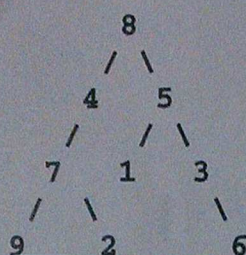


Name _____ Entry Number _____ Gp No. _____

Your Lab day _____ Your TA _____

Answer all the questions in the space provided for each question. You may use the answer script for rough work.

1. [2] Write the preorder traversal of the following tree:



8 - 4 - 7 - 9 - 2 - 5 - 1 - 3 - 6

2. [9] Complete the following function which accepts a sorted linked list in ascending order, with each node containing an integer, and returns the list after removing all nodes which appear more than once. Thus given the input

34-40-55-55-62-78, it should return 34-40-62-78

node *duplicate(node *A) {

~~node~~ node *p = A;

if (p == NULL)

{ cout << "In Empty list.";

return; }

while (p->next != NULL) {

while ((A->data) == (A->next->data)) {

node *h = A;

~~int~~ word1 = A->data;

A = A->next;

delete(h); if (A->next == NULL) break;

if (A->data == word1)

{ node *n = A;

A = A->next;

delete(h); if (A == NULL) return;

(cont) p = A;

~~p = A;~~ if (p->next == NULL) break;

while (p->data == p->next->data)

{ node *h = p;

~~int~~ word2 = p->data;

p = p->next;

delete(h);

~~if (p == NULL) return;~~

if (p->next == NULL) break;

if (p->data == word2)

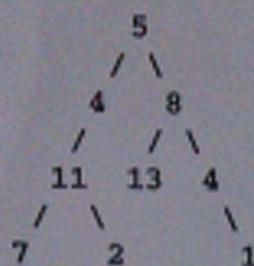
{ node *h = p;

p = p->next;

delete(h);

if (p == NULL) return;

3. [10] Given a tree and an integer, write a function which returns true if there is a path from the root down to a leaf, such that product of all the values along the path equals the given integer, otherwise return false. For example given the tree



Root-to-leaf paths for the above tree :

path 1: 5 4 11 7

path 2: 5 4 11 2

path 3: 5 8 13

path 4: 5 8 4 1

For this problem, we will be concerned with the product of the values of such a path -- for example, the product of the values on the 5-4-11-7 path is $5 * 4 * 11 * 7 = 1540$.

bool funct (node * A, int n) {
 if (p->data is divisible on then return false;
 else return funct (p->left, n/p->data)
 || funct (p->right, n/p->data);
}

no checking of whether root is null or not.
 no terminating condition

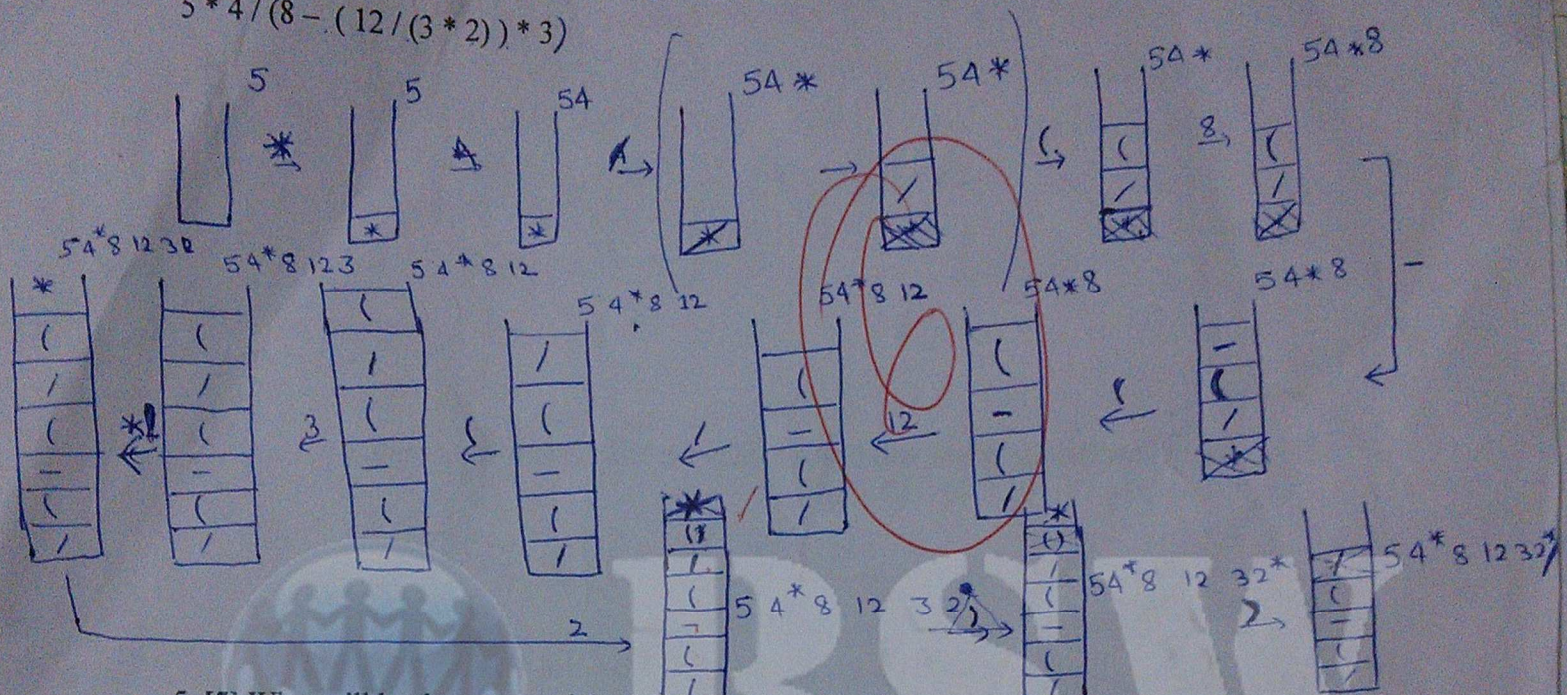
no correct logic

4

4. [6] Convert the following expression into postfix notation using a stack. Show the entries in the stack every time a character is read from the expression.

