CSE 105, Lab 8, Summer 2006
cse.unl.edu/∼cstrope/csce105su06/lab/lab8

Instructor: Cory Strope

August 6, 2006

1. Arrays: Collections of data of the same type.

x[0]	x[1]	x[2]	x[3]	x[4]	x[5]
5	103	25	1	13	7

2. Characteristics of an array from a users view

   - Declaration of above array `int x[6]`; which means an array named `x` with 6 elements that are integers.
   - Value can be referenced by its location within the array, starting at index 0.
   - If there are n elements, index of array runs from 0 to n - 1.
   - The value stores at `x[4]` is 13.

3. Characteristics of an array from the computers view.

   - Declaration of above array `int x[6]`; means that the computer has reserved 8 locations of adjacent memory that is large enough to hold 8 integers.
   - The amount of memory needed to hold 8 elements of integers is different from the amount of memory needed to hold 8 elements of doubles.

4. Declaring arrays: `type array name[size]`

   - Type of data to be stored in array. Data types require different amounts of memory to store them and this depends on the processor of the computer. An example is as follows:
     - `char` - 2 bytes.
     - `int` - 4 bytes.

    - `double` - 8 bytes.

- Name of array: this is the name by which the array elements are referenced.
- Number of elements: The maximum number of elements that the array can hold.
- Array must be declared before it can be used, otherwise a compiler error for unknown symbol will occur.
- Once an array is declared, each element must be initialized for the value stored in them to be valid.

5. Initializing arrays:
    - One element at a time, element referenced by its location in the array : `x[0]= 5;`
    - All elements at once `x = { 5, 103, 25, 1, 13, 7};`
6. Multi-dimensional arrays: `type array name[rows][columns]`
    - first dimension is rows
    - second dimension is columns
    - third dimension is pages....level... sheet
    - what you call the next dimensions is up to you.
    - initializing multi-Dimensional arrays:
        - element at a time : `x[i][j]= 2;`
        - all elements at once `x = { { 2, 3, 5, 7, 11, 13},{3, 7, 8, 32, 5, 9},{4, 6, 9, 33, 2, 1} };`
7. Sorting data
    - To sort data is will be needed to swap the values between two variables. Review:
        - Let `int X = 5` and `int Y = 13`
        - You want to swap the values of these two variables.
        - You need a third variable as a “holding” variable, commonly `int temp`
        - The code to swap the values between `X` and `Y` would be as follows:

          ```
          temp = X;
          X = Y;
          Y = temp;
          ```
8. Programming:

(a) Look at the selection sort on the example page of the class web site. Run this program, and make the modification that is suggested.

(b) Download `game.c` and `game.x`. Note that `game.x` is an executable file, so you will need to save it in your `cse` account. Look at `game.c`. In my old age, I am becoming soft... I have already written half of a program for you. However, you will have to write the rest of the code; this code is the function `Finished`, which will take the array of already correct guesses and output 1 if the all positions have been matched, and will output 0 otherwise. You will also have to fill in the `main` function so that your game works exactly like `game.x`.

Isn't this fun! We're making games!!