CS1010 Programming Methodology
Week 8: Searching and Sorting (Selected Answers)

> Every act of conscious learning requires the willingness to suffer an injury to one's self-esteem. That is why young children, before they are aware of their own self-importance, learn so easily. ~*Thomas Szasz*

3. We illustrated sorting algorithms using integer arrays in class. Determining whether one element, say a[i], is smaller than another, say a[j], is simply done by comparing a[i] with a[j] (e.g.: `if (a[i] < a[j])`).

   What if the array elements are more complex (for example, a structure comprising more than one component, to be covered later), or the comparison criterion is more complex?

   Suppose you want to sort an integer array of 6 elements in increasing order of the first 3 digits of each element, how would you modify the selection sort program **selection_sort.c** that was given in class?

   A sample run is shown below:

```
Enter size: 6
Enter 6 values:
12345
9870
32
5555555
801784
729
After sort:
32 12345 5555555 729 801784 9870
```

   Answer: See **selection_sort_modified.c**
   The key idea here is that we need to replace the comparison `(a[i] < a[j])` with a function **`lessthan(a[i], a[j])`**.

5. **Insertion Sort**
   Insertion Sort is another basic exchange sort besides Selection Sort and Bubble Sort. Refer to the PowerPoint file in the CS1010 module → "CA" → "Discussion" for the Insertion Sort algorithm. Implement Insertion Sort on an integer array.

   Answer: See **insertion_sort.c**

7. **Search for pattern**

In the minesweeper game, the character '*' represents a mine and the character '-' represents a safe cell on a minefield. Assuming that you have an 8×8 minefield, and a 2×3 pattern, write a program **search_pattern.c** to count the number of times the pattern appears in the minefield. A sample run is shown below.

Note: if you use scanf on characters, you will encounter some errors, because the 'enter' is itself a character, which will be read in. One way to overcome this is to use a space in the format specifier so that whitespace/enter characters are ignored, e.g.

```
scanf (" %c", &charvariable);
```

```
Enter minefield:
---****-
-*-**-**
********
-*-*--*-
********
**-*-**-
--------
********
Enter search pattern:
***
*-*
Answer = 4
```

Answer: See **search_pattern.c**