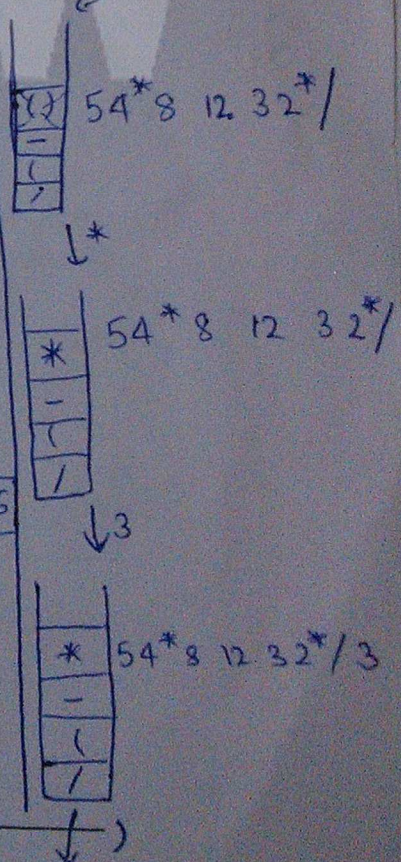
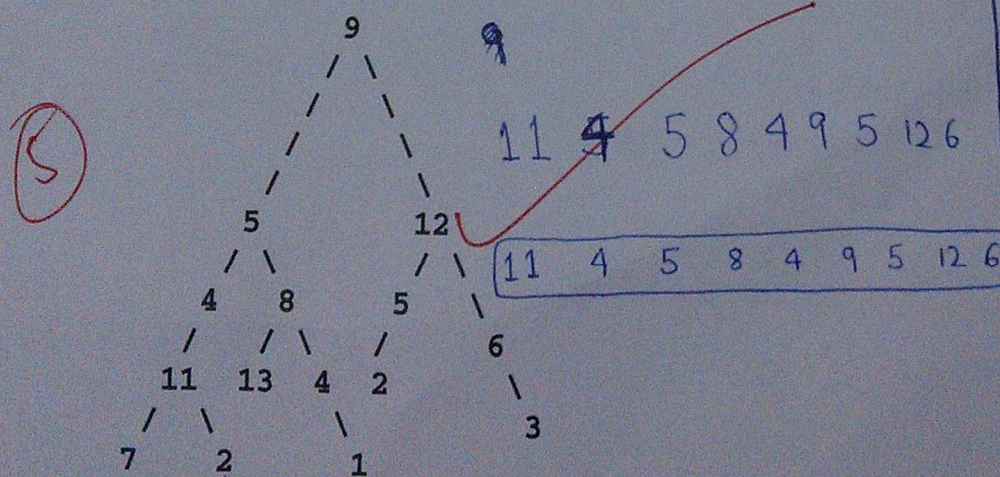
The character on the arrow is the character read from expression

$$5 * 4 / (8 - (12 / (3 * 2)) * 3)$$

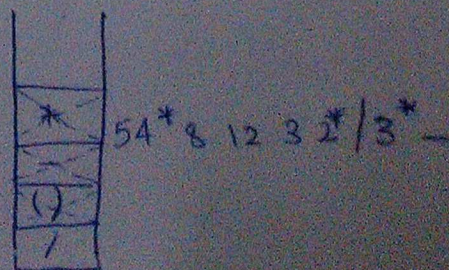


5. [5] What will be the output of the following function if it takes the given tree as input?

```
void wonder ( node *p) {
    wonder ( p-> left);
    if (p-> right != NULL || p-> left != NULL){
        cout << p-> data << " ";
    }
    wonder (p-> right);
}
```



$$5 4 * 8 12 3 2 * / 3 * - /$$



6. [3] What is the maximum number of nodes in a binary tree if it contains 120 leaf nodes? Justify your answer.

Max nodes = 240 (Correct answer is infinite)

③ → As we go down, no of nodes(max) is doubled. In 8th level we have 128 nodes(max). But we need 120. So we ~~del~~ remove 8 nodes ~~exist~~ + 4 nodes from above level, 2 from above that & 1 from above that is removed. total nodes = $(1 + 2 + 4 + 8 + 16 + 32 + 64 + 128) - (1 + 2 + 4 + 8) = 16 + 15$

7. [5] Indicate how the list below will change when we call the following routine
first → 57 - 55 - 92 - 40 - 62 - 84 - 78

void magic(node *first)
{
 struct node *p = first,
 *q = NULL,
 *r;

while (p != NULL)
{

 r = q;
 q = p;
 p = p->next;
 q->next = r;

}

q = first; → error: first = q;
 should be there

Output:

first → 78 - 84 - 62 - 40 - 92 - 55