CS 351 Programming Assignment 6

Due: December 4, 2014

In this assignment you are to write a utility library that contains two generic functions in C: a generic swap function and a generic quick sort function. There are three files in this assignment.

1. myutil.h

This file contains two function prototypes:

```
(1)  void swap(void *, void *, unsigned);
```

In this function, the first two arguments are the data to be swapped, and the third argument is amount of space (number of bytes) of each data to be swapped.

```
(2)  void quickSort(void *, unsigned, unsigned,
                    int (*) (void *, void *));
```

In this function, the first argument is the array to be sorted, its second argument is the number of elements in the array, its third argument is the number of bytes in each array element, and its fourth argument is pointer to a function, called the "comparison function", which is used to compare two elements in the array. The comparison function takes as arguments two pointers to elements of the array. It returns an int that is <, =, or > 0, depending on whether the value pointed to by first argument is considered to be <, =, or > the value pointed to by the second argument. Note that the quicksort function should call the generic swap function when it needs to swap two array elements.

2. myutil.c

This file contains implementation of two functions declared in myutil.h header file. You should use C Example 17 as a basis to write these two functions.

3. program6.c

You are to write a program that reads student information from the data file ‘‘**Assignment6.data**’’ which is available on Kodiak.

The data file is in the following format:

```
DANISE,AMY A            Biology        58
CHICOSKY,CHESTER L      Computer       66
BUSHA,TRACEY M          Accounting     83
CARTER,MARY L           History        86
...
```

where the student’s name is on columns 1 - 20 (at most 20 characters), student’s major is on columns 25 - 34 (at most 10 characters), and the student’s grade starts on columns 40. Assume the data file contains at most 25 students.

You should write three comparison functions. The first comparison function is used to sort the array in **descending** order by **grade.** The second comparison function is used to sort the array in **ascending** order by student’s **name.** The third comparison function is used to sort the array in **ascending** order according to student’s **major**, and for students in the same major, it sorts data in **descending** order by their **grade**.

Your main program should create 3 arrays of student structures, each of size 25. Then read data from the data file into all three arrays. Print any of the original arrays, then call your generic quick sort using the first comparison function to sort the first array, call your generic quick sort using the second comparison function to sort the second array, and call your generic quick sort using the third comparison function to sort the third array. Then print each sorted array.

Here is the code of quick sort and binary search in C. You need to convert them to **generic**.

```
void sort(int list[], int left, int right)
{
    int i, j;
    int pivot = list[(left + right)/2];

    i = left, j = right;

    while (i <= j)
    {
        while (list[i] < pivot)
            ++i;

        while (list[j] > pivot)
            --j;

        if (i <= j)
        {
            swap(&list[i], &list[j]);
            ++i;
            --j;
        }
    }

    if (left < j)
        sort(list, left, j);

    if (i < right)
        sort(list, i, right);
}
```

```
void quickSort(int list[], int size)
{
    sort(list, 0, size-1);
}
```