I certify that every answer in this assignment is the result of my own work; that I have neither copied off the Internet nor from any one else's work; and I have not shared my answers or attempts at answers with anyone else.

1: Using the pseudo-code syntax introduced in class, write an algorithm EGYPTIANMULTIPLICATION(n1, n2) to multiply two positive integers n1, n2 using the Egyptian method of multiplication we covered in class through an example (see lecture file intro.pdf).
Assume that the only arithmetic operation available is addition of integers. (Doubling is achieved by addition.) Comparison operators ($>$, $\leq$, $<$, $\geq$, $=$) are available. For the second last operation, finesse the delete rows step by calling a separate algorithm FINDROWS which returns an array containing the row numbers that would not be deleted; and use it for the last step.
You could use two arrays for the two columns with a common index for the rows, i.e., L[i], R[i] would contain the i th entry for the left and right columns.
Include meaningful comments such that a grader can understand your algorithm.
(Recap: Array indices start at 1; each array has a size attribute which contains the upper index of the valid segment of the array. We saw an example in the lab.)

```
EGYPTIANMULTIPLICATION(n1, n2)
answer ← 0
L ← [1]
R ← [n2]
i ← 2
while i < n1 and L[i − 1] + L[i − 1] < n1 do
      L[i] ← L[i-1] + L[i-1]
      R[i] ← L[i-1] + L[i-1]
      i ← i + 1
F ← FINDROWS(L, n1)
for j ← 1 to F.size do
     answer ← answer + R[F[j]
return answer

FINDROWS(L, n1)
F ← [L[L.size]]
sum ← F[1]
i ← 2
for k ← L.size down to 1 do
      if sum + L[k] = n1 do
            F[i] ← k
            i ← i + 1
      else if sum + L[k] < n1
            sum ← sum + L[k]
return F
```