Quiz-1

1. Solve the following recurrence relation using the expansion method. [10]

$T(n) = T(n-1) + c_1 n^2 + c_2$

$T(1) = c$

Answer:

$$T(n) = T(n-1) + c_1 n^2 + c_2$$

$$= (T(n-2) + c_1(n-1)^2 + c_2) + c_1 n^2 + c_2$$ **// Give 0 marks if this step is not correct**

$$= T(n-2) + c_1(n^2 + (n-1)^2) + 2c_2$$

$$= (T(n-3) + c_1(n-2)^2 + c_2) + c_1(n^2 + (n-1)^2) + 2c_2$$

$$= T(n-3) + c_1(n^2 + (n-1)^2 + (n-2)^2) + 3c_2$$ **// total 3 marks if this step is correct**

The kth term is:

$$= T(n-k) + c_1(n^2 + (n-1)^2 + (n-2)^2 + (n-k+1)^2) + kc_2$$ **// total 5 marks if this step is correct**

Substituting, k = n-1

$$T(n) = T(1) + c_1(2^2 + 3^2 + \cdots + n^2) + (n-1)c_2$$ **// total 10 marks if this step is correct**

$$= c + c_1\left(\frac{n(n+1)(2n+1)}{6} - 1\right) + (n-1)c_2$$

2. Write an algorithm for the tower of Hanoi problem with **four** towers. In addition to the source and destination tower, there are two temporary towers. The goal is to move $\boldsymbol{n}$ discs from the source tower to the destination tower using temporary towers. For all $\boldsymbol{n \geq 4}$, your algorithm should take less number of moves than the number of moves in the three towers solution. [15]

Give 0 marks if pseudo-code or a C code is not given

```
void move(int n, char *src, char *dst, char *tmp1, char *tmp2)
{
     If (n == 0) {
         return;
     }
     if (n == 1) {           // deduct 5-marks if the algorithm
                             // don't handle this base case
        printf("move from %s to %s\n", src, dst);
        return;
     }
     move(n-2, src, tmp1, dst, tmp2);      // give zero if n-1 discs
                                           // are moved at this step
     printf("move from %s to %s\n", src, tmp2);
     printf("move from %s to %s\n", src, dst)
     printf("move from %s to %s\n", tmp2, dst);
     // deduct 5-marks if all three print statements are not given
     move(n-2, tmp1, dst, src, tmp2); // give zero if this step is missing
}
```