15110 PRINCIPLES OF COMPUTING – EXAM 2A – SPRING 2012

Name SOLUTIONS Section _____

Directions: Answer each question neatly in the space provided. Please read each question carefully. You have 50 minutes for this exam. No electronic devices allowed. Good luck!

1	
2	
3	
4	
5	
TOTAL	

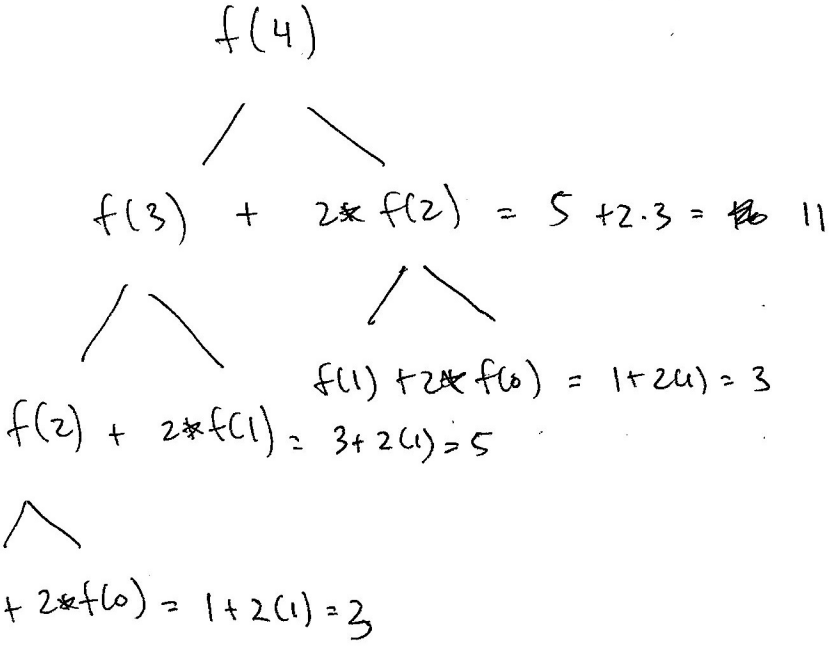
1. (20 pts)The following question deals with recursion and recursive algorithms.

(a) (6 pts) The function *f* is defined recursively as follows:

$$f(n) = \begin{cases} 1 & \text{if } n = 0 \text{ or } n = 1 \\ f(n-1) + 2 * f(n-2) & \text{if } n > 1 \end{cases}$$

Compute f(4) by drawing a recursion tree showing all of the computation required and then use your tree to compute the answer.

Recursion Tree: *f*(4) = 11



(b) (4 pts) You are asked to guess a secret number between 1 and 1 billion by asking yes/no questions of the form "Is the number less than X?" (e.g. "Is the number less than 1 million?")

What algorithm would you use to efficiently find the secret number? binary search

Using your algorithm, what is the minimum number of questions you need to ask so you are guaranteed to find the secret number? 30
HINT: 1 billion bytes is approximately equal to 1 GB. 1 billion $\approx 2^{30}$

$\log_2 2^{30} = 30$

(c) (6 pts) Complete the following Python function recursively so that it computes 3^n for $n \geq 0$. You may assume n is an integer. Do not use a loop in your answer.

```
def power3(n):
    if n==0:
        return 1
    else:
        return 3 * power3(n-1)
```

$3^n = 3 \cdot 3^{n-1}$

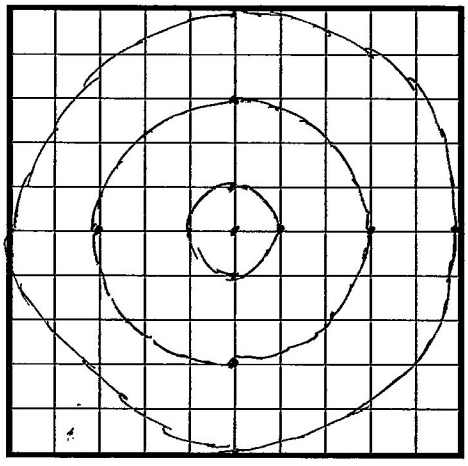
(d) (4 pts) On a computer system, a user can draw a circle using the function

```
draw_circle(center_x, center_y, radius)
```

where `center_x`, `center_y`, and `radius` are given in pixels.

