Hints for Exam 1 review problems

1. [20 pts] Transform the following infix expression into its equivalent postfix expression using a stack. Trace the state of the operator stack as each character of the infix expression is processed. Show the contents of the operator stack at the indicated points in the infix expressions (points A, B and C).

R + P * Q / T – ( F – J + K ) * N

Resulting Postfix Expression: R P Q * T / F J – K + N * –

2. [12 pts] Evaluate the following postfix expression, showing the values in the stack at each indicated point in the Postfix string (points A, B, and C).

8 5 * 4 6 + / 2 8 * – 4 5 + *

The Final value of the expression is –108
3. [10 pts] Express using summations the number of arithmetic operations executed by the following code in lines 6 and 9. You don’t have to evaluate the summations.

```c
1   sum =0;
2   a =0; b= 5;
3   for ( i=0, i <n; i++) {
4       for (j=5; j<=m; j++) {
5           for ( k = 1; k < j;  k++){
6             f = sum + b +k ;
7         }
8       }
9       a = (a+ i * 7)/ 4;
10   }
```

\[
\sum_{i=0}^{n-1} \left( \sum_{j=5}^{m} \sum_{k=1}^{j-1} 2 \right) + 3
\]

4. [20 pts] Evaluate the following using summation rules. No points will be awarded if the summation rules are not used. Show all of your work to get credits on this question.

```c
10(\sum_{i=1}^{n} i - \sum_{i=1}^{n-21} i )
10 \left[ \frac{n(n+1)}{2} \right. - \left. \frac{(n-21)(n-20)}{2} \right]
5 \left[ n^2 + n - \frac{n^2 - 41n + 420}{2} \right]
= 210n - 2100
```
\[ \sum_{i=0}^{n} 4i - 2 \]

\[ 4 \sum_{i=0}^{n} i - \sum_{i=0}^{n} 2 \]

\[ 4 n( n + 1) / 2 - 2( n - 0 + 1) \]

\[ = 2 n^2 + 2 n - 2n - 2 \]

\[ = 2 n^2 - 2 \]

5. a) [10 pts] Consider the circular implementation of array based queue, that we discussed in the class, with size = 35. How many elements can be added to the queue in following situations?

i) front = –1, rear = –1 ______35_______ (as Q is empty)

ii) front = 0, rear = 16 ______18_______ (first 17 slots are filled up)

iii) front = 6, rear = 0 ______5_______ (see below)

iv) front = 16, rear = 16 ______34_______
5. b) [10 pts] Assuming that there are \( n \) items in a stack and \( m \) items in a queue, what is the time complexity of performing following operations on the array based queues and stacks that we have studied in the class. Express your result in terms of big-O notation.

i) one pop operation______O( 1 )__________

ii) \( m \) enqueue operations____O( m )__________

6. a). [ 6 pts ] For the following function write the recurrence relation to indicate the total number of operations \( T(n) \). You are not required to solve it.

```c
int operation( int  x,  int n )
{
    if (n < 0)
        return 5;
    else return ( x - 3 + operation( x, n - 3 ) ) * operation(x,n - 3);
}```
\[ T(n) = 2T(n-3) + 5 \]

6 b)
int puzzle(int N)
{
    if (N == 1) return 1;
    if (N % 2 == 0)
        return (1 + puzzle(N/2));
    else
        return (2 + puzzle(3*N+1));
}

What do the following function calls evaluate to?
puzzle(3):
= return (2 + puzzle(10))
= ... + (1 + puzzle(5))
= + (2 + puzzle(16))
= + (1 + puzzle(8))
= + (1 + puzzle(4))
= +(1 + puzzle(2))
= +(1 + puzzle(1))
= +(1 + 1)
= +(1 + 2)
= +(1 + 3)
= +(1 + 4)
= +(2 + 5)
= +(1 + 7)
= return(2 + 8)
= 10

puzzle(5): \hspace{1cm} \textbf{Answer: 7}\hspace{1cm}

6c) Write a recursive function which returns the number of integers greater than 50 in the array dd containing k elements.

int count(int dd[], int k)
{
    if (k < 0) return 0;
    else
if (dd[k] <= 50)
  return count(dd, k -1);
else
  return 1 + count (dd,k - 1)
}

More Problems:

7.
1. read the 2 dimensional array elements into A[][]
2. find  r_large[i], the largest value in each row
3. for the array r_large find the largest value.

This algorithm can now be expanded further to include more detailed steps.

OR
1. read the 2 dimensional array elements into A[][]
2. set max = A[1,1]
3. for i = 1 to n do steps 4 to 5
4. for j= 1 to m do step 5
5. if A[i,j] > max, set max = A[i,j]
6. Print max.

8.
1. Let n be the given integer. Let r be the reversed integer.
2. Set r = 0.
3. while n > 0 do steps 4 to 6
4. m = n mod 10
5. r = r * 10 + m
6. n = n/10
7. print r

9.
   a) Nk, as there is one multiplication in each loop
   b) 2 x 40 x t = 80 t
   c) 2 multiplications 40x20 times that is 1600 multiplications + 1 multiplication 40 times.
      Total: 1640 multiplications

10. $10^3 = N^3$ => $N = 30$
How to find the log values?

\[ b = \log_a m, \quad a^b = m \]

To find \( \log_2 16 \) use \( 2^b = 16 \)
This gives \( b = 4 \).

To find \( \log_2 64 \) use \( 2^b = 64 \)
This gives \( b = 6 \).

\[ \frac{\log 16}{28 \text{ ms}} = \frac{\log 64}{x} \]

\[ \frac{\log 16}{28 \text{ ms}} = \frac{\log 64}{x} \Rightarrow \]

\[ \frac{4}{28 \text{ ms}} = \frac{6}{x} \]

\[ x = 42 \text{ ms} \]

12. Suppose each of the following expressions represent the number of logical operations in an algorithm as a function of \( n \), the size of the list being manipulated. For each expression determine the dominant term and classify the algorithm in big-O terms.

a) \( 4n - 30 \) \( \quad O(n) \)

b) \( 10 n + 20 \log_2 n \) \( \quad O(n) \)

c) \( 6 \log_2 n + 30 \) \( \quad O(\log_2 n) \)

d) \( 5n + 40 \) \( \quad O(n) \)

e) \( 10 n + 2 n \log_2 n \) \( \quad O(n \log n) \)

f) \( 6 n + 2 (\log_2 n)^2 \) \( \quad O(n) \)

g) \( 20 n^3 + 2 n^2 \log_2 n + n^3 \log_2 n + 5 n^2 \) \( \quad O(n^3 \log_2 n) \)
h) 5 \quad O(1)

i) 10 n^2 + 2 n \log_2 n \quad O(n^2)

13. Given an array A of size n containing elements in random order, an array B of size n containing elements organized in increasing order, a stack C containing n elements, express the worst time complexity of following operations in terms of Big-O. Assume efficient algorithms are being used.

a) to print the last element of array A. \quad O(1)
   as no need to search for the last element

b) to print the last element of array B. \quad O(1)
   as no need to search for the last element

c) to print smallest element of array A. \quad O(n)
   as one has to scan all the n items to search for the smallest

d) to add a new element to C. \quad O(1)
   Push needs one operation

e) to search for a specific element in B \quad O(\log n)
   One can use binary search as the elements are ordered