Homework Assignment Set 3

Problems 2, 3,4 and 5 have been adopted from the notes and hand outs of the Algorithm course instructed by Prof. Abhijit Das, Dept. Of CSE, IIT Kharagpur

I would like to thank Prof. Abhijit Das

Q1. (a) The conventional algorithm for evaluating a polynomial $a_n x^n + a_{n-1} x^{n-1} + \ldots + a_0$ at x =c can be expressed as pseudo code by :

```
polynomial(c,a0 , a1,.....an  real numbers ){
power = 1; y = a0;
for i = 1 to n{
 power = power * c;
 y = y + ai*power;
}
```

i. Evaluate $3x^2+x+1$ at x=2 by working through each step of the algorithm
ii. Exactly how many additions and multiplications are used to evaluate a polynomial of degree n at x = c

(b) There is a more efficient algorithm (in terms of number of multiplications and additions) for evaluating polynomials. It is known as **Horner's rule.** The following pseudo code shows how to use this method to find the value of $a_n x^n + a_{n-1} x^{n-1} + \ldots + a_0$ at x =c

```
horner (c,a0 , a1,.....an  real numbers ){
     y = an;
     for i = 1 to n{
        y = y * c  + an-i;
}
```

i. Evaluate $3x^2+x+1$ at x=2 by working through each step of the algorithm
ii. Exactly how many additions and multiplications are used to evaluate a polynomial of degree n at x = c

Q2.Arrange the following functions in the increasing order of their growth rates:

$(\sqrt{2})^n$, $2^{\sqrt{n}}$, $n^2 (\log n)^2$, $(n \log n)^2$, $n^{\sqrt{n}}$, n^n, $(\log n)^n$

Q3. Establish that two sorted arrays each of size `n/2` can be merged in O(`n`) time **[Moderate]**

Q4. You are given two sorted arrays $\mathbf{a_0,a_1.....a_{s-1}}$ and $\mathbf{b_0,b_1,.......b_{t-1}}$ of integers and let n=s+t. You may assume that all a_i and b_j are distinct. Your task is to compute that the number of pairs of (i,j) of indices with $a_i>b_j$. Where **0<=i<=s-1 and 0<=j<=t-1**. Devise an algorithm to solve the problem in `O(n)` time. **[Hard]**

Q5. Deduce that the following function recursively computes Fibonacci series in linear time **[Moderate]**

```
int F (int n, int *Fprev )
{
    int Fn_1, Fn_2;



    if(n == 0){
          *Fprev = 1;
          return 0;
    }

    if( n==1){
          *Fprev = 0;
          return 1;

    }

    Fn_1 = F(n-1,&Fn_2);
    *Fprev = Fn_1;
    return(Fn_1+ Fn_2);

}
```

Q6. Sparse matrix denotes a square matrix having few non-zero elements in each row.

(a) Define a data type for storing a sparse matrix

(b) Write a function that add two sparse matrices

(c) Write a function that multiplies two `nxn` sparse matrices in $O(n^2)$ time

---