# Assignment 2 - Algorithm analysis (2 weeks)

**Problem 1**. Compute the computational complexity of the following loops:

a. 
```
for (cnt1 = 0, i = 1; i <= n; i++)
        for (j = 1; j <= n; j++)
                cnt1++;
```

b. 
```
for (cnt2 = 0, i = 1; i <= n; i++)
        for (j = 1; j <= i; j++)
                cnt2++;
```

c. 
```
for (cnt3 = 0, i = 1; i <= n; i *= 2)
        for (j = 1; j <= n; j++)
                cnt3++;
```

d. 
```
for (cnt4 = 0, i = 1; i <= n; i *= 2)
        for (j = 1; j <= i; j++)
                cnt4++;
```

**Problem 2**. Determine the complexity of the addition, multiplication, and transposition of nxn matrices as follows:

a. Addition

```
for (i = 0; i < n; i++)
        for (j = 0; j < n; j++)
                a[i][j] = b[i][j] + c[i][j];
```

b. Multiplication

```
for(i = 0; i < n; i++)
{
        for(j = 0; j < n; j++)
        {
                sum = 0;
                for(k = 0; k < n; k++)
                        sum = sum + b[i][k] * c[k][j];
                a[i][j] = sum;
        }
}
```

c. Transposition

```
for(i = 0; i < n - 1; i++)
        for(j = i+1; j < n; j++)
```

```
{
        tmp = a[i][j];
        a[i][j] = a[j][i];
        a[j][i] = tmp;
}
```

**Problem 3**. Find the complexity of the following function:

```
int Min(int a[], int n)
{
        int i, min;
        min = a[0];
        for(i = 1; i < n; i++)
                if (a[i] < min)
                        min = a[i];
        return min;
}
```

**Problem 4.** Prove that:

a. $4n^2 + 7n + 1 = O(n^2)$

b. $n^2 - 3n + 1 = \Omega(n)$

c. $log(2n + k) = \Theta(log(n))$, where $k$ is a constant

d. $\sum_{i=1}^{n} \log(i) = O(n \log(n))$

**Problem 5.** Give an efficient algorithm to determine if there exists an integer $i$ such that $a_i = i$ in an array of integers $a_1 < a_2 < a_3 < \ldots < a_n$. What is the running time of your algorithm?

**Problem 6**. A polynomial is defined as follows:

$$P(x) = \sum_{i=0}^{n} a_i x^i$$

a. Write a function to compute $P(x)$.

b. Evaluate the complexity of the function.

c. Write another function to compute $P(x)$ according to Hörner scheme

$$P(x) = \sum_{i=0}^{n} a_i x^n = a_0 + x(a_1 + x(a_2 + \ldots + x(a_{n-1} + a_n x)\ldots))$$

d. Compare the complexity of the above functions.

**Problem 7.** Given a snip code as follows:

```
p = 1;
for (i = 1; i <= n; i++)
        p = p * a;
```

a. Count the number of multiplications.

b. Can you find any solution that requires less running time than above algorithm?