Computer Science Foundation Exam

August 14, 2015

Section I A

COMPUTER SCIENCE

**NO books, notes, or calculators may be used,
and you must work entirely on your own.**

Name: _____

UCFID: _____

Question #	Max Pts	Category	Passing	Score
1	10	DSN	7	
2	10	ANL	7	
3	10	ALG	7	
4	10	ALG	7	
5	10	ALG	7	
TOTAL	50			

You must do all 5 problems in this section of the exam.

Problems will be graded based on the completeness of the solution steps and <u>not</u> graded based on the answer alone. Credit cannot be given unless all work is shown and is readable. Be complete, yet concise, and above all <u>be neat</u>.

1) (10 pts) DSN (Recursive Functions)

Consider the problem of transforming a positive integer X into a positive integer Y when the only two operations you are allowed are adding 1 to the current number or multiplying the current number by 2. Write a recursive function that returns the minimum number of steps necessary to transform X to Y. If X > Y, return 1000000000, to indicate that no solution exists. (For example, if X = 13 and Y = 28, the correct response would be 2 - first you would add 1 to 13 to obtain 14, then multiply 14 by 2 to obtain 28.) Feel free to call the provided function. *Note: don't worry about the run time of your function - assume that the inputs are such that the run time is relatively small, even when written using straight-forward recursion. There is a clever, efficient solution without recursion but please write the slower recursive solution since the goal of this question is to test recursive thinking.*

```
#define NO_SOLUTION 1000000000

int min(int x, int y) {
    if (x < y) return x;
    return y;
}

// Returns the minimum number of steps to transform x into y, or
// 100000000 to indicate no solution.
int minSteps(int x, int y) {









}
```

2) (10 pts) ANL (Summations and Algorithm Analysis)

Consider the following segment of code, assuming that n has been previously declared and initialized to some positive value:

```
int i, j, k;
for (i = 1; i <= n; i++){
    for(k =1; k <= i; k++){
        j = k;
        while(j > 0)
            j--;
    }
}
```

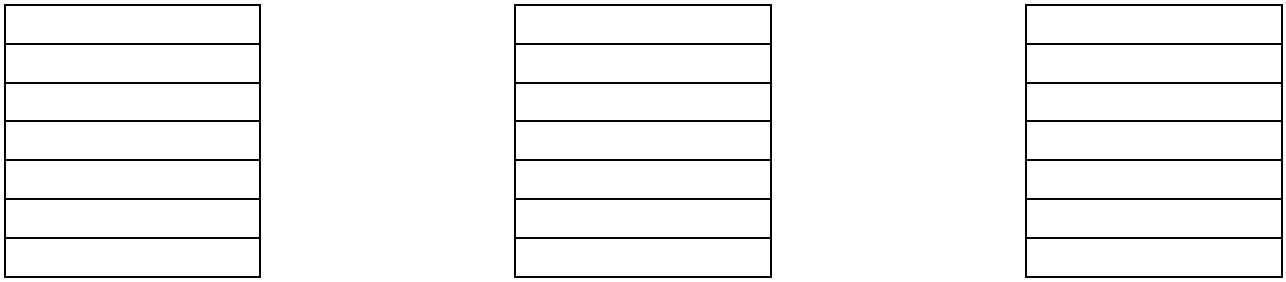
(a) (3 pts) Write a summation (3 nested sums) equal to the number of times the statement `j--;` executes, in terms of n.

(b) (7 pts) Determine a closed form solution for the summation above in terms of n.

3) (10 pts) ALG (Stacks)

Use a stack to evaluate the postfix expression below. Please show the state of the stack at the exact point in time during the algorithm that the marked (A, B, C) locations are reached while processing the expression. Also, write down the equivalent infix expression, placing parentheses when necessary.

				A				**B**						**C**
5	**8**	**3**	*****	**+**	**6**	**+**	**7**	**/**	**4**	**2**	**–**	*****		



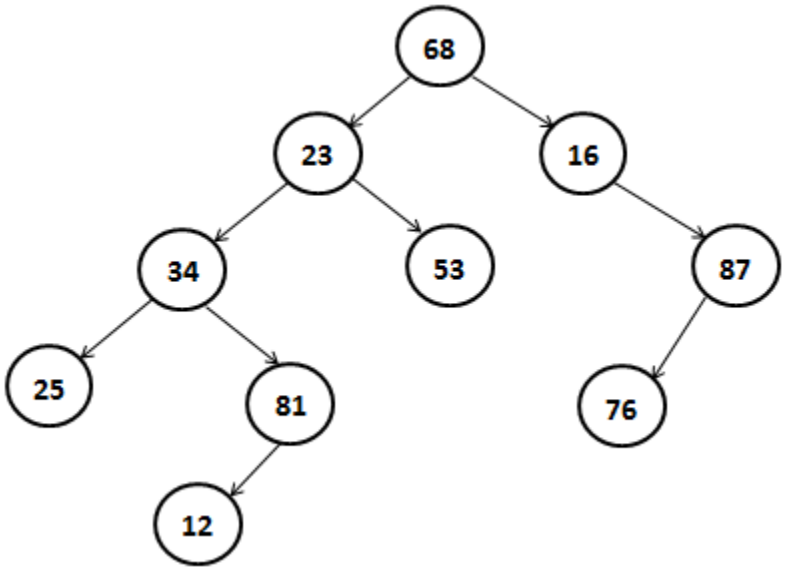
A B C

Value of the Expression: _____

Equivalent Infix Expression: _____

4) (10 pts) ALG (Binary Trees)

Please give the preorder, inorder and postorder traversal of the binary tree shown below. In addition, determine whether or not the tree is a valid binary search tree.



