Computer Science Foundation Exam

August 2, 2002

COMPUTER SCIENCE I

Section I B

No Calculators!

KEY

Name: _____

SSN: _____

Score: ¤ 50

In this section of the exam, there are four (4) problems

You must do all of them.

The weight of each problem in this section is indicated with the problem. The algorithms in this exam are written in C. Any algorithms that you are asked to produce should use a syntax that is clear and unambiguous. Partial credit cannot be given unless all work is shown.

As always, be complete, yet concise, and above all <u>be neat</u>. Credit cannot be

given when your results are unreadable.

1. **(10 points – 8pts(a), 2pts(b))**

   Answer questions (a) and (b) for the recursive function shown below.

   (a) Provide a trace of the execution of this function when it is initially called with the statement: **x = f(3,15)**.

   (b) What is the final value that is assigned to **x** by this function?

```
int  f (int x, int y)
{
     if (x == y || x == 7 || y == 9)
          return y–x;
     else
          return (f(x+1, y) + f(x, y–3));
}
```

Solution:

x = f(3,15) = f(4,15) + f(3,12)
= f(5,15) + f(4,12) + f(3,12)
= f(6,15) + f(5,12) + f(4,12) + f(3,12)
= f(7,15) + f(6,12) + f(5,12) + f(4,12) + f(3,12)
= 8 + f(7, 12) + f(6, 9) + f(5,12) + f(4,12) + f(3,12)
= 8 + 5 + 3 + f(6,12) + f(5,9) + f(4,12) + f(3,12)
= 8 + 5 + 3 + f(7,12) + f(6, 9) + f(5,9) + f(4,12) +f(3,12)
= 8 + 5 + 3 + 5 + 3 + 4 + f(5,12) + f(4, 9) + f(3,12)
= 8 + 5 + 3 + 5 + 3 + 4 + f(6,12) + f(5,9) + f(4,9) + f(3,12)
= 8 + 5 + 3 + 5 + 3 + 4 + f(7,12) + f(6,9) + f(4,9) + f(3,12)
= 8 + 5 + 3 + 5 + 3 + 4 + 5 + 3 + 5 + f(4,12) + f(3,9)
= 8 + 5 + 3 + 5 + 3 + 4 + 5 + 3 + 5 + f(5,12) + f(5, 9)+ f(3,9)
= 8 + 5 + 3 + 5 + 3 + 4 + 5 + 3 + 5 + f(6,12) + f(6,9)+ f(5, 9)+ f(3,9)
= 8 + 5 + 3 + 5 + 3 + 4 + 5 + 3 + 5 + f(7,12) + f(6,9)+ f(6,9)+ f(5, 9)+ f(3,9)
= 8 + 5 + 3 + 5 + 3 + 4 + 5 + 3 + 5 + 5 + 3 + 3 + 4 + 6
= 21 + 15 + 16 + 10
= 36 + 26
= 62

2. (10 points – 2 points each)

(a) What is the decimal equivalent of the binary number: $\mathbf{(1001101)_2}$? (Show your work.)

Answer: **77**

Solution:

$(1 \times 2^6) + (1 \times 2^3) + (1 \times 2^2) + (1 \times 2^0)$
64 + 8 + 4 + 1
77

(b) What is the binary equivalent of the decimal number: $\mathbf{(232)_{10}}$? (Show your work.)

Answer: **11101000**

Solution:
232/2 = 116 remainder = 0
116/2 = 58 remainder = 0
58/2 = 29 remainder = 0
29/2 = 14 remainder = 1
14/2 = 7 remainder = 0
7/2 = 3 remainder = 1
3/2 = 1 remainder = 1
1/2 = 0 remainder = 1

(c) Solve for **x** in the equation: $\mathbf{\log_x 512 = 3}$. (Show your work.)

Solution: $x^3 = 512$, **x = 8**

LAST **FIRST**

(d) The "access policy" for a stack is:_____ In _____Out.

FIRST **FIRST**

(e) The "access policy" for a queue is:_____ In_____Out.

