1. Both a C structure (a struct) and an array may store multiple data values, but otherwise structs and arrays are very different types. Explain how a struct object named \( S \) differs from an array object named \( A \) in terms of these aspects:
   a. (4 pts) The types of data that can be stored in \( S \) and \( A \)
   b. (4 pts) The ways that data stored in \( S \) and \( A \) are accessed

2. (7 pts) Define a C structure on the next page named \texttt{struct book} that has three data members:
   title: book title, a character string (may be as long as 49 characters)
   year: the year the book was published, an integer
   pages: the total number of pages in the book, another integer
Problems 3-5 refer to the struct book you defined for Problem 2.

3. Write C statements to accomplish the following steps in order:
   a. (3 pts) Define an object of struct book named cs16text.
   
   b. (5 pts) Let char s[] = “Engineering Problem Solving with C”. Copy this string into the title field of cs16text. Properly use the library function strcpy to do this. (See Etter 6.6, or http://www.cplusplus.com/reference/clibrary/cstring/strcpy/ for more information on using strcpy.)
   
   c. (4 pts) Set cs16text’s year field to 2005 and the pages field to 448. You must use two separate statements to do these things.
4. Consider the following code:

```c
void change1( struct book s ) {
    s.year = 1000;
}
void change2( struct book* s ) {
    s->year = 2000;
}
int main() {
    struct book temp = { "title", 2012, 50 };  
    change1( temp );
    printf( "%i\n", temp.year ); // prints 2012
    change2( &temp );
    printf( "%i\n", temp.year ); // prints 2000
    return 0;  
}
```

a. (4 pts) Calling `change1` does not change the value of `temp.year`. Why not?

b. (4 pts) Calling `change2` does change the value of `temp.year`. Why?
5. (9 pts) Write function definition for a function named printBook that takes a pointer to a struct book as its only parameter. printBook neatly prints the fields of the structure like the following example, and does not return anything. Here is how the function would word at the ch prompt (using the object created in Problem 3):

    ch> printBook( &cs16text ); // address of object passed
    Engineering Problem Solving with C
    published 2005
    448 pages

Pre-lab End. Adapted from Michael Costanzo by Kyle Dewey.