

# Apex College

## BCIS Program

Affiliated to Pokhara University



### Data Structure & Algorithms

#### Lab Report

17  
Radix Sort Algo

Date: 22-07-2022

#### Submitted by:

Ishwor Shrestha

Roll no.: 2018-BCIS-414

#### Submitted to:

Pravakar Ghimire, &

Anmol Shrestha

Apex College

## #Lab 17 Objective

- To understand the implementation of radix sort algorithm.

## #Introduction

Radix sort is a non-comparative sorting algorithm that is used to sort the data in lexicographical (dictionary) order. It uses counting sort as a subroutine, to sort an array of integer digit by digit and array of string characters by character. It is a stable sorting method that uses a counting sort as subroutine.

## #Source Code

---

```
#include <stdio.h>
```

```
int getMax (int arr [], int n) {
```

```
    int mx = arr [0];
```

```
    int i;
```

```
    for (i = 1; i < n; i++)
```

```
        if (arr [i] > mx)
```

```
            mx = arr [i];
```

```

        return mx;
    }

void countSort (int arr[], int n, int exp) {
    int output [n];
    int i, count [10] = {0};

    for (i = 0; i < n; i++)
        count [(arr[i] / exp) % 10] ++;

    for (i = 1; i < 10; i++)
        count[i] += count[i-1];

    for (i = n-1; i >= 0; i--) {
        output [count [(arr[i] / exp) % 10] - 1] = arr[i];
        count [(arr[i] / exp) % 10] --;
    }

    for (i = 0; i < n; i++)
        arr[i] = output[i];
}

```

```

void radixSort (int arr[], int n) {
    int m = getMax (arr, n);
    int exp;

    for (exp = 1; m/exp > 0; exp *= 10)
        countSort (arr, n, exp);
}

```

```

void print (int arr[], int n) {
    int i;
    for (i = 0; i < n; i++)
        printf ("%d ", arr[i]);
}

```

```
int main() {
```

```
int arr[] = { 170, 45, 75, 90, 802, 29, 2, 66 };
```

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

```
radix sort(arr, n);
```

```
print(arr, n);
```

```
return 0;
```

```
}
```

# Activities

We performed radix sort operation.

# Conclusion.

I learned about radix sort algorithm