In the 200 X 200 window shown below, draw what would be displayed by the following recursive function if it is initially called with `x` = 100, `y` = 100, and `r` = 100. (Grid lines are provided for you to help you draw your picture.)

0,0



200,200

```
def recursive_draw(x, y, r):
    if r > 0:
        draw_circle(x, y, r)
        recursive_draw(x, y, r-40)
```

Sorry drawing isn't my strong suit

2. (20 pts) This problem focuses on the representation of data in a computer.
The following tables may be helpful in this question:

2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
1024	512	256	128	64	32	16	8	4	2	1

Bin	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

(a) (2 pts) Compute the decimal value of the byte 11111111 if it is interpreted as an unsigned integer.

$2^8 - 1 = 255$

HINT: $\sum_{i=0}^{n} 2^i = 2^{n+1} - 1$.

flip 11111111 → 00000000 → add 1 → 00000001 => 1

(b) (2 pts) Compute the decimal value of the byte 11111111 if it is interpreted as a signed 2's complement integer.

-1

(c) (2 pts) Express the byte 11011001 in hexadecimal.

13 9 => D9

D9

(d) (2 pts) The ASCII character 'Y' is represented in binary using 7 bits as 1011001. The character is to be sent via satellite using *odd parity*. What eighth bit is sent along with this byte: 1 or 0?

1

(e) (2 pts) Suppose that exactly one bit is corrupted during transmission of the eight bits from part (d) and is "flipped" (either from 0 to 1 or 1 to 0). Which of the following is true? Select the appropriate letter and write it here:

B

(A) The receiver cannot detect the error.
(B) The receiver can detect the error but cannot determine which bit is wrong.
(C) The receiver can detect the error and can correct the bit that is wrong.

(f) (4 pts) In an HTML file for a webpage, the designer used the following font tag that changes the color of the font based on the 6 digit hexadecimal value 804020.

```
<font color="#804020">This is a colorful sentence.</font>
```

The 6 digit hexadecimal value specifies the amount of red, green and blue for the font's color, respectively. Express the amount of green in the font as an integer between 0 and 255, inclusive. Show your work.

green => 0x40 in hex

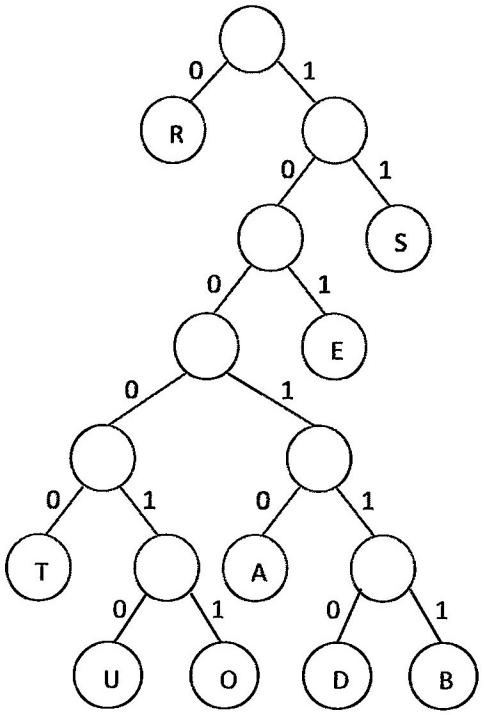
= 0100 0000 in binary

$= 2^6 = 64$

(g) (2 pts) Does the MP3 sound file format use a *lossless* or *lossy* compression algorithm?

lossy

(h) (4 pts) Based on the following Huffman tree:



What word is represented by the following binary string based on the Huffman tree:

1110000100100101

S T A R E

STARE

Suppose we want to encode words made using the nine letters from the tree above using a *fixed-width encoding* with the fewest bits possible for each letter.
How many bits are required to encode each letter?

4 bits

9 letters

each bit can 0 or 1

need 4 bits

3 bits can only have $2^3 = 8$ different combinations

3. (20 pts) This problem focuses on the structuring of data in a computer.

