Assignment 12: The Second Weight

12.1 Universal Pointers

The type `void` represents an empty set, but the `void` pointer (`void *`) does universal pointers that can point to any data of any type. A generic function can be defined using `void` pointers. The following code shows an example of using `qsort`, the standard library function implementing the famous algorithm called Quicksort by Tony Hoare:

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

int cmp(const void *a, const void *b) {
    const int *x = a, *y = b;
    return *x - *y;
}

void print(int a[], int n) {
    for (int i = 0; i < n; i++)
        printf("%2d%s", a[i], (i == n - 1)? "\n": " ");
}

int main() {
    const int MAX = 10, LB = 10, UB = 99, WIDTH = UB-LB+1, size = MAX;
    int a[MAX];
    srand(time(NULL));
    for (int i = 0; i < MAX; ++i)
        a[i] = rand() % WIDTH + LB;
    print(a, size);
    qsort(a, size, sizeof *a, cmp);
    print(a, size);
    return 0;
}
```

The function `qsort` is declared in `<stdlib.h>`.

12.2 Programming Assignment 12: weight2.c

Write a program calculating the weights of data items and finding the second heavy data item. There two kinds of data items: character strings and integers. For an integer data n, the weight of n is $|n|$, the absolute value of n. For a string data s, the weight of s is the sum of the ASCII code. For example, the weight of -5 is 5 and that of `"Hello"` is 500. If the representation of a data item starts with a numeral or sign, it should be regarded as an integer item. Assume that there's no such case of invalid data items, say `+a` nor `12b`.

Your program should define and use a type `Data` to store either `int` or `char *` for character string. Define and use a function making a data item and calculating the weight of a data item:

```c
Data *read_data(char word[]);
int   weight(Data *data);
```

where the data items are passed using pointers.

Your program is to read from standard input. The input consists of two lines. The first line of the input contains n ($n > 1$), the number of data items. The second line of the input contains n data items separated by a space. The maximum length of the string representing a data item is 35. Your program should print to standard output. The output consists of two lines. The first line contains the data items whose weight is the second, separated by a space. The order of the data items should be the same in the input line. Printing a data

item, the string data should be printed as is but the integer data should be printed with a sign. The second line contains the weight of the data items. Assume that there no such input not containing the second heavy data item(s).

Additional requirements for bonus points

- Define and use the standard library function `qsort`.
- Preserve the original array using `qsort` since the Quicksort is not a stable sort algorithm.

Input	**Output**
5 Hello -701 Hola 500 ABC	Hello +500 500
7 lots of people are lost in 100	lots lost 450
5 A +100 A -100 A	A A A 65