Algorithm Homework 2C

학번 / 이름: 2014310407 / 이 준 혁

1. 구현

가장 먼저, 입력을 받아서 저장할 배열이 필요했다. 각 숫자를 배열에다 한 자리씩 저장한 후, radix sort 를 실행시키려 했다. 한 행의 길이가 10 인, 20 행짜리 2 차원 정수형 배열을 만들었다. (최대 입력이 200 byte 이므로)

그리고 getchar 함수를 이용해, 입력이 '₩n', 즉 엔터가 아닐 때까지 입력을 받았다. 그리고 입력이 16 진수 숫자이면 그 값을 숫자로 바꿔서 배열에 저장했고, 공백이면서 총 입력 받은 숫자가 10 개가 되면 한 숫자의 입력을 다 받은 것 이므로, 다음 배열부터 처음부터 입력 받게 했다. 이 때 주어진 숫자가 16 진수 숫자인지 확인하기 위해 is_num 함수를, 입력 받은 아스키코드 문자 값을 정수로 저장하기 위해 to_num 함수를 사용했다.

이렇게 모두 입력을 받으면, 각 자릿수마다 Radix_Sort 함수를 통해 자릿수별로 정렬하게 했고, 또한 이차원 배열을 출력하는 printarr 함수를 통해 한 번 자릿수들이 정렬될 때마다 출력되도록 했다.

Radix_Sort 함수는 이차원 배열과 총 몇 개의 일 차원 배열이 있는 지, 또 몇 번 째 자릿수를 정렬 할 것인지를 입력으로 받아, Counting sort 를 이용하여 해당되는 자릿수에 따라 배열들을 정렬하는 함수이다.

우선 정렬하는 대상이 숫자가 아닌 배열이므로, 이를 정렬하기 위해 각 배열의 주소 값을 저장할 2 차원 포인터를 선언 후 메모리를 할당했다. 즉, 2 차원 배열이 아닌, 포인터들의 배열을 선언한 것이다. (출력 배열이라 지칭) 또한 counting sort 를 위해 추가적인 배열이 필요한데, 16 진수 값은 0 ~ 15 이므로 16 개 크기를 가지는 배열을 선언 후 메모리를 할당하며, 0 으로 초기화 시켰다. (추가 배열이라 지칭)

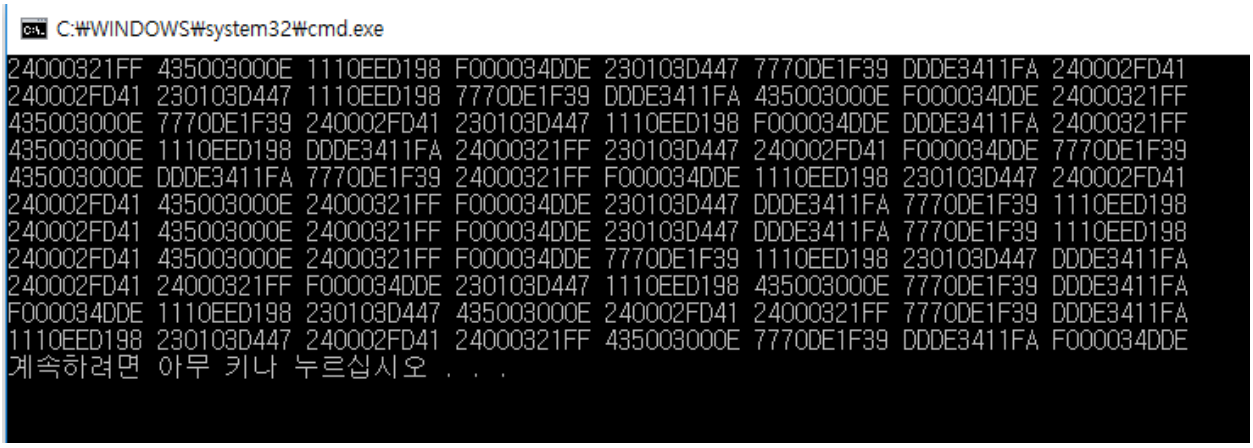
그리고 counting sort 와 마찬가지로, 입력 데이터의 주어진 자릿수 번째 값을 index 로 가지는 추가 배열의 값을 1 씩 증가 시킨 후, 이 배열의 값을 누적되게, 현재 배열의 값에 전 index 번째 값을 더했다.

그리고 입력 배열에서 i 번째 배열의 주어진 자릿수 번째 값에 해당되는 추가 배열의 값을 index 로 가지는 출력 배열에, i 번째 배열의 주소 값을 넣었다. 즉, 숫자가 하나의 배열로 이루어져 있으므로, 자릿수 하나하나를 옮기는 방식이 아닌, 배열의 주소 값을 정렬하는 방식으로 진행하였다.

그리고 해당되는 추가 배열의 값을 1 씩 감소시켰다. 이렇게 함수를 진행시키면, 출력 배열에는 주어진 수들의 자릿수 번째 값 순서대로 정렬되게 되고(Counting sort), 이를 다시 입력 배열에 대입함으로써 정렬을 완성시켰다.

이 과정을 모든 자릿수(10 개)에 대해 반복했고, 매 번 출력하게 했다. 모든 자릿수에 대해 시행했다면 값들이 제대로 정렬된 상태로 나오게 될 것이다.

2. 결과



첫번째 줄은 일의 자리 순으로, 두 번째 줄은 십의 자리 순으로, 정렬되어서 최종적으로는 오름차순으로 잘 정렬됨을 확인할 수 있다.