PDP
IT–ACADEMY

Tartiblash algoritmlari

Reja:

- Tartiblash algoritmlari haqida
- SELECTION SORT
- INSERTION SORT
- BUBBLE SORT
- Amaliy mashqlar

PDP
IT-ACADEMY

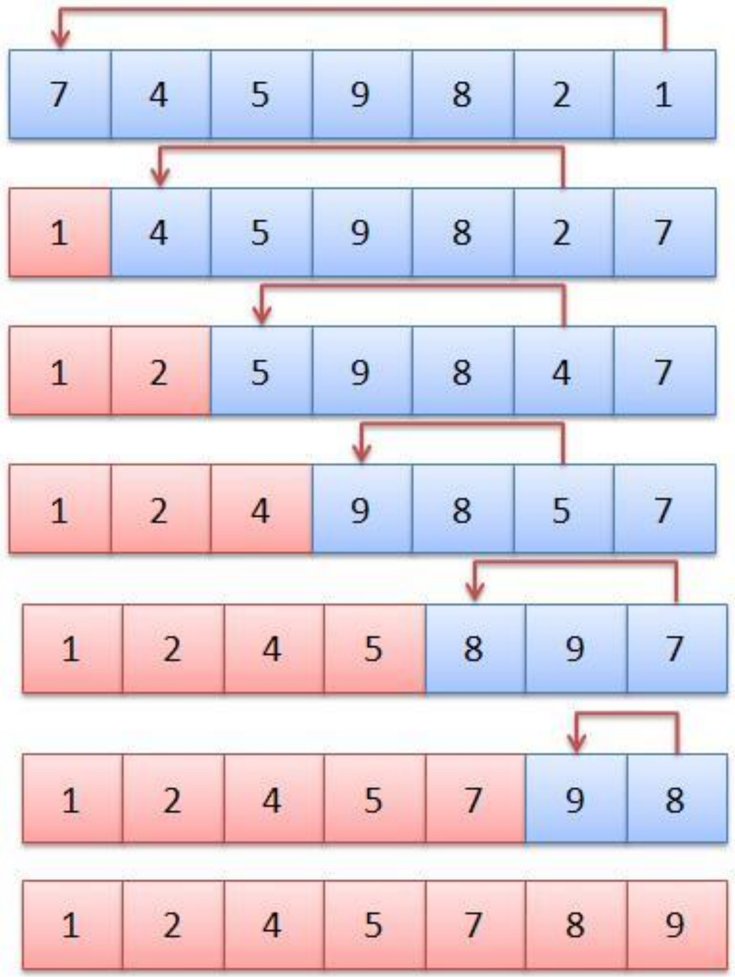
Tartiblash algoritmlari

PDP
IT-ACADEMY

PDP
IT–ACADEMY

SELECTION SORT

PDP
IT–ACADEMY



PDP
IT-ACADEMY

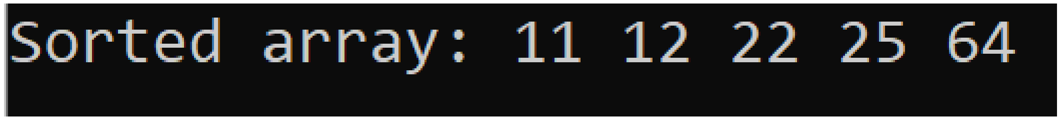
```
void selectionSort(int arr[], int n)
{
    int i, j, min_idx;

    // 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]);
    }
}
```

PDP IT-ACADEMY

```
/* Function to print an array */
void printArray(int arr[], int size)
{
    int i;
    for (i=0; i < size; i++)
        cout << arr[i] << " ";
    cout << endl;
}
```

PDP
IT-ACADEMY

```cpp
int main()
{
    int arr[] = {64, 25, 12, 22, 11};
    int n = sizeof(arr)/sizeof(arr[0]);

    selectionSort(arr, n);

    cout << "Sorted array: ";
    printArray(arr, n);

    return 0;
}
```

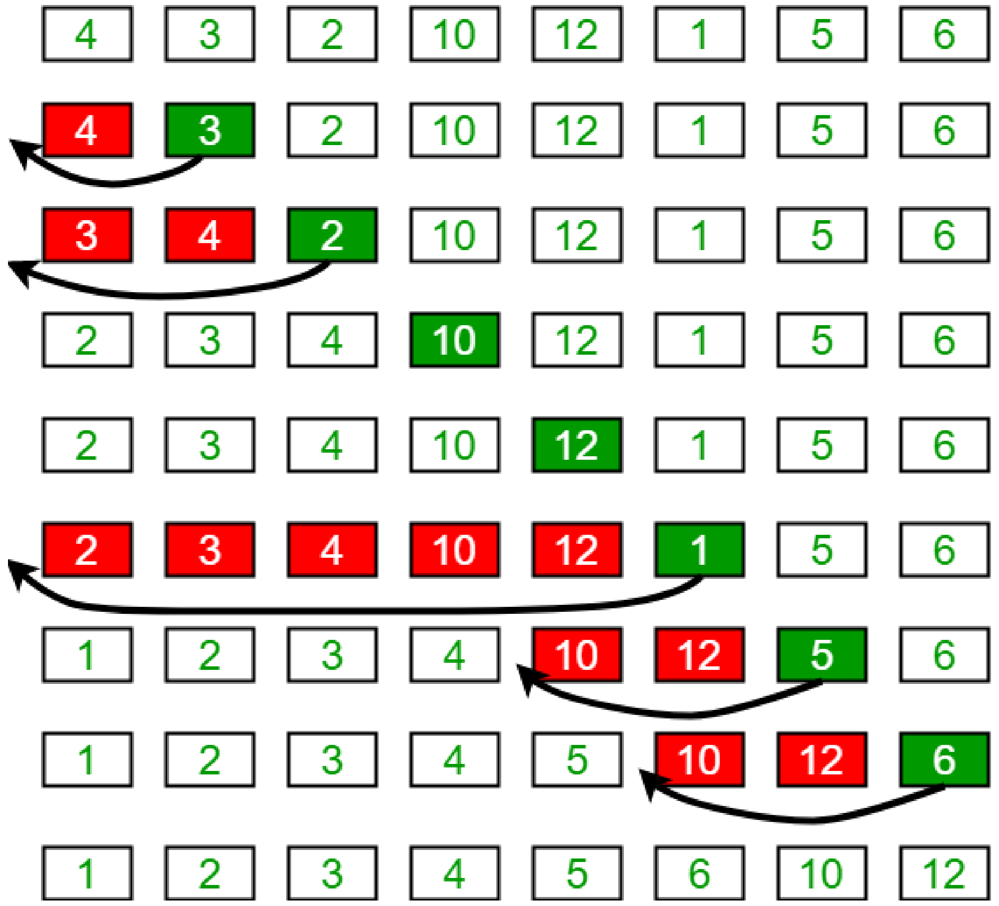


```
Sorted array: 11 12 22 25 64
```

INSERTION SORT

PDP IT-ACADEMY

Insertion Sort Execution Example

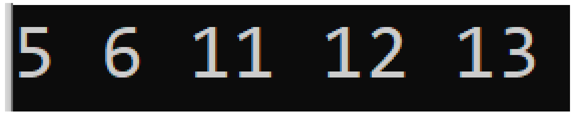


```
/* Function to sort an array using insertion sort*/
void insertionSort(int arr[], int n)
{
    int i, key, j;
    for (i = 1; i < n; i++)
    {
        key = arr[i];
        j = i - 1;


        /* Move elements of arr[0..i-1], that are
        greater than key, to one position ahead
        of their curent position */
        while (j >= 0 && arr[j] > key)
        {
            arr[j + 1] = arr[j];
            j = j - 1;
        }
        arr[j + 1] = key;
    }
}
```

PDP
IT-ACADEMY

```
/* Function to print an array */

void printArray(int arr[], int size)
{
    int i;
    for (i=0; i < size; i++)
        cout << arr[i] << " ";
    cout << endl;
}
```

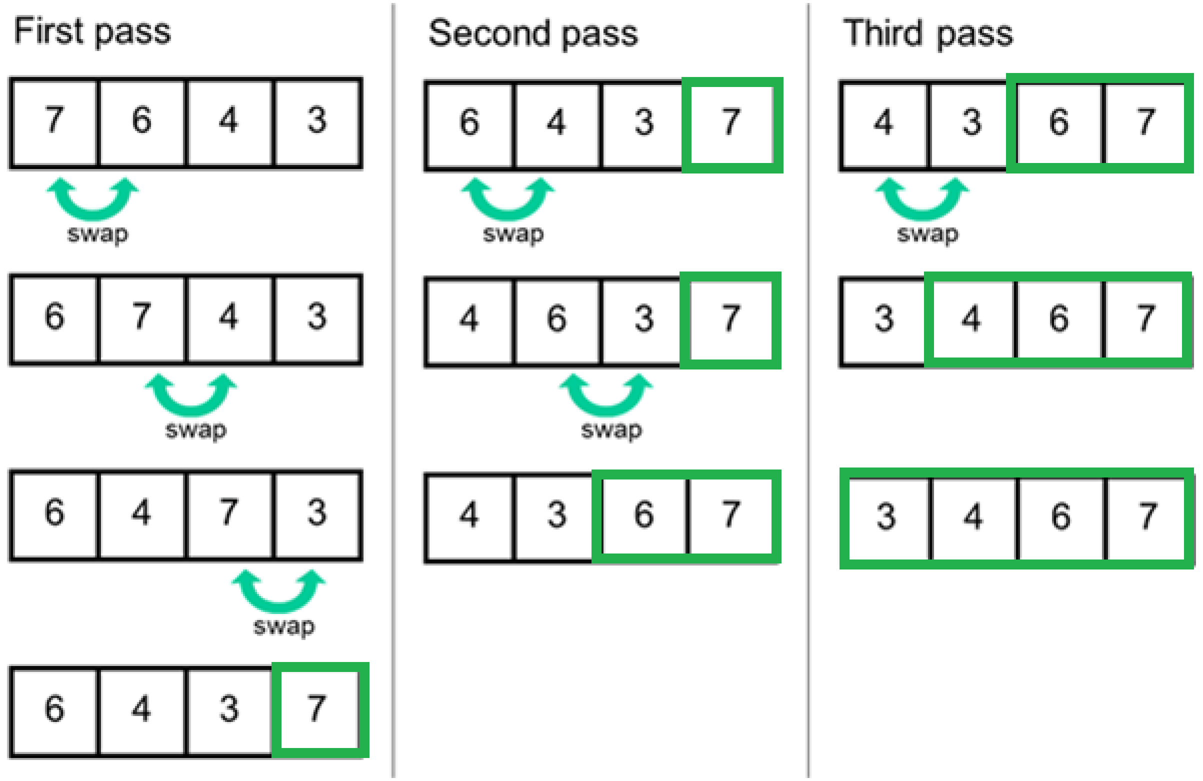
```
int main()
{
    int arr[] = { 12, 11, 13, 5, 6 };
    int n = sizeof(arr) / sizeof(arr[0]);

    insertionSort(arr, n);

    printArray(arr, n);


    return 0;
}
```



```
5 6 11 12 13
```

PDP
IT-ACADEMY

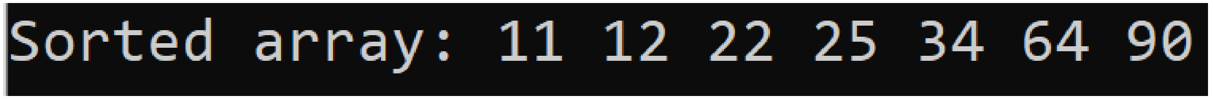
BUBBLE SORT


First pass
7 6 4 3
swap
6 7 4 3
swap
6 4 7 3
swap
6 4 3 7
Second pass
6 4 3 7
swap
4 6 3 7
swap
4 3 6 7
Third pass
4 3 6 7
swap
3 4 6 7
3 4 6 7

```
// A function to implement bubble sort

void bubbleSort(int arr[], int n)
{
    int i, j;
    for (i = 0; i < n-1; i++)

    // Last i elements are already in place

    for (j = 0; j < n-i-1; j++)
        if (arr[j] > arr[j+1])
            swap(arr[j], arr[j+1]);
}
```

```
/* Function to print an array */

void printArray(int arr[], int size)
{
    int i;
    for (i=0; i < size; i++)
        cout << arr[i] << " ";
    cout << endl;
}
```

```cpp
int main()
{
    int arr[] = {64, 34, 25, 12, 22, 11, 90};
    int n = sizeof(arr)/sizeof(arr[0]);

    bubbleSort(arr, n);

    cout<<"Sorted array: ";
    printArray(arr, n);

    return 0;
}
```



```
Sorted array: 11 12 22 25 34 64 90
```

Amaliy mashqlar

PDP
IT-ACADEMY

Butun sonlardan iborat 7 ta elementli massiv berilgan. Massiv elementlari tasodifiy sonlar generatori yordamida aniqlanadi.

Massiv elementlarini o’sish tartibida tartiblash uchun SELECTION SORT asosida har bir qadamni alohida-alohida Excel dasturida yozing.

Butun sonlardan iborat 7 ta elementli massiv berilgan. Massiv elementlari tasodifiy sonlar generatori yordamida aniqlanadi.

Massiv elementlarini o’sish tartibida tartiblash uchun INSERTION SORT asosida har bir qadamni alohida-alohida Excel dasturida yozing.

Butun sonlardan iborat 7 ta elementli massiv berilgan. Massiv elementlari tasodifiy sonlar generatori yordamida aniqlanadi.

Massiv elementlarini o’sish tartibida tartiblash uchun BUBBLE SORT asosida har bir qadamni alohida-alohida Excel dasturida yozing.

PDP
IT-ACADEMY

E`tiboringiz uchun rahmat!