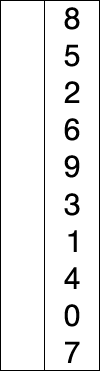
**Selection sort**

في البداية تستخدم خوارزميات الفرز (الترتيب)في إعادة فرز مجموعة من العناصر بالاعتماد على عامل مقارنة محدد.

وإحدى هذه الخوارزميات تعرف بالفرز بالتحديد حيث تقوم بترتيب العناصر إما تصاعديا أو تنازليا

مبدأ العمل :

بافتراض أن الترتيب سيكون تصاعدي تقوم المصفوفة بالبحث عن أصغر عنصر في المصفوفة وتضعه في بداية المصفوفة نلاحظ أن المصفوفة الأصلية تنقسم إلى مصفوفتين فرعتين مصفوفة تحوي العناصر المرتبة والتي تكون فارغة في البداية

ومصفوفة تحوي العناصر غير المرتبة والتي تحوي جميع عناصر المصفوفة الأصلية قبل البدء بعملية الترتيب

في كل دورة يؤخذ العنصر ذو القيمة الأصغر ويوضع في مكانه الصحيح في المصفوفة الفرعية المرتبة وهكذا حتى تصبح المصفوفة الفرعية المرتبة تحوي جميع العناصر كل منها في موضعه الصحيح.

**التعقيد الزمني :**

في الحالة الأسوء يكون التعقيد الزمني o(n2)

أما في الحالة الأفضل التي تكون فيها عناصر المصفوفة مرتبة بالفعل يكون التعقيد الزمني o(n)

تطبيق عملي :

لدينا مصفوفة حجمها N تتكون من أعداد صحيحة غير مرتبة نحتاج لترتيب هذه المصفوفة ترتيبا تصاعديا .

// C++ program for implementation of selection sort

#include <bits/stdc++.h>

using namespace std;

void swap(int \*xp, int \*yp)

{

    int temp = \*xp;

    \*xp = \*yp;

    \*yp = temp;

}

void selectionSort(int arr[], int n)

{

    int i, j, min\_idx;

    // One by one move boundary of unsorted subarray

    for (i = 0; i < n-1; i++)

    {

        // Find the minimum element in unsorted array

        min\_idx = i;

        for (j = i+1; j < n; j++)

        if (arr[j] < arr[min\_idx])

            min\_idx = j;

        // Swap the found minimum element with the first element

        swap(&arr[min\_idx], &arr[i]);

    }

}

/\* Function to print an array \*/

void printArray(int arr[], int size)

{

    int i;

    for (i=0; i < size; i++)

        cout << arr[i] << " ";

    cout << endl;

}

// Driver program to test above functions

int main()

{

    int arr[] = {64, 25, 12, 22, 11};

    int n = sizeof(arr)/sizeof(arr[0]);

    selectionSort(arr, n);

    cout << "Sorted array: \n";

    printArray(arr, n);

    return 0;

}

// This is code is contributed by rathbhupendra