

# Programming Design Worksheet - Redfield

## for CS1310 (programs 2-7) and CS1311 (programs 1-6)

Copy this file. Type and past images to create new documents for designs.  
Print it for class (if you must miss, submit one file to Designs).

First name **Davide** Last name **Russillo**

Design for program name **Lottery**

### DATA

#### Variables needed in WORDS for main and globally

constant lower bound of range of possible guesses initialized as 1  
constant upper bound of range of possible guesses initialized as 50  
constant length of the arrays initialized as 7  
guesses array of integers with length len  
results array of integers with length len

#### C DECLARATIONS for main & global

```
const range_lo = 1;  
const range_hi = 50;  
const int len = 7;  
int guesses[len];  
int results[len];
```

*(STARTING TicTacToe:put image; or draw: Insert, Drawing; or put at end of the file)*

#### draw in RAM with possible values

	0	1	2	3	4	5	6	
guesses	33	12	49	2	50	27	11	(before sort)

	0	1	2	3	4	5	6	
guesses	2	11	12	27	33	49	50	(after sort)

	0	1	2	3	4	5	6	
results	14	12	1	12	31	10	43	(before sort)

	0	1	2	3	4	5	6	
results	1	10	12	12	14	31	43	(before sort)

## **Algorithm to PSEUDOCODE level** for each function

(remember to indent under if, switch, while, do-while, for)

### **main:**

```
    set random seed to current time
    print enter len numbers between range_lo and range_hi
    call set_guesses_and_results(guesses, results, len)
    call bubble_sort(guesses, len)
    call bubble_sort(results, len)
    call check_guesses(guesses, results)
```

**other functions (bold the names):** (put them before main in the program!)

```
void set_guesses_and_results(int *guesses, int *results, int len)
    let valid_number be false (bool type)
    for each i from 0 to len-1
        while not valid_number
            print number (i):
            input number into guesses[i]
            set valid_number to guesses[i] >= than range_lo or
                                     <= than range_hi
        set results[i] to random number between range_lo and range_hi

void bubble_sort(int *array, int len)
    let swapped be true (bool type)
    declare int temp
    while swapped
        set swapped to false
        for i from 0 to len-2
            if array[i] > array[i+1]
                set temp = array[i]
                set array[i] = array[i+1]
                set array[i+1] = temp
            set swapped to true

void check_guesses(int *guesses, int *results, int len)
    let int correct_guesses be 0
    for each i from 0 to len-1
        if guesses[i] equals results[i]
            print you guessed guesses[i] correctly
            increment correct_guesses by one
    print correct guesses: correct_guesses
```