Exam 3 – Chapters 7 & 9
CSC 2100-002/003
29 Mar 2017

- Read through the entire test first BEFORE starting
- Put your name at the TOP of every page
- The test has 4 sections worth a total of 100 points
  - True/False – 10 questions, 20 points
  - Multiple Choice – 10 questions, 30 points
  - Short Answer – 2 questions, 10 points
  - Code – 1 question, 40 points

Section 1 – True / False

1. T F Each individual element of an array can be accessed by the array name and an element number, called a subscript.

2. T F The ampersand (&) is used to dereference a pointer variable in C++.

3. T F An individual array element can be processed like any other type of C++ variable.

4. T F With pointer variables you can access, but you cannot modify, data in other variables.

5. T F Assume array1 and array2 are the names of arrays. To assign the contents of array2 to array1, you would use the following statement.

   array1 = array2;

6. T F A pointer can be used as a function argument, giving the function access to the original argument.

7. T F It is legal to subtract a pointer variable from another pointer variable.

8. T F C++ limits the number of array dimensions to two.

9. T F C++ does not perform array bounds checking, making it possible for you to assign a pointer the address of an element out of the boundaries of an array

10. T F The amount of memory used by an array depends upon the array's data type and the number of elements in the array
Section 2 – Multiple Choice

11. Which statement displays the address of the variable num1?
   a. cout << num1;
   b. cout << *num1;
   c. cin >> &num1;
   d. cout << &num1

12. What does the following statement do?
    vector<int> v(10);
   a. It creates a vector object and initializes all of its elements to the value 10.
   b. It creates a vector object with a starting size of 10.
   c. It creates a vector object and initializes the first element with the value 10.
   d. It creates a vector object that can store only values of 10 or less

13. The following statement:
    int *ptr = new int;
   a. results in a compiler error
   b. assigns an integer less than 32767 to the variable named ptr
   c. assigns an address to the variable named ptr
   d. creates a new pointer named int

14. What will the following code do?
    const int SIZE = 5;
    double x[SIZE];
    for(int i = 2; i <= SIZE; i++)
    {
        x[i] = 0.0;
    }
   a. Each element in the array is initialized to 0.0
   b. Each element in the array, except the first, is initialized to 0.0
   c. Each element in the array, except the first and the last, is initialized to 0.0
   d. An error will occur when the code runs

15. A pointer variable is designed to store ________.
   a. any legal C++ value
   b. only floating-point values
   c. a memory address
   d. an integer
16. If you leave out the size declarator in an array definition
   a. You must furnish an initialization list
   b. You are not required to initialize the array elements
   c. All array elements will default to values of zero
   d. Your array will contain no elements

17. When you pass a pointer as an argument to a function, you must ________.
   a. declare the pointer variable again in the function call
   b. dereference the pointer variable in the function prototype
   c. use the #include<func_ptr.h> statement
   d. not dereference the pointer in the function's body

18. A two-dimensional array can have elements of ______________ data types
   a. one
   b. two
   c. four
   d. A or B

19. What will the following code output?
   ```
   int number = 22;
   int *var = &number;
   cout << var << endl;
   ```
   a. The address of the number variable
   b. 22
   c. An asterisk followed by 22
   d. An asterisk followed by the address of the number variable

20. Arrays may be ______________ at the time they are ______________.
   a. resized, executed
   b. re-scoped, deleted
   c. initialized, declared
   d. pre-compiled, typecast
Section 3 – Short Answer

21. Explain two differences between heap and stack memory

22. Explain how parallel array works and why are they better than multi-dimensional arrays
Section 4 – Code

22. Write a program that:
   
a. Creates a dynamically allocated array of size 2
b. Stores values inputted by the user into the array until a sentinel value is reached
c. Resizes the array to currentSize x 2 when necessary
d. Creates a second array AFTER the sentinel is reached that stores all the values in the first array, but in reverse order
e. Displays the contents of both arrays
f. Your program should use at LEAST 2 functions to accomplish the above tasks.