Problem 1 (Extended Euclidean Algorithm)

Study the extended Euclidean algorithm (page 937 from CLRS book).

1. Compute the values (d, x, y) that EXTENDED-EUCLID$(899, 493)$ returns;
2. Assume that the greatest common divisor of two number a and n is 1. Using the extended Euclidean algorithm as black-box, devise an algorithm that computes a number x such that the remainder of the division of numbers $a\dot{x}$ and n is 1;
3. Given $a_0, \ldots, a_n$, devise an algorithm that computes numbers $x_0, \ldots, x_n$ such that $gcd(a_0, \ldots, a_n) = a_0x_0 + \ldots + a_nx_n$. Show that the number of divisions performed by your algorithm is $O(n + \lg(\max\{a_0, \ldots, a_n\}))$.

Problem 2 (Asymptotic Analysis)

1. Which is asymptotically larger: $\log(\log^* n)$ or $\log^*(\log n)$?
   ($\log^* n$ *is the number of times the logarithm function must be iteratively applied before the result is less than or equal to 1*—see also: `https://en.wikipedia.org/wiki/Iterated_logarithm`)
2. Let $f(n)$ and $g(n)$ to be asymptotically non-negative functions. Using the basic definition of Θ-notation prove that $max(f(n), g(n)) = \Theta(f(n) + g(n))$.
3. CLRS 3.1-4
4. CLRS 3.1-8

Problem 3 (Divide and Conquer)

1. You are calling FIND-MAXIMUM-SUBARRAY$(A, 1, 8)$ on an array of 8 elements (see the code in page 72 in the CLRS book). Write down the state of the operating system stack after each recursive call (you should begin with an empty stack and end with an empty stack).
2. CLRS 4.1-1
3. CLRS 4.1-4

Problem 4 (Master Theorem)

Give asymptotic upper and lower bounds for $T(n)$ in each of the following recurrences. Assume that T(n) is constant for $n \leq 2$. Make your bounds as tight as possible, and justify your answers.

- $T(n) = 2T(n/2) + n^4$
- $T(n) = T(7n/10) + n$
- $T(n) = 16T(n/4) + n^2$
- $T(n) = 7T(n/3) + n^2$
- $T(n) = 7T(n/2) + n^2$
- $T(n) = 2T(n/4) + \sqrt{n}$