CS 161 Exam II Winter 2018 **FORM 1**

Please put your name and form number on the scantron.

True (A)/False (B) (28 pts, 2 pts each)

1. The following array declaration is legal

   double scores[]={0.1,0.2,0.3};

2. The following array has 12 indexed variables.

   int myArray[12]={1,2,3,6,5,4,7,1,2};

3. Indices are numbered starting at 1.

4. If a * is used in a declaration of a variable, ~~it is dereferencing that variable~~.

5. An & in a parameter listing indicates that ~~the variable will be passed via a pointer~~.

6. The pass by value method of passing arguments allows for the passed value to be altered at the address of its variable.

7. ~~The heap~~ is a subsection of memory which will be automatically freed at the end of a function.

8. An example of a memory leak is when memory is left allocated on the heap without a way to accesses it.

9. Recursion must have a base case in order to stop.

10. Following code means p is getting the address of a.

    int *p = &a;

11. A segmentation fault may occur as a result of attempting to access memory the program does not have access to.

12. C++ limits the number of array dimensions to two.

13. Arrays may only be declared dynamically.

14. C-style strings and C++ strings are the same thing.

```
char s[10];
cin >> s;
"hello"
 01234
cin >> s;
"jennifer"
```

Multiple Choice (72 pts, 3 pts each)

15. What are the valid indices for the array shown below?
    int myArray[25];
    a. 0-25
    b. 0-24
    c. 1-25
    d. 1-24

16. Which of the following assigns to p1 the address of value?
    a. *p1=&value;
    b. p1=value;
    c. p1=&value;
    d. &p1 = *value;

17. Call-by-reference parameters are passed
    a. nothing
    b. the actual argument.
    c. the value in the actual argument.
    d. the address of the argument.

18. Which of the following statements correctly prints out the value that is in the memory address that the pointer p1 is pointing to?
    a. cout << &p1;
    b. cout << p1;
    c. cout << int* p1;
    d. cout << *p1;

19. What is the output of the following code?

```
int *p1, *p2;
p1 = new int;
p2 = new int;
*p1=11;
*p2=0;
p2=p1;
cout  << *p1 <<" " << *p2 << endl;
```

    a. 11 0
    b. 0 11
    c. 11 11
    d. 0 0

20. Which of the following correctly declares a dynamic array of strings?
    a. p1 = new string(13);
    b. p1 = new string[];
    c. p1 = new string[13];
    d. p1 = new stringArray(13);

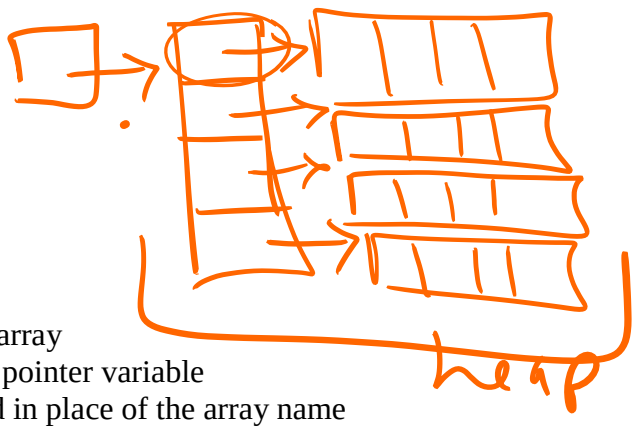
21. Which of the following statements correctly returns the memory from the dynamic array pointer to by p1 to the freestore?
    a. delete [] p1;
    b. delete p1[];
    c. delete *p1;
    d. delete p1;

22. What is wrong with the following function body?

```
void calculate(int count, float price, float& cost)
{
        if (count < 0)
                cost=0.0;
        else
                cost=count*price;
        return;
}
```

    a. void functions may not have a return statement.
    b. void functions must return a value
    c. nothing
    d. can not mix reference and value parameters

23. If a program requires a dynamically allocate two-dimensional array, you would allocate the memory by using
    a. p1 = new int*[numRows];
       for(int i=0; i < numRows; i++)
               p1[i] = new int[numColumns];
    b. p1 = new int*[numRows][numColumns];
    c. p1 = new[numRows][numColumns]int;
    d. none of the above

