INTRODUCTION TO COMPUTER PROGRAMMING: FUNDAMENTALS OF C

Gilbert Pajela

Summer 2017

→ Lab 4: Arrays and Pointers

1. Write a program using pointers that prompts the user for two numbers and outputs the minimum.
2. Consider the following code:

   ```
   int arr[10];
   int *aptr = arr+3;
   ```

   Write a statement to print the last element of the array using only aptr (not arr).
3. Write a program that reads ten numbers, computes their average, and finds out how many numbers are below the average.
4. Write a program that subtracts two 3-by-3 matrices.
5. Given two vectors of length N that are represented with one-dimensional arrays, write a code fragment that computes the Euclidean distance between them (the square root of the sums of the squares of the differences between corresponding entries).
6. Given an array of N elements with each element between 1 and N, write an algorithm to determine whether there are any duplicates. You do not need to preserve the contents of the given array, but do not use an extra array.
7. Write a program to take two numerical lists of the same length ended by a sentinel value and store the lists in arrays x and y, each of which has 20 elements. Let n be the actual number of data values in each list. Store the product of corresponding elements of x and y in a third array, z, also of size 20. Display the arrays x, y, and z in a three-column table. Then compute and display the square root of the sum of the items in z. Make

up your own data, and be sure to test your program on at least one data set with number lists of exactly 20 items. One data set should have lists of 21 numbers, and one set should have significantly shorter lists.

8. A normalized vector X is defined as

$$x_i = \frac{v_i}{\sqrt{\sum\limits_{i=1}^{n} v_i^2}} \qquad i = 1, 2, ..., n$$

   Each element of the normalized vector X is computed by dividing the corresponding element (v_i) of the original vector by the square root of the sum of the squares of all the original vector's elements. Design and test a program that repeatedly scans and normalizes vectors of different lengths. Define functions `scan_vector`, `normalize_vector`, and `print_vector`.

9. Write an interactive program that plays a game of hangman. Store the word to be guessed in successive elements of an array of individual characters called `word`. The player must guess the letters belonging to `word`. The program should terminate when either all letters have been guessed correctly (the player wins) or a specified number of incorrect guesses have been made (the computer wins). Hint: Use another array, `guessed`, to keep track of the solution so far. Initialize all elements of `guessed` to the `'*'` symbol. Each time a letter in `word` is guessed, replace the corresponding `'*'` in `guessed` with that letter.

Exercises are adapted from the following source(s):

1. Hanly, J. R. (2012). *Problem solving and program design in C.* Pearson Education.
2. Liang, Y. D. (2007). *Introduction to Java programming: comprehensive version.* Pearson Prentice Hall.
3. Sedgewick, R., & Wayne, K. (2007). *Introduction to programming in Java: an interdisciplinary approach.* Addison-Wesley Publishing Company.