ASSIGNMENT 3

Radix Sort (Base 10)

CSA0303 – DATA STRUCTURES FOR Problem Solving

NAME: KAVYA SHRI G

REG.NO: 192421052

AIM: To sort a list of integers using Radix Sort with stable sorting and digit-based bucketing.

ALGORITHM:

- 1. Find the maximum number to determine the number of digits.
- 2. Perform counting sort for each digit from least significant to most significant.
- 3. Use counting sort to sort the array based on the current digit.
- 4. Store the result in a temporary output array and copy it back to the original array.
- 5. Repeat the process for all digit places.
- 6. Print the array after each pass.

CODE:

```
#include <stdio.h>
#include <stdlib.h>
int getMax(int arr[], int n) {
  int max = arr[0];
  for (int i = 1; i < n; i++)
    if (arr[i] > max)
      max = arr[i];
  return max;
```

```
}
void countingSort(int arr[], int n, int exp) {
  int output[n];
  int count[10] = \{0\};
  for (int i = 0; i < n; i++)
    count[(arr[i] / exp) % 10]++;
  for (int i = 1; i < 10; i++)
    count[i] += count[i - 1];
  for (int i = n - 1; i >= 0; i--) {
    output[count[(arr[i] / exp) % 10] - 1] = arr[i];
    count[(arr[i] / exp) % 10]--;
  }
  for (int i = 0; i < n; i++)
     arr[i] = output[i];
}
void radixSort(int arr[], int n) {
  int max = getMax(arr, n);
  for (int exp = 1; max / exp > 0; exp *= 10)
    countingSort(arr, n, exp);
}
int main() {
  int arr[] = {17032, 45234, 80234, 24, 2, 66};
  int n = sizeof(arr) / sizeof(arr[0]);
  radixSort(arr, n);
  printf("Sorted array:\n");
  for (int i = 0; i < n; i++)
```

```
printf("%d ", arr[i]);
printf("\n");
return 0;
}
```

OUTPUT

```
Output

Sorted array:
2 24 66 17032 45234 80234

=== Code Execution Successful ===
```