3. (15 points – 5 points each)

Find the closed form expression in terms of the parameter N (and M where indicated) or an exact value if the summation limits are known, for each of the following summations. Show all of your work, an answer alone is not sufficient to receive full credit.

a) $\sum_{i=0}^{N}(4i+8) =$

Solution: $2N^2 + 10\,N + 8$

$$\sum_{i=0}^{N}(4i+8) = 4\sum_{i=0}^{N} i + 8\sum_{i=0}^{N} 1 = \frac{4N(N+1)}{2} + 8(N+1) = 2N^2 + 2N + 8N + 8 = 2N^2 + 10N + 8$$

b) $\sum_{i=1}^{3N-5}(2i+5) =$

Solution: $9N^2 - 12N - 5$

$$\sum_{i=1}^{3N-5}(2i+5) = 2\sum_{i=1}^{3N-5} i + 5\sum_{i=1}^{3N-5} 1 = \frac{2(3N-5)(3N-4)}{2} + 5(3N-5) = 9N^2 - 12N - 5$$

c) $\sum_{i=17}^{33}(4i-6) =$

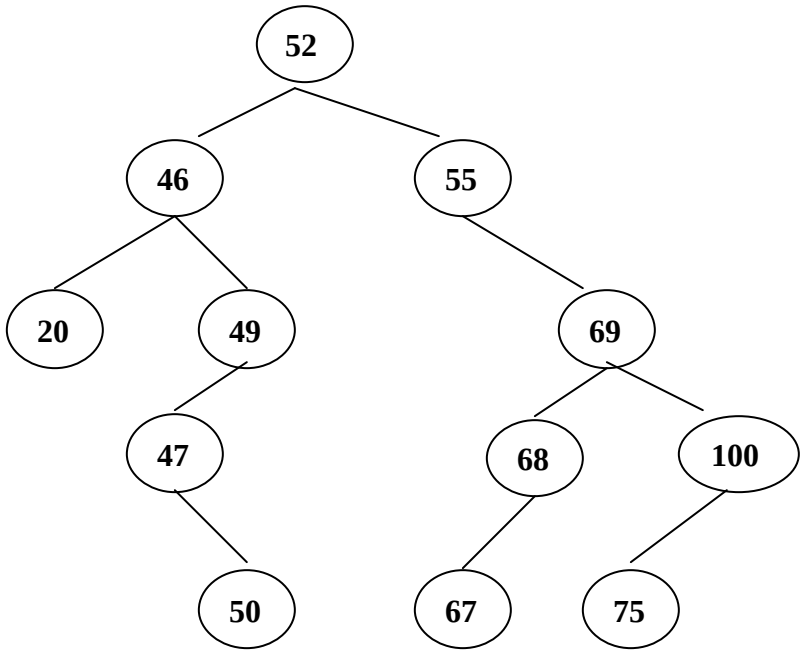
Solution: 1598

$$\sum_{i=17}^{33}(4i-6) = \left(4\sum_{i=1}^{33} i - 4\sum_{i=1}^{16} i\right) - \left(6\sum_{i=1}^{33} 1 = 6\sum_{i=1}^{16} 1\right) = \frac{4(33)(34)}{2} - \frac{4(16)(17)}{2} - 6(33) + 6(16)$$

$$2(33)(34) - 2(16)(17) - 6(33) + 6(16) = 2244 - 544 - 198 + 96 = 1598$$

4. (15 points – 3pts(a), 5pts(b), 7pts(c))

Given the following Binary Tree, answer the questions below :



(a) Is this a valid Binary Search Tree? (circle one) YES NO

(b) List the nodes of this tree in the order that they are visited in a preorder traversal:

52	46	20	49	47	50	55	69	68	67	100	75

first node visited — last node visited

(c)Execute the algorithm shown below using the tree shown above. List the output in the spaces below and leaving any unused spaces blank. Assume that the initial call is: **P4(root, 50)** and that the tree nodes and pointers are defined as shown.

```
  struct treeNode{
       int data;
          struct treeNode *left, *right;
  }
  struct treeNode *tree_ptr;

  void P4(struct tree_ptr *node_ptr,  int key)
{
  if (node_ptr != NULL)
  {
     P4(node_ptr->right, (key + 15));
     if (node_ptr->data > key){
          P4(node_ptr->left,  (key - 5));
          printf("%d ",node_ptr->data);
  }
```

```
}
```

100 46 52