

```

/*--- topscores.c ---*/
//Mike Hanling
//topscores.c

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

typedef struct {
    char* first;
    char* last;
    int tscore;
} player;

void selectionSort(player* data, int size);
int before(player a, player b);

int main() {
    //get filename
    printf("filename: ");
    char filename[128];
    scanf(" %s", filename);
    FILE* fin = fopen(filename, "r");

    //find out how many
    int total = 0;
    fscanf(fin, " %i", &total);

    //read in all data
    player* allp = calloc(total, sizeof(player));
    for (int i = 0; i < total; i++) {
        allp[i].first = calloc(128, sizeof(char));
        allp[i].last = calloc(128, sizeof(char));
        fscanf(fin, " %s %s %i", allp[i].first, allp[i].last, &allp[i].tscore);
    }

    //sort alphabetically first, high scores second
    selectionSort(allp, total);

    //print each score (already highest) for each name
    printf("%s %s %i\n", allp[0].first, allp[0].last, allp[0].tscore);
    int iprint = 0;
    for (int i = 1; i < total; i++) {
        if (strcmp(allp[i].first, allp[iprint].first) != 0 ||
            strcmp(allp[i].last, allp[iprint].last) != 0)
        {
            printf("%s %s %i\n", allp[i].first, allp[i].last, allp[i].tscore);
            iprint = i;
        }
    }

    return 0;
}

void selectionSort(player* data, int size) {
    for(int length = size; length > 1; --length) {
        // Find imax, the index of the largest
        int imax = 0;
        for(int i = 1; i < length; ++i) {
            if (before(data[imax], data[i])) {
                imax = i;
            }
        }
    }
}

```

```

// Swap data[imax] & the last element
player temp = data[imax];
data[imax] = data[length - 1];
data[length - 1] = temp;
    }
}

int before(player a, player b) {
    int samefirst = strcmp(a.first, b.first);
    int samelast = strcmp(a.last, b.last);
    if (samelast == 0 && samefirst == 0) {
        return a.tscore > b.tscore;
    }
    if (samelast == 0) {
        return samefirst < 0;
    }

    return samelast < 0;
}

```