SEM 3\Exp9\Bucket_Sort.c

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

#define BUCKET_SIZE 10

typedef struct Bucket
{
    int count;
    int *values;
} Bucket;

void bucketSort(int array[], int n);
void insertionSort(int array[], int n);

void bucketSort(int array[], int n)
{
    Bucket buckets[BUCKET_SIZE];
    for (int i = 0; i < BUCKET_SIZE; i++)
    {
        buckets[i].count = 0;
        buckets[i].values = (int *)malloc(n * sizeof(int));
    }

    for (int i = 0; i < n; i++)
    {
        int bucketIndex = array[i] / BUCKET_SIZE;
        buckets[bucketIndex].values[buckets[bucketIndex].count++] = array[i];
    }

    for (int i = 0; i < BUCKET_SIZE; i++)
    {
        if (buckets[i].count > 0)
        {
            insertionSort(buckets[i].values, buckets[i].count);
        }
    }

    int index = 0;
    for (int i = 0; i < BUCKET_SIZE; i++)
    {
        for (int j = 0; j < buckets[i].count; j++)
        {
            array[index++] = buckets[i].values[j];
        }
        free(buckets[i].values);
    }
}

void insertionSort(int array[], int n)
{
    for (int i = 1; i < n; i++)
    {
        int key = array[i];
        int j = i - 1;
        while (j >= 0 && array[j] > key)
        {
            array[j + 1] = array[j];
            j--;
        }
        array[j + 1] = key;
    }
}

```