Preorder: ____ , ____ , ____ , ____ , ____ , ____ , ____ , ____ , ____ , ____

Inorder: ____ , ____ , ____ , ____ , ____ , ____ , ____ , ____ , ____ , ____

Postorder: ____ , ____ , ____ , ____ , ____ , ____ , ____ , ____ , ____ , ____

Is this valid binary search tree? (Circle your answer.) YES NO

5) (10 pts) ALG (Base Conversion)

(a) (5 pts) Convert 1654_8 to hexadecimal.

(b) (5 pts) Convert 1925_{10} to octal.

Computer Science Foundation Exam

August 14, 2015

Section I B

COMPUTER SCIENCE

NO books, notes, or calculators may be used,
and you must work entirely on your own.

Name: _____

UCFID: _____

Question #	Max Pts	Category	Passing	Score
1	10	ANL	7	
2	10	ANL	7	
3	10	DSN	7	
4	10	DSN	7	
5	10	ALG	7	
TOTAL	50		35	

You must do all 5 problems in this section of the exam.

Problems will be graded based on the completeness of the solution steps and <u>not</u> graded based on the answer alone. Credit cannot be given unless all work is shown and is readable. Be complete, yet concise, and above all <u>be neat</u>.

1) (10pts) ANL (Algorithm Analysis)

Consider the recursive function `diminish` shown below:

```
double diminish(int m, int n){
    if (n == 0)
        return m;
    return  1.0/2*diminish(m,n-1)
}
```

(a) (3 pts) Let T(n) represent the run time of the function `diminish`. Write a recurrence relation that T(n) satisfies.

(b) (6 pts) Using the iteration method, determine a closed-form solution (Big-Oh bound) for T(n).

(c) (1 pt) In terms of the values of `m` and `n`, respectively, what does the function call `diminish(m,n)` return? (You may assume that `m` and `n` are both positive.)

2) (10 pts) ANL (Algorithm Analysis)

(a) (5 pts) An algorithm for searching for a housing contract in a database of n records takes $O(\lg n)$ time. When $n = 2^{20}$, one million searches can be performed in one fifth of a second. If we increase the database to size $n = 2^{25}$, how long will 500,000 searches take?

(b) (5 pts) A shortest distance algorithm on an $n \times m$ street grid runs in $O(nm)$ time. If the algorithm takes 2 seconds to run on a 4000×3000 sized grid, how long will it take on a grid of size 2000×18000 sized grid?

3) (10 pts) DSN (Linked Lists)

Write a function, `moveFrontToBack`, that takes in a pointer to the front of a ***doubly*** linked list storing an integer, moves the first node of the list to the back of the list and returns a pointer to the new front of the list. If the list contains fewer than two elements, the function should just return the list as it is. (Note: prev points to the previous node in the list and next points to the next node in the list.)

Use the struct definition provided below.

```
typedef struct dllnode {
    int value;
    struct dllnode* prev;
    struct dllnode* next;
} dllnode;

dllnode* moveFrontToBack(dllnode* front) {




}
```

4) (10 pts) DSN (Binary Trees)

Mark and his buddy Travis have devised a password scheme to secure files that they send among themselves. Their scheme hides the password in a string of English letters. The password is the alphabetically ordered sequence of the consonants in the string. So as not to have to compute the password each time, Mark has written a function called `printPassword`, which takes the letters of the original string stored in a binary search tree and prints out the password. For example, if the string in the message is **mental**, the password printed out would be **lmnt**. Or if the string was ***fragile***, then the password would be ***fglr.*** You may call the following function in your solution:

```
// Returns 1 if c is a consonant, 0 otherwise.
int isConsonant(char c)
```

Using the struct definition given below, complete the function in the space provided.

```
typedef struct treenode {
    char ch;
    struct treenode *left;
    struct treenode *right;
} treenode;

void printPassword(treenode* root) {









}
```

5) (10 pts) ALG (Sorting)

(a) (4 pts) Consider sorting the array below in ascending order using Bubble Sort. Show the contents of the array after each iteration of the outer loop.

Original	6	12	1	9	4	2
1st iteration						
2nd iteration						
3rd iteration						
4th iteration						
5th iteration						

(b) (6 pts) Please provide the best case and worst case run times (Big-O) for each of the following three sorting algorithms, in terms of n, the number of elements being sorted.

