File - C:\Users\Administrator\CLionProjects\Problem1\problem1 code.cpp

```cpp
#include<iostream>

using namespace std;


int maximum(int array[], int arraySize) {
    int max = 0;
    for (int i = 0; i < arraySize; i++)
        if (array[i] > max)
            max = array[i];
    return max;
}

void sort(int array[], int arraySize, int exp)
{
    int output[arraySize];
    int bucket[10] = {0};
    for (int i = 0; i < arraySize; i++)
        bucket[ (array[i]/exp)%10 ]++;
    for (int i = 0; i < 9; i++)
        bucket[i+1] = bucket[i] + bucket[i+1];
    for (int i = 0; i <arraySize; i++)
    {
        output[bucket[(array[i]/exp)%10]-1]=array[i];
        bucket[ (array[i]/exp)%10 ]--;
    }
    for (int i = 0; i < arraySize; i++)
        array[i] = output[i];
}

void Problem1Sort(int array[], int arraySize){
    int M = maximum(array, arraySize);
    for (int exp = 1; M/exp > 0; exp = exp*10)
        sort(array, arraySize, exp);
}


void print(int array[], int arraySize) {
    for (int a = 0; a < arraySize; a++) {
        cout << array[a] <<endl;
```

Get the maximum for calculating how large exponential number (10^n) is required.

10 buckets to store numbers each time

calculate the position of numbers.

build and output result.

use sort function

print function

File - C:\Users\Administrator\CLionProjects\Problem1\problem1 code.cpp

```
41      }
42  }
43
44  int main() {
45      int arraySize;
46      cout << "enter array size: ";
47      cin >> arraySize;
48      int h;
49      h = arraySize;
50      int array[h] = {0};
51      for (int i = 0; i < arraySize; i++) {
52          cout << "enter " << i + 1 << " number:";
53          cin >> array[i];
54      }
55      Problem1Sort(array, arraySize);
56      print(array, arraySize);
57      return 0;
58  }
59
```

main function