1. Consider the following variant of the selection algorithm that we saw in class:

```
MODIFIEDSELECT(A, j):
    divide A into groups of 9 elements each
    find the median \( b_i \) of each group by brute force
    let \( B = \{b_1, b_2, b_3, \ldots, b_{\lceil n/9 \rceil} \} \)
    \( b = \text{MODIFIEDSELECT}(B, \lceil n/18 \rceil) \)
    partition A into \( A_{\text{less}} \) and \( A_{\text{greater}} \) using \( b \) as pivot
    if \( |A_{\text{less}}| = j \)
        return \( b \)
    else if \( |A_{\text{less}}| > j \)
        return \( \text{MODIFIEDSELECT}(A_{\text{less}}, j) \)
    else
        return \( \text{MODIFIEDSELECT}(A_{\text{greater}}, j - |A_{\text{less}}|) \)
```

Analyze the running time of this modified algorithm by writing a recurrence relation for it and solving it. Briefly justify the recurrence relation that you derived.

2. (a) Euclid’s algorithm for finding the greatest common divisor of two non-negative numbers \( a, b \) is the following:

```
EUCLID(a, b):
    if \( b > a \)
        return \( \text{EUCLID}(b, a) \)
    else
        if \( b = 0 \)
            return \( a \)
        else
            return \( \text{EUCLID}(b, a \mod b) \)
```

(i) Show by induction that the algorithm correctly computes the greatest common divisor of \( a \) and \( b \).

(ii) Assuming that the mod operation and other basic arithmetic operations take constant time, show that the running time of the algorithm is polynomial in the input size. \([\text{Note that the input size is } \Theta(\log a + \log b)]\)

(b) A slow version of the Euclid algorithm is the following.

```
SLOWEUCLID(a, b):
    if \( b > a \)
        return \( \text{SLOWEUCLID}(b, a) \)
    else
        if \( b = 0 \)
            return \( a \)
        else
            return \( \text{SLOWEUCLID}(b, a - b) \)
```

\(^1\)The selection algorithm that we saw in class divides \( A \) into groups of 5 elements.
Show via a class of examples that the above algorithm can take exponential time in the input size.