(a) (7 pts) A hash table of size 11 is used to store integer keys. The hash table is implemented as an array of "buckets" with a hash function of h(key) = key modulo 11. Store the following keys into the table using the given hash function.

18, 34, 98, 29 $18 \% 11 = 7$ $34 \% 11 = 1$ $98 \% 11 = 10$ $29 \% 11 = 7$

0	1	2	3	4	5	6	7	8	9	10
	34						18 29			98

Using Python, show how this hash table of 4 integers would be represented as a two-dimensional array.

table = [[], [34], [], [], [], [], [], [18, 29], [], [], [98]]
0 1 2 3 4 5 6 7 8 9 10

In general, if n data elements are evenly distributed within the hash table and the hash table is sufficiently large so that each bucket contains only a few elements, then a search for an element in the hash table takes how much time using big O notation? O(1)

(b) (3 pts) You are given a sequence of 1 million integers that can be stored in an *array* or a *linked list*.

Which structure is faster if you want to access the 800,000th integer? array

Which structure is faster if you want to insert a new integer between the first and second integers in the sequence? linked list

Which structure requires less memory in a computer? array

(c) (5 pts) Show how to compute the following RPN expression using a stack, showing the contents of the stack each time something is pushed or popped. The first three stacks are shown for you.

9 5 2 + 6 * 3 1 - / +

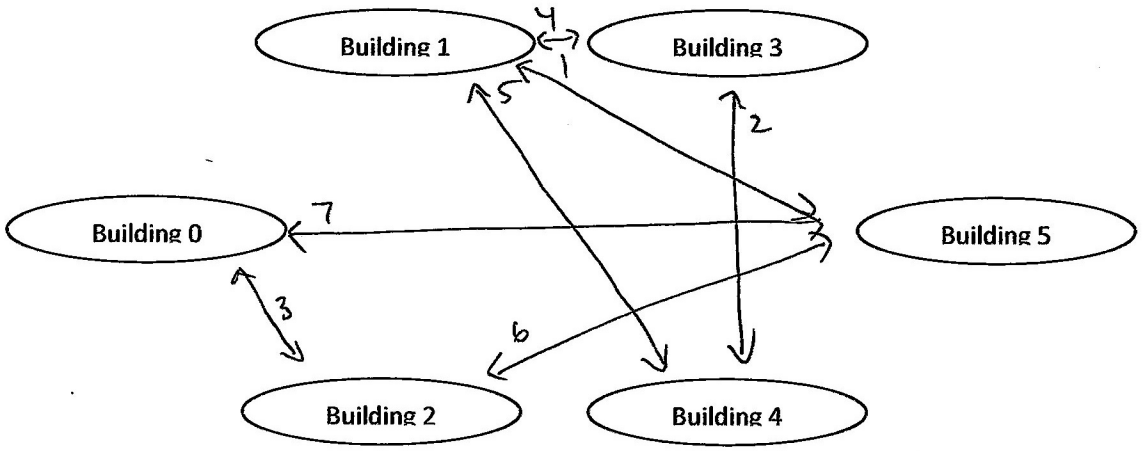
Stacks:

		2		6		3	3	2		
	5	5	7	7	42	42	42	42	21	
9	9	9	9	9	9	9	9	9	9	30

(d) (5 pts) A college campus has 6 buildings, connected with 7 direct walking paths between buildings. The matrix below represents a graph for this campus, where the weights indicate the amount of time it takes to walk down each walkway in minutes.

	0	1	2	3	4	5
0	0	∞	3	∞	∞	7
1	∞	0	∞	4	5	1
2	3	∞	0	∞	∞	6
3	∞	4	∞	0	2	∞
4	∞	5	∞	2	0	∞
5	7	1	6	∞	∞	0