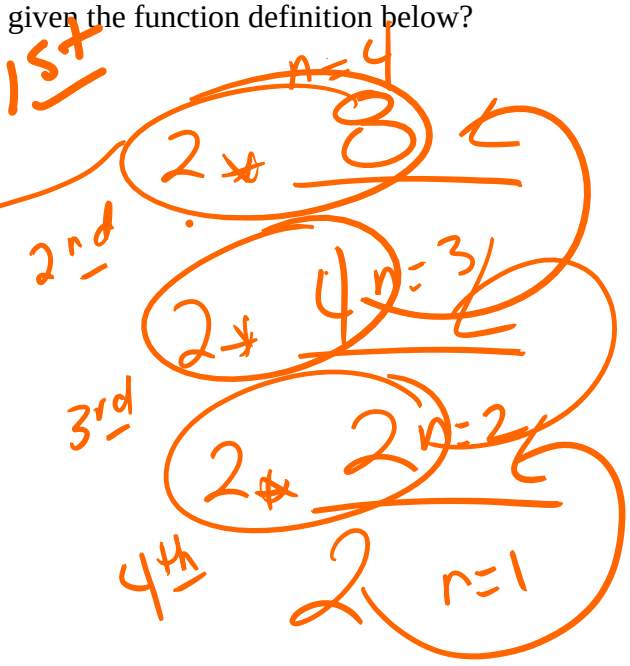


24. Which of the following is not true?
    a. a pointer can be assigned the address of an array
    b. a static array can be assigned the value in a pointer variable
    c. if a pointer points to an array, it can be used in place of the array name
    d. if a pointer points to an array, the pointer does not know how many elements are in the array.

25. What is the output of the following function call, given the function definition below?

```
int tester (int n) {
        if (n == 1)
                return 2;
        return 2 * tester (n-1);
}
cout << tester(4);
```

a. 2
b. 4
c. 16
d. 32



26. The function

```
int fact (int k) {
        return k*fact(k-1);
        if (k == 0) return 1;
}
```

a. Computes the factorial on an integer k passed to it as a parameter.
b. Returns the value 1 if it is passed a value of 0 for the parameter k.
c. Does not correctly handle its base case.
d. Works for all non-negative values of k, but not for negative numbers.

27. The name of an array stores the _____ of the first array element.

a. Value
b. memory address
c. element number
d. data type

28. 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 array elements
c. all array elements default to zero values
d. your array will contain no elements

29. What is the last legal index that can be used with the following array?

```
int values[5];
```

a. 0
b. 5
c. 6
d. 4

30. To assign the contents of one array to another, you must use

a. the assignment operator with the array names
b. the equality operator with the array names
c. a loop to assign the elements of one array to the other array
d. Any of these

31. A two-dimensional array of characters can contain
    a. strings of the same length
    b. strings of different lengths
    c. uninitialized elements
    d. All of these

32. The escape sequence that represents the null terminator is
    a. \n
    b. \t
    c. \0
    d. Nullptr

33. The C-string company[12] can hold
    a. twelve characters
    b. thirteen characters
    c. eleven characters and the null terminator
    d. twelve characters and the null terminator

34. It is _____ to pass an argument to a function that contains an individual array element, such as numbers[3].
    a. illegal in C++
    b. legal in C++
    c. not recommended by the ANSI committee
    d. not good programming practice

35. When you work with a dereferenced pointer, you are actually working with _____.
    a. a variable whose memory has been allocated
    b. a copy of the value pointed to by the pointer variable
    c. the actual value/contents of the variable whose address is stored in the pointer variable
    d. all of these

36. Dynamic memory allocation occurs _____.
    a. when a new variable is created by the compiler.
    b. when a new variable is created at runtime.
    c. when a pointer fails to dereference the right variable
    d. when a pointer is assigned an incorrect address.

37. If a function calls itself, it is known as being _____.
    a. recursive
    b. repetitive
    c. distinct
    d. none of the above

38. Command line arguments allow user input to be taken before the program runs as _____.
    a. C++ strings
    b. C-style strings
    c. integers
    d. none of the above