**selection sort:**

**selection sort : هي خوارزمية لترتيب المصفوفة بطريقه سهله وبسيطة بطريقة معينة .**

**طريقة عمل خوارزمية الترتيب :**

1. **ابحث عن أصغر عنصر بالمصفوفة و أبدله بأول عنصر بالمصفوفة , والمبادلة تعني أن تضع كل عنصر مكان عنصر أخر , و إذا وجدت أن أصغر عنصر بالمصفوفة في أول خانة فلا داعي لإجراء تغيير فهو في مكانه الصحيح .**
2. **بحث عن أصغر عنصر في المصفوفة ناقصاً الجزء المُرتب.**
3. **لتوضيح معنى الجزء المرتب ، عندما وضعت في الخطوة الأولى أصغر عنصر في الخانة الأولى فأنت تعلم أن هذا العنصر قد أخذ مكانه الصحيح و هنا نسمي الخانة الأولى بالجزء المرتب ، و لإيجاد ثاني أصغر عنصر بالمصفوفة ستبحث في عناصر المصفوفة غير الجزء المرتب .**
4. **ثم نعير الخطوات الى ان نصل الى اخر عنصر بالمصفوفة .**

**مثال :**

**مصفوفة مكونه من 5 عناصر غير مرتبه :**

**(3.2.6.9.8)**

1. **نبحث عن اصغر عنصر موجود بالمصفوفة = 2**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 8 | 9 | 6 | 2 | 3 |

1. **نضع العنصر 2 ف اول خانه بالمصفوفة ونأخذ العنصر الموجود بالخانة الأولى ونضعه في مكان العنصر2 السابق**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 8 | 9 | 6 | 3 | 2 |

1. **نبحث عن اصغر عنصر موجود من دون الجزء المرتب = 3**

**وجدنا ان العنصر ف مكانه الصحيح لا نحتاج تغيير موقعه**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 8 | 9 | 6 | 3 | 2 |

1. **نبحث عن اصغر عنصر موجود من دون الجزء المرتب = 6**

**وجدنا ان العنصر ف مكانه الصحيح لا نحتاج تغيير موقعه**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 8 | 9 | 6 | 3 | 2 |

1. **نبحث عن اصغر عنصر موجود من دون الجزء المرتب = 8**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 8 | 9 | 6 | 3 | 2 |

1. **نضع العنصر =8 ف أول خانه بعد الجزء المرتب لنضمها مع الجزء المرتب .**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 9 | 8 | 6 | 3 | 2 |

**كما تلاحظون الأن ان المصفوفة ترتبت بالشكل الصحيح**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 9 | 8 | 6 | 3 | 2 |

**خوارزمية الترتيب بلغة الجافا :**

public static void selectionSort(int[] arr)

{

for (int i = 0; i < arr.length - 1; i++)

{

int index = i;

for (int j = i + 1; j < arr.length; j++)

{

if (arr[j] < arr[index])

{

index = j ;

}

}

int smallerNumber = arr[index];

arr[index] = arr[i];

arr[i] = smallerNumber;

}

}