
Submissions: This assignment is due on the 3rd of September, 2020. Please note:

1. Each student must submit his or her own assignment.

2. Solutions should preferably be typed in Latex, MSWord or other such word processing software, or printed clearly. In any case, have a single pdf file of the solutions, containing solutions to all problems in order (if you dont solve a problem write ”No solution” against its number), and upload to its folder on elearn.

3. You must write your name and UUID clearly on your submitted assignment.

Academic Integrity: You are encouraged to work in groups, but everyone must write out their own solutions. Absolutely no word to word copying is allowed. If you have worked with other students on the assignment or referred to external sources, please mention all names and sources on your assignment.

Partial solutions: Document your efforts at solving a problem even if you cannot solve it. Write why your approach failed.

Problem 1 [30 pts]: Recall the algorithm for the “Towers of Hanoi” problem that we covered in class. It takes $2^n - 1$ moves to move $n$ disks. Prove that no algorithm can do better than this.

Problem 2 [20 pts]: Consider a function $T(k, l)$ defined on two positive integral parameters. The recursive definition of the function is as follows:

$$T(k, l) = \begin{cases} 
k & \text{if } l = 1 \\
\ell & \text{if } k = 1 \\
T(k - 1, \ell) + T(k, \ell - 1) + 1 & \text{otherwise}
\end{cases}$$

Prove that $T(k, \ell) = \binom{k + \ell}{\ell} - 1$.

Problem 4 [20 pts]: Suppose we are given the polynomial $A(x) = a_0 + a_1x + \ldots + a_{n-1}x^{n-1}$. Show how we can evaluate it at a point $x_0$ using $O(n)$ arithmetic operations. (This is not the running time – we are just counting the number of arithmetic operations, but a real estimation of the run time will take bit complexity issues into account.)

Hint: $A(x) = a_0 + x(A_1(x))$ where $A_1(x) = a_1 + a_2x + \ldots + a_{n-1}x^{n-2}$.

Problem 3 [5 + 5 + 20 = 30 pts]: Consider a set of pairs of values $(x_0, y_0), (x_1, y_1), \ldots, (x_{n-1}, y_{n-1})$, where $x_i \neq x_j$ for $i \neq j$. We want to show that there is a degree $n$ polynomial $P(z)$ that can be “fitted” through these points. To do this show the following:

- Show that the polynomial $P_i(z) = \frac{(z-x_0)(z-x_{i-1})(z-x_{i+1})\ldots(z-x_{n-1})}{(x_i-x_0)(x_i-x_{i-1})(x_i-x_{i+1})\ldots(x_i-x_{n-1})}$ has the property that $P_i(x_i) = 1$ and $P_i(x_j) = 0$ if $j \neq i$. The product uses all terms $(z-x_j)$ except $(z-x_i)$ in the numerator.
• Now show how to construct the required polynomial.

• Given the \( n \) pairs of values analyze the fastest algorithm that you can come up with to evaluate the polynomial in its coefficient representation. To do this focus on analyzing how to evaluate the product in the numerator of \( P_i(z) \).