Sort	**Best Case**	**Worst Case**
Merge Sort		
Quick Sort		
Insertion Sort		

Computer Science Foundation Exam

August 14, 2015

Section II A

DISCRETE STRUCTURES

NO books, notes, or calculators may be used,
and you must work entirely on your own.

Name: _____

UCFID: _____

Question	Max Pts	Category	Passing	Score
1	15	PRF (Induction)	10	
2	15	PRF (Logic)	10	
3	10	PRF (Sets)	6	
4	10	NTH (Number Theory)	6	
ALL	50		32	

You must do all 4 problems in this section of the exam.

Problems will be graded based on the completeness of the solution steps and <u>not</u> graded based on the answer alone. Credit cannot be given unless all work is shown and is readable. Be complete, yet concise, and above all <u>be neat</u>.

1) (15 pts) PRF (Induction)

Use mathematical induction to prove that, for every positive integer n,

$$3 \mid (5^n + (-1)^{n+1})$$

2) (15 pts) PRF (Logic)

Validate the following argument using the laws of logic, substitution rules or rules of inference. List the rule used in each step and label the steps used in each derivation.

$$\begin{array}{l} (p \vee r) \rightarrow q \\ \neg s \\ (p \wedge r) \vee s \\ \neg q \vee t \\ \hline \therefore t \end{array}$$

3) (10 pts) PRF (Sets)

Prove the following statement:

Let A and B be two finite sets. Prove that $A \nsubseteq B \Rightarrow A \neq \emptyset \wedge \Big(\exists x \left(x \in (A \cup B)\right)\Big)$.

4) (10 pts) NTH (Number Theory)

Prove for an arbitrary prime number p that there always exists some composite number q where $\gcd(p, p+q) > 1$.

Computer Science Foundation Exam

August 14, 2015

Section II B

DISCRETE STRUCTURES

**NO books, notes, or calculators may be used,
and you must work entirely on your own.**

Name: _____

UCFID: _____

Question	Max Pts	Category	Passing	Score
1	15	CTG (Counting)	10	
2	15	PRB (Probability)	10	
3	10	PRF (Functions)	6	
4	10	PRF (Relations)	6	
ALL	50		32	

You must do all 4 problems in this section of the exam.

Problems will be graded based on the completeness of the solution steps and <u>not</u> graded based on the answer alone. Credit cannot be given unless all work is shown and is readable. Be complete, yet concise, and above all <u>be neat</u>.

1) (15 pts) CTG (Counting)

Please leave your answers in factorials, permutations, combinations and powers. Do not calculate out the actual numerical value for any of the questions. Justify your answers.

Each of the following questions concerns a classroom with five girls and five boys. For each part below, treat each of these 10 students as being distinguishable from one another.

(a) (3 pts) The teacher wants to create five teams, each with one girl and one boy. How many different sets of teams can the teacher create? Note: two sets of teams are different if at least one pair of students on the same team in the first set of teams is on different teams in the second set of teams.

(b) (5 pts) In how many ways can the class line up such that no girl is next to another girl and no boy is next to another boy?

(c) (7 pts) How many subsets of students in the class contain more girls than boys? (Please give an exact numeric answer. Note that the 5^{th} row of Pascal's Triangle is 1, 5, 10, 10, 5 and 1.)

2) (15 pts) PRB (Probability)

(a) (7 pts) 2% of the population has disease A. Given that a patient has disease A, a test, T, correctly identifies that the patient has the disease 95% of the time. Given that a patient does NOT have disease A, T correctly identifies that the patient doesn't have the disease 80% of the time. You've taken test T and have tested positive for disease A. What is the actual probability that you have the disease? (Leave your answer as a division of two real numbers, each expressed with three digits after the decimal.)

(b) (8 pts) Consider a partition of an array of 2n+1 distinct integers where the partition element is randomly chosen. (Note: this is where we place all integers less than the partition element on the left of it and all the integers greater than the partition element on the right of it.) Let L equal the number of values less than the partition element and R equal the number of values greater than the partition element. (Thus, L + R = 2n.) Calculate the expected value of |L - R|.

3) (10 pts) PRF (Functions)

Let $f(x) = \sqrt{3x^2 + 5x + 7}$ and $g(x) = 2^{x^2}$, both with the domain $x \geq 0$.

(a) (7 pts) What is $g(f(x))$? (Simplify your answer for full credit.)

(b) (3 pts) What is the range of $g(f(x))$, given that its domain is all real values of $x \geq 0$.

4) (10 pts) PRF (Relations)

(a) (5 pts) Let A = {1, 2, 3, 4}, B = {x, y, z} and C = {m, n}. Let R = {(1, x), (1, z), (2, y), (4, x), (4, y), (4, z)} and S = {(x, m), (y, m), (y, n)}. What is $S \circ R$?

(b) (5 pts) Let A = {1, 2, 3}. There are 5 equivalence relations over $A \times A$. Explicitly list all five.