```
1 // Radix Sort in C Programming
 3 #include <stdio.h>
 5 int getMax(int array[], int n)
 6 {
 7
       int max = array[0];
 8
       for (int i = 1; i < n; i++)</pre>
 9
           if (array[i] > max)
10
                max = array[i];
11
       return max;
12 }
13 void countingSort(int array[], int size, int place)
14 {
15
       int output[size + 1];
16
       int max = (array[0] / place) % 10;
17
18
       for (int i = 1; i < size; i++)</pre>
19
20
           if (((array[i] / place) % 10) > max)
21
                max = array[i];
22
       }
23
       int count[max + 1];
24
25
       for (int i = 0; i < max; ++i)</pre>
26
           count[i] = 0;
27
28
       for (int i = 0; i < size; i++)</pre>
29
           count[(array[i] / place) % 10]++;
30
31
       for (int i = 1; i < 10; i++)
32
           count[i] += count[i - 1];
33
34
       for (int i = size - 1; i >= 0; i--)
35
36
           output[count[(array[i] / place) % 10] - 1] = array[i];
37
           count[(array[i] / place) % 10]--;
38
       }
39
40
       for (int i = 0; i < size; i++)</pre>
41
           array[i] = output[i];
42 }
43 void radixsort(int array[], int size)
44 {
45
       int max = getMax(array, size);
46
47
       for (int place = 1; max / place > 0; place *= 10)
48
           countingSort(array, size, place);
49 }
50 void printArray(int array[], int size)
51 {
52
       for (int i = 0; i < size; ++i)</pre>
53
       {
54
           printf("%d ", array[i]);
55
56
       printf("\n");
57 }
58 int main()
59 {
       int array[] = {121, 432, 564, 23, 1, 45, 788};
60
61
       int n = sizeof(array) / sizeof(array[0]);
       radixsort(array, n);
62
63
       printArray(array, n);
```

F	-ile -	C:\U	sers\Andrea	D'Agg\CLionPro	ojects\radixSort\	main.c		
ſ	64	}			ojects\radixSort\			
		-						
1								