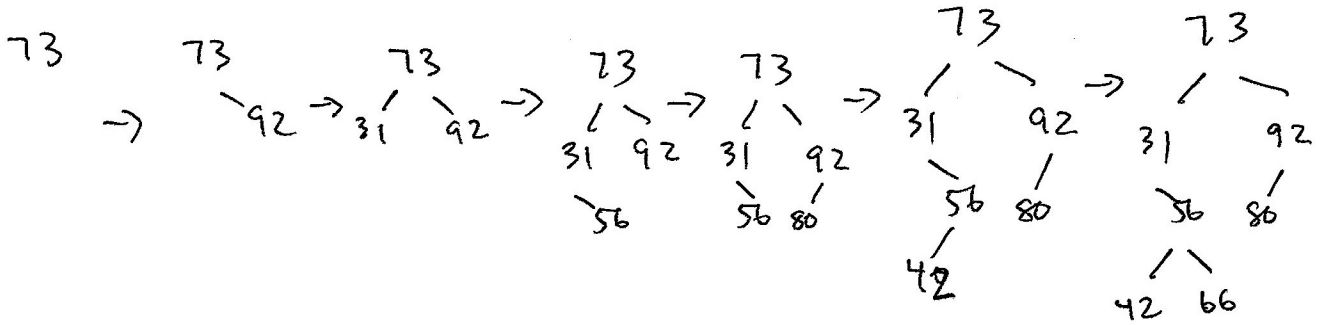
Complete the graph below by adding in the appropriate weighted edges based on the representation given above for the college campus. (If edges cross in your graph, you may assume that there is a bridge so students can't change paths midway between buildings.) Be sure to complete your graph so it is unambiguous for maximum credit.



4. (20 pts) This question deals with binary trees.

(a) (4 pts) Draw the binary search tree that results by inserting the following integers into the tree in the order shown.

73 92 31 56 80 42 66



(b) (4 pts) If you have a binary search tree with 1000 nodes,

what is the minimum number of levels this tree can have? $\log_2 1000 \approx 10$

what is the maximum number of levels this tree can have? 1000

Consider a tree where you insert the numbers 1, 2, 3 ... 1000 consecutively

(c) (4 pts) A heap of 1000 nodes is stored in an array, level by level, starting with the root at index 0. For the heap node stored at index 4 in the array,

at what index is its left child stored? NOT

at what index is its right child stored? CONFUSED

(d) (4 pts) For a max-heap, the maximum is always stored in the root. Describe a simple algorithm in English for finding the maximum in a **binary search tree**.

Keep going right

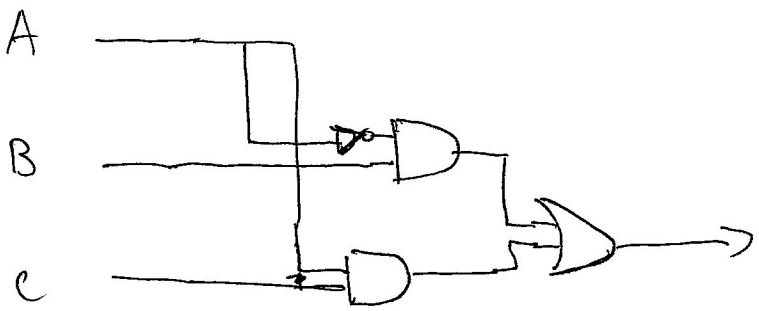
5. (20 pts) The following question involves Boolean logic and abstraction.

(a) (9 pts) Let $S = (A \wedge C) \vee (B \wedge \neg A)$, where A, B and C are Boolean variables. Fill in the truth table below to compute S.

A	B	C	$A \wedge C$	$B \wedge \neg A$	S
0	0	0	0	0	0
0	0	1	0	0	0

0	1	0	0	1	1
0	1	1	0	1	1
1	0	0	0	0	0
1	0	1	1	0	1
1	1	0	0	0	0
1	1	1	1	0	1

(b) (6 pts) The Boolean value S can be computed by an electronic circuit. Draw this circuit at the gate level of abstraction.



(c) (2 pts) A Boolean function contains 7 different Boolean variables. How many different possible logical assignments are there for this function? In other words, how many rows would your truth table need to define this function?

$2^7 = 128$

(d) (3 pts) Use DeMorgan's Law to transform the following Python expression to an equivalent statement

`¬(x > 0 or y == 100)` ⇔ $x <= 0$ and $y != 100$

not $(x > 0)$ and not $(y == 100)$

$\therefore$ $x <= 0$ and $y != 100$