18\_基本情報アルゴリズム④

**課題1**

数値配列に対して選択ソートを使用し、昇順に並び替えて表示するプログラムを作成しなさい。

並び替えの途中経過も表示すること。

◆選択ソートの流れ



選択ソートはこのように単純でアルゴリズムが容易だが、全ての要素と比較する為、

時間計算量は最悪パターンでも最良パターンでもいずれも*О(n²)*となる。

ファイル名：Kad18\_1.java

＜実行結果＞

\*\*\*\*\*\*\*\*\* 元データ \*\*\*\*\*\*\*\*\*

2, 5, 8, 1, 4, 7, 3, 6, 9, 0,

\*\*\*\*\*\* 選択ソート開始 \*\*\*\*\*\*

0, 5, 8, 1, 4, 7, 3, 6, 9, 2,

0, 1, 8, 5, 4, 7, 3, 6, 9, 2,

0, 1, 2, 5, 4, 7, 3, 6, 9, 8,

0, 1, 2, 3, 4, 7, 5, 6, 9, 8,

0, 1, 2, 3, 4, 7, 5, 6, 9, 8,

0, 1, 2, 3, 4, 5, 7, 6, 9, 8,

0, 1, 2, 3, 4, 5, 6, 7, 9, 8,

0, 1, 2, 3, 4, 5, 6, 7, 9, 8,

0, 1, 2, 3, 4, 5, 6, 7, 8, 9,

\*\*\*\*\*\* 選択ソート完了 \*\*\*\*\*\*

|  |  |  |  |
| --- | --- | --- | --- |
|  | 戻り値 | 名前 | 引数 |
| メソッド | void | selectSort | int[] data |

◆selectSortメソッドの処理

引数で受け取った配列を選択ソートを使用して昇順に並び替える。

|  |  |  |  |
| --- | --- | --- | --- |
|  | 戻り値 | 名前 | 引数 |
| メソッド | void | printArray | int[] data |

◆printArrayメソッドの処理

引数で受け取った配列の値を表示する。

for(int d : data){

System.out.print(d + ", ");

}

System.out.println(""); //改行

◆mainメソッドの処理

①以下の数値配列を宣言する。

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

②数値配列を引数にprintArrayメソッドを呼び出し、ソート前の配列の値を表示する。

System.out.println("\*\*\*\*\*\*\*\*\* 元データ \*\*\*\*\*\*\*\*\*");

printArray(～省略～);

③数値配列を引数にselectSortメソッドを呼び出し、ソートする。

System.out.println("\*\*\*\*\*\* 選択ソート開始 \*\*\*\*\*\*");

selectSort(～省略～);

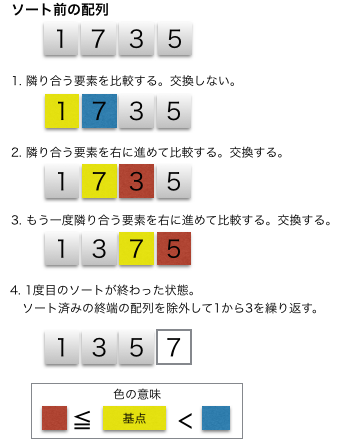
System.out.println("\*\*\*\*\*\* 選択ソート完了 \*\*\*\*\*\*");

**課題2**

数値配列に対してバブルソートを使用し、昇順に並び替えて表示するプログラムを作成しなさい。

並び替えの途中経過も表示すること。

◆バブルソートの流れ



バブルソートは選択ソートと同じく単純でアルゴリズムを記述しやすいが、選択ソートと比べると

最良パターンの場合は比較する回数が配列の個数と同じ回数で済むため、時間計算量は*О(n)*となる。最悪パターンの場合は選択ソートと変わらず*О(n²)*となる。

ファイル名：Kad18\_2.java

＜実行結果＞

\*\*\*\*\*\*\*\*\* 元データ \*\*\*\*\*\*\*\*\*

2, 5, 8, 1, 4, 7, 3, 6, 9, 0,

\*\*\*\*\* バブルソート開始 \*\*\*\*\*

2, 5, 1, 4, 7, 3, 6, 8, 0, 9,

2, 1, 4, 5, 3, 6, 7, 0, 8, 9,

1, 2, 4, 3, 5, 6, 0, 7, 8, 9,

1, 2, 3, 4, 5, 0, 6, 7, 8, 9,

1, 2, 3, 4, 0, 5, 6, 7, 8, 9,

1, 2, 3, 0, 4, 5, 6, 7, 8, 9,

1, 2, 0, 3, 4, 5, 6, 7, 8, 9,

1, 0, 2, 3, 4, 5, 6, 7, 8, 9,

0, 1, 2, 3, 4, 5, 6, 7, 8, 9,

\*\*\*\*\* バブルソート完了 \*\*\*\*\*

|  |  |  |  |
| --- | --- | --- | --- |
|  | 戻り値 | 名前 | 引数 |
| メソッド | void | bubbleSort | int[] data |

◆bubbleSortメソッドの処理

引数で受け取った配列をバブルソートを使用して昇順に並び替える。

|  |  |  |  |
| --- | --- | --- | --- |
|  | 戻り値 | 名前 | 引数 |
| メソッド | void | printArray | int[] data |

◆printArrayメソッドの処理

引数で受け取った配列の値を表示する（課題1で作成した同じメソッドをコピーして流用する）

◆mainメソッドの処理

①以下の数値配列を宣言する。

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

②数値配列を引数にprintArrayメソッドを呼び出し、ソート前の配列の値を表示する。

System.out.println("\*\*\*\*\*\*\*\*\* 元データ \*\*\*\*\*\*\*\*\*");

printArray(～省略～);

③数値配列を引数にbubbleSortメソッドを呼び出し、ソートする。

System.out.println("\*\*\*\*\* バブルソート開始 \*\*\*\*\*");

bubbleSort(～省略～);

System.out.println("\*\*\*\*\* バブルソート完了 \*\*\*\*\*");