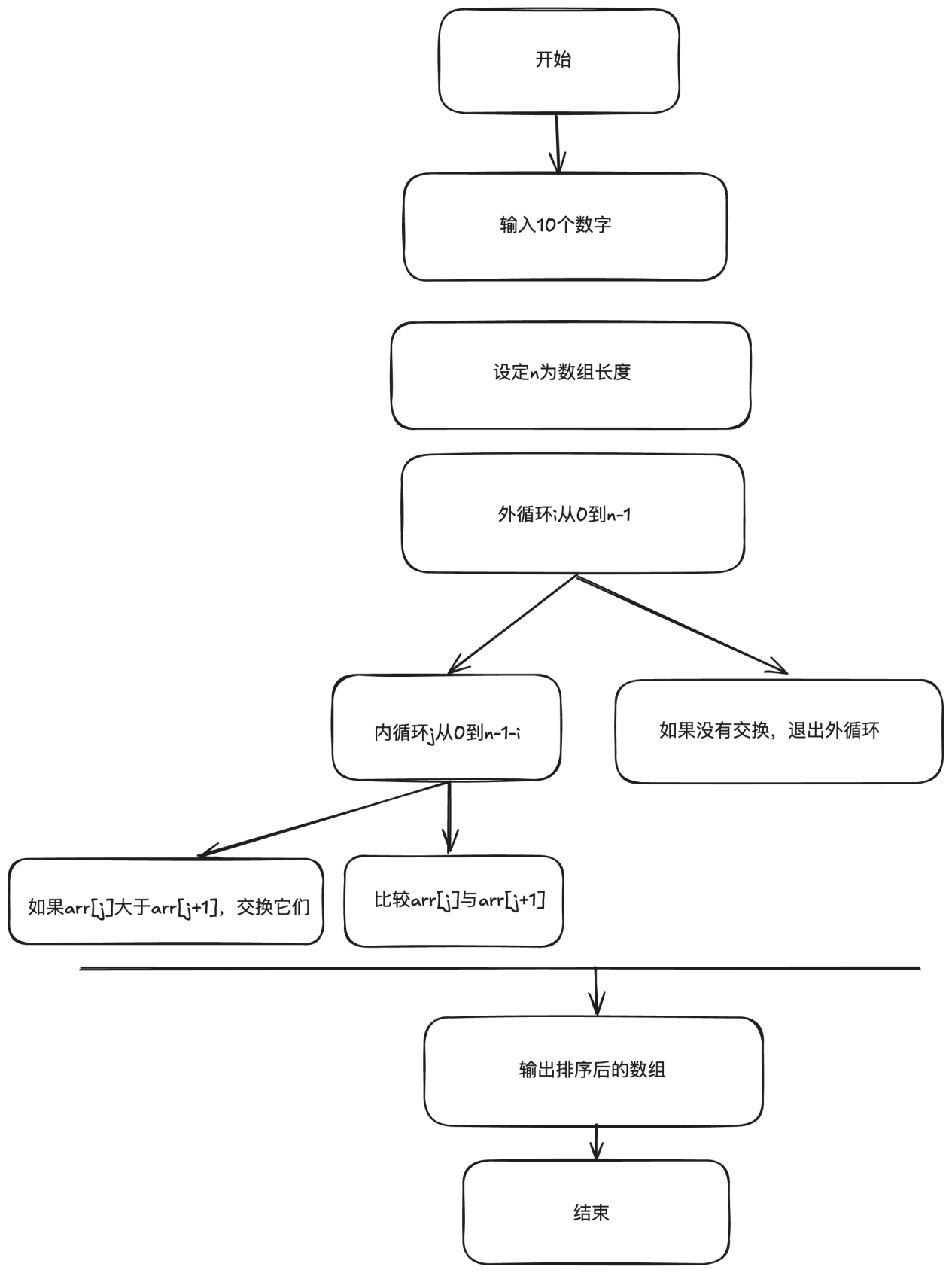
**代码运行截图：**



**2.题目：**

|  |
| --- |
| 冒泡排序编程：  执行过程 ：（1）比较相邻的元素，如果前一个比后一个大，就交换。（2）两两比较，直到最后一对。经过第一轮排序，最大的数在最后。（3）每次对剩下的元素重复以上步骤，直到没有任何一对数字需要比较。 |

**思维导图：**



**完整代码：**

|  |
| --- |
| JavaScript import java.util.Scanner;  public class BubbleSort {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.in);  double[] numbers = new double[10];   System.out.println("请输入10个double型数字：");  for (int i = 0; i < 10; i++) {  numbers[i] = scanner.nextDouble();  }   bubbleSort(numbers);   System.out.println("排序后的结果：");  for (double num : numbers) {  System.out.print(num + " ");  }  }   public static void bubbleSort(double[] arr) {  int n = arr.length;  boolean swapped;  for (int i = 0; i < n - 1; i++) {  swapped = false;  for (int j = 0; j < n - 1 - i; j++) {  if (arr[j] > arr[j + 1]) {  // 交换  double temp = arr[j];  arr[j] = arr[j + 1];  arr[j + 1] = temp;  swapped = true;  }  }  // 如果没有交换，数组已经有序  if (!swapped) break;  }  } } |

**代码运行截图：**

