

CSC 122 : Computer Science II – Exam #1

Name: _____

This exam is closed book, closed notes, and closed neighbor. You **must** write all your answers on this exam sheet – any answer not on these sheets will not be graded. There are a total of 50 points on this exam. Good luck!

1. Consider the following array:

16	8	11	4	5	2	9	13
----	---	----	---	---	---	---	----

Show the results of running selection sort on this array. Show the first two passes through selection sort. (2 points)

2. Show the results of running bubble sort on the original array from question #1. Show the results of the first two passes. (2 points)

3. Answer the following questions about sorting. For all parts, assume that running time includes comparisons between items in the array and swaps or assignments for values in the array. (3 points)

(a) Show an array that would result in the best running time for bubble sort.

--	--	--	--	--	--

(b) Show an array that would result in the worst running time for insertion sort.

--	--	--	--	--	--

(c) Show an array for which bubble sort would have a worse running time than insertion sort.

--	--	--	--	--	--

4. Consider the following code fragment:

```
for ( int i = 0; i < n - 1; i++ ) {  
    for ( int j = 0; j < n - 1; j++ ) {  
        cout << "Hello!" << endl;  
    }  
}
```

How many "Hello!"s are printed? Give an exact answer (in terms of n) as well as an answer in $O()$ notation. (2 points)

5. Consider the following code fragment. (3 points)

```
for (int i = 0; i < n; i++ ) {
    j = 1;
    while( j < n ) {
        cout << "Hello!" << endl;
        j = j * 2;
    }
    cout << "Hello" << endl;
}
```

- (a) How many "Hello"s are printed when $n = 5$?
- (b) How many "Hello"s are printed in terms of n ?
- (c) How many "Hello"s are printed in $O()$ notation?

6. What is the $O()$ of the following expressions? Explain your answers. (3 points)

(a) $n^3 + 10n^2 + 100n + 1000$

(b) $n \lg n + n^2 + 10$

(c) $n^5 + 2^n$

7. Consider the following function. (4 points)

```
void mystery( int n, int m ) {
    if ( m > n ) {
        cout << m << " ";
        mystery( m - 1, n );
    }
    else if ( m < n ) {
        cout << n << " ";
        mystery( m + 1, n );
    }
    else
        cout << "Done" << endl;
}
```

What is the output of the following function calls? If the function never terminates, indicate so.

- (a) `mystery(3, 3)`
- (b) `mystery(5, 4)`
- (c) `mystery(-1, 2)`

8. Consider the following function. (4 points)

```
int enigma( int n ) {
    if ( n == 0 )
        return 1;
    else if ( n < 5 )
        return n + enigma( n - 2 );
    else if ( ( n % 2 ) == 0 )
        return 2 * enigma( n / 2 );
    else
        return 3 * enigma( n - 1 );
}
```

What is the output of the following function calls? If the function never terminates, indicate so.

(a) enigma(2)

(b) enigma(9)

(c) enigma(12)

9. Given the following HasCl code:

```
test1([ ], y) = [ ]
test1([x:xs], y) = [x * y : test1(xs, y)]
```

what is the value of test1([2, 3, 4], 5)? (2 points)

10. Given the following HasCl code:

```
test2(0, [x:xs]) = x
test2(n, [x:xs]) = test2(n-1, xs)
test2(n, [ ]) = 0
```

what is the value of test2(1, [1, 2, 3]) + test2(2, [1, 2, 3]) + test2(3, [1, 2, 3])? (2 points)

11. Write a HasCl function named powersOf2 which takes an integer, n , and returns a list containing decreasing powers of 2, from 2^n down to 1 (recall that the HasCl expression a^n gives the n th power of a). You may assume that the input will be non-negative. For example, the value of powersOf2(4) is [16, 8, 4, 2, 1]. (6 points)
12. Write a HasCl function named diffs, which takes a list of integers and returns a list giving the differences between successive elements of the list. If the list has fewer than 2 elements, then the result is the empty list. For example, the value of diffs([1, 2, 3]) is [1, 1], the value of diffs([2, 7, 1, 8]) is [5, -6, 7], and the value of diffs([42]) is []. (6 points)
13. Write a C++ function called printStars that takes one integer parameter called n . This function should print a single line of n stars (*) on the screen. To receive full credit, this function must be written recursively. Little to no credit will be given for a non-recursive solution. For example, printStars(5) would print ***** to the screen. (5 points)
14. Write a C++ function called calcProduct that takes two integer parameters, m and n . This function should return the product of all the integers from m to n inclusive. To receive full credit, this function must be written recursively. Little to no credit will be given for a non-recursive solution. For example, calcproduct(5, 3) should return 60 (5 x 4 x 3). You cannot assume that the parameters are in any particular order. (6 points)