**Data Structure\_2071035 Lee Somin**

**Technical Report – radix\_sort.cpp**

*Theorical Explanation of Functions in ‘radix\_sort.cpp’*

**typedef struct data**

This structure contains the array of id and score.

**selection\_sort\_stable**

Inputs: data \*list, int n

Return: non

This function sorts the given data in stable manner. It executes the selection sort when the score of current element is smaller than the least score. To make the sort result stable, the function compares weather the element’s id is smaller than the least element’s id, when the score of the current element is equal to the least score. If the id is smaller than the original ‘least’ element’s id, the function executes SWAP operation.

**main**

In main function, the id and the score array is initialized. Memory is allocated to the 'data \*input' and the elements are set to the address of the initialized id and score. It prints out the original data, and executes the stable slection sort by calling 'slsection\_sort\_stable()'. Then, it prints out the sorted data.

(Result Screen Shot Continued)

**Result:**

The element with smaller id is infront of the element with bigger id, if their score are same. Therefore, the sorting is stable.

텍스트이(가) 표시된 사진

자동 생성된 설명