Write a program to reverse an array or string

Iterative way:

1) Initialize start and end indexes.

start = 0, end = n-1

2) In a loop, swap arr[start] with arr[end] and change start and end as follows.

start = start +1; end = end – 1

C

```
// Iterative C program to reverse an array
#include<stdio.h>

/* Function to reverse arr[] from start to end*/
void rvereseArray(int arr[], int start, int end)
{
    int temp;
    while (start < end)
    {
        temp = arr[start];
        arr[start] = arr[end];
        arr[end] = temp;
        start++;
        end--;
    }
}

/* Utility that prints out an array on a line */
void printArray(int arr[], int size)
{
  int i;
  for (i=0; i < size; i++)
    printf("%d ", arr[i]);

  printf("\n");
}

/* Driver function to test above functions */
int main()
{
    int arr[] = {1, 2, 3, 4, 5, 6};
    printArray(arr, 6);
    rvereseArray(arr, 0, 5);
    printf("Reversed array is \n");
    printArray(arr, 6);
```

```
    return 0;
}
```

Run on IDE

Java

```
// Java program to reverse an array
import java.io.*;

class ReverseArray {

    /* Function to reverse arr[] from start to end*/
    static void rvereseArray(int arr[], int start, int end)
    {
        int temp;
        if (start >= end)
            return;
        temp = arr[start];
        arr[start] = arr[end];
        arr[end] = temp;
        rvereseArray(arr, start+1, end-1);
    }

    /* Utility that prints out an array on a line */
    static void printArray(int arr[], int size)
    {
        int i;
        for (i=0; i < size; i++)
            System.out.print(arr[i] + " ");
        System.out.println("");
    }

    /*Driver function to check for above functions*/
    public static void main (String[] args) {
        int arr[] = {1, 2, 3, 4, 5, 6};
        printArray(arr, 6);
        rvereseArray(arr, 0, 5);
        System.out.println("Reversed array is ");
        printArray(arr, 6);
    }
}
/*This code is contributed by Devesh Agrawal*/
```

Run on IDE

Output:

```
1 2 3 4 5 6
Reversed array is
6 5 4 3 2 1
```

Time Complexity: O(n)

Recursive Way:

1) Initialize start and end indexes

start = 0, end = n-1

2) Swap arr[start] with arr[end]

3) Recursively call reverse for rest of the array.

C

```c
// Recursive C program to reverse an array
#include <stdio.h>

/* Function to reverse arr[] from start to end*/
void rvereseArray(int arr[], int start, int end)
{
   int temp;
   if (start >= end)
     return;
   temp = arr[start];
   arr[start] = arr[end];
   arr[end] = temp;
   rvereseArray(arr, start+1, end-1);
}

/* Utility that prints out an array on a line */
void printArray(int arr[], int size)
{
  int i;
  for (i=0; i < size; i++)
    printf("%d ", arr[i]);

  printf("\n");
}

/* Driver function to test above functions */
int main()
{
    int arr[] = {1, 2, 3, 4, 5};
    printArray(arr, 5);
    rvereseArray(arr, 0, 4);
    printf("Reversed array is \n");
    printArray(arr, 5);
    return 0;
}
```

Run on IDE

Java

```java
// Recursive Java Program to reverse an array
import java.io.*;

class ReverseArray {

    /* Function to reverse arr[] from start to end*/
    static void rvereseArray(int arr[], int start, int end)
    {
        int temp;
```

```
        if (start >= end)
            return;
        temp = arr[start];
        arr[start] = arr[end];
        arr[end] = temp;
        rvereseArray(arr, start+1, end-1);
    }

    /* Utility that prints out an array on a line */
    static void printArray(int arr[], int size)
    {
        for (int i=0; i < size; i++)
            System.out.print(arr[i] + " ");
        System.out.println("");
    }

    /*Driver function to check for above functions*/
    public static void main (String[] args) {
        int arr[] = {1, 2, 3, 4, 5, 6};
        printArray(arr, 6);
        rvereseArray(arr, 0, 5);
        System.out.println("Reversed array is ");
        printArray(arr, 6);
    }
}
/*This article is contributed by Devesh Agrawal*/
```

Run on IDE

Output:

```
1 2 3 4 5 6
Reversed array is
6 5 4 3 2 1
```

Time Complexity: O(n)

Please write comments if you find any bug in the above programs or other ways to solve the same problem.

41 Comments Category: Arrays Tags: array

Related Posts:

- Longest Span with same Sum in two Binary arrays
- Count Inversions of size three in a give array
- Find the subarray with least average
- Count triplets with sum smaller than a given value
- Find zeroes to be flipped so that number of consecutive 1's is maximized
- Reorder an array according to given indexes
- Find maximum value of Sum(i*arr[i]) with only rotations on given array allowed
- Find maximum average subarray of k length

(Login to Rate and Mark)

1.2 Average Difficulty : **1.2/5.0**
Based on **17** vote(s)

Add to TODO List

Mark as DONE

Writing code in comment? Please use code.geeksforgeeks.org, generate